



Proceedings of  
**The Annual Conference of the  
Prognostics and Health Management  
Society 2012**

**PHM'12**

ISBN - 978-1-936263-05-9

Minneapolis, Minnesota, USA

September 23 - 27, 2012

***Edited by:***

Indranil Roychoudhury

Jose R. Celaya

Abhinav Saxena

# Table of Contents

## Full Papers

- 2 Anomaly Detection Based on Information-Theoretic Measures and Particle Filtering Algorithms  
*Marcos E. Orchard, Benjamin Olivares, Matias Cerda, and Jorge F. Silva*
- 9 Application of Inductive Monitoring System to Plug Load Anomaly Detection  
*Christopher Teubert and Scott Poll*
- 17 Health Assessment of Composite Structures in Unconstrained Environments Using Partially Supervised Pattern Recognition Tools  
*Emmanuel Ramasso, Vincent Placet, Rafael Gouriveau, Lamine Boubakar, and Noureddine Zerhouni*
- 28 Use of COTS Functional Analysis Software as an IVHM Design Tool for Detection and Isolation of UAV Fuel System Faults  
*Octavian Niculita, Phil Irving, and Ian K. Jennions*
- 46 A Batch Detection Algorithm Installed on a Test Bench  
*Jerome Lacaille and Valerio Gerez*
- 53 Spare Parts Inventory Control for Non-Repairable Items Based on Prognostics and Health Monitoring Information  
*Leonardo R. Rodrigues and Takashi Yoneyama*
- 63 Initial Condition Monitoring Experience on a Wind Turbine  
*Eric Bechhoefer, Mathew Wadham-Gagnon, and Bruno Boucher*
- 71 A Distributed Approach to System-Level Prognostics  
*Matthew Daigle, Anibal Bregon, and Indranil Roychoudhury*
- 83 Peer-to-Peer Collaborative Vehicle Health Management - The Concept and An Initial Study  
*Yilu Zhang, Xinyu Du, and Mutasim Salman*
- 93 Fault Diagnosis of Gas Turbine Engine LRUs Using the Startup Characteristics  
*Kyusung Kim, Onder Uluyol, Girija Parthasarathy, and Dinkar Mylaraswamy*
- 103 Correction of Data Gathered by Degraded Transducers for Damage Prognosis in Composite Structures  
*Kyle R. Mulligan, Nicolas Quaegebeur, Patrice Masson, and Sylvian Letourneau*
- 111 Application of Symbolic Regression to Electrochemical Impedance Spectroscopy Data for Lubricating Oil Health Evaluation  
*Carl Byington, Nicholas Mackos, Garrett Argenna, Andrew Palladino, Johan Reimann, and Joel Schmitigal*
- 122 Detecting Injector Deactivation Failure Modes In Diesel Engines Using Time And Order Domain Approaches  
*Mitchell Lebold, Scott Pflumm, Jeffrey Banks, Jonathon Bednar, Karl Reichard, Kenneth Fischer, and Joseph Stempnik*
- 134 Constructing an Efficient Self-Tuning Aircraft Engine Model for Control and Health Management Applications  
*Jeffrey B. Armstrong and Donald L. Simon*
- 147 Maintenance Planning with Prognostics for Systems Located In Various Places  
*F. Camci, M. Sevкли, M. Karakas, and I. K. Jennions*
- 152 Bayesian Framework Approach for Prognostic Studies in Electrolytic Capacitor under Thermal Overstress Conditions  
*Chetan S. Kulkarni, Jose R. Celaya, Kai Goebel, and Gautam Biswas*
- 163 Real-Time Corrosion Monitoring of Aircraft Structures with Prognostic Applications  
*Douglas Brown, Duane Darr, Jefferey Morse, and Bernard Laskwoski*
- 175 Markov Modeling of Component Fault Growth Over A Derived Domain of Feasible Output Control Effort Modifications  
*Brian Bole, Kai Goebel, and George Vachtsevanos*
- 183 Condition Indicators for Gears  
*Renata Klein*

- 191 Statistical Health Grade System against Mechanical failures of Power Transformers  
*Chao Hu, Pingfeng Wang, Byeng D. Youn, Wook-Ryun Lee, and Joung T. Yoon*
- 204 A Comparison Study of Methods for Parameter Estimation in the Physics-based Prognostics  
*Dawn An, Joo-Ho Choi, and Nam H. Kim*
- 215 Structural Health Monitoring on Metallic Aircrafts Using Flexible and Bulk PZT Transducers: Case of Corrosion Detection and Crack Localization  
*Hamza Boukabache, Christophe Escriba, Sabeha Zedek, and Jean-Yves Fourniols*
- 224 A Tutorial for Model-based Prognostics Algorithms based on Matlab Code  
*Dawn An, Joo-Ho Choi, and Nam H. Kim*
- 233 A Design Methodology of Optimized Diagnosis Functions for High Lift Actuation Systems  
*Christian Modest and Frank Thielecke*
- 249 Fault Diagnostics Using Network Motif Signature  
*Tsai-Ching Lu, David L. Allen, Yilu Zhang, and Mutasim A. Salman*
- 256 Condition-Based Maintenance with Both Perfect and Imperfect Maintenance Actions  
*Phuc Do Van, Alexandre Voisin, Eric Levrat, and Benoit Lung*
- 265 Prognostics and Health Monitoring in the Presence of Heterogeneous Information  
*Gregory Bartram and Sankaran Mahadevan*
- 276 A Multi-Resolution Experimental Methodology for Fatigue Mechanism Verification of Physics-Based Prognostics  
*Jian Yang, Wei Zhang, and Yongming Liu*
- 285 Reliability Growth Analysis of Satellite Systems  
*John W. Evans, Mark P. Kaminskiy, and Luis D. Jr Gallo*
- 292 Multi-turbine Associative Model for Wind Turbine Performance Monitoring  
*Onder Uluyol and Girija Parthasarthy*
- 302 An Evidential Evolving Prognostic Approach and its Application to PRONOSTIA's Data Streams  
*Lisa Serir, Emmanuel Ramasso, and Noureddine Zerhouni*
- 311 Time Distribution Mapping: a Generic Transient Signal Monitoring Technique for Prognostic Methods  
*Michael E. Sharp and J Wesely Hines*
- 319 Prognostics Method for Analog Electronic Circuits  
*Arvind S. S. Vasan, Bing Long, and Michael Pecht*
- 326 An Efficient Deterministic Approach to Model-based Prediction Uncertainty Estimation  
*Matthew Daigle, Abhinav Saxena, and Kai Goebel*
- 336 Lithium-ion Battery Remaining Useful Life Estimation Based on nonlinear AR Model combined with Degradation Feature  
*Datong Liu, Yue Luo, Yu Peng, Xiyuan Peng, and Michael Pecht*
- 343 Sensor Selection with Grey Correlation Analysis for Remaining Useful Life Evaluation  
*Yu Peng, Yong Xu, Datong Liu, and Xiyuan Peng*
- 353 An Unscented Kalman Filter Based Approach for the Health-Monitoring and Prognostics of a Polymer Electrolyte Membrane Fuel Cell  
*Xian Zhang and Pierluigi Pisu*
- 362 Transducer Development for Nonintrusive Load Monitoring of Rotating Machinery  
*Mateja Putic, Nathan K. Brown, and Paul Muskopf*
- 369 A Reasoning Architecture for Expert Troubleshooting of Complex Processes  
*Abdul Naveed, Jiaming Li, Bhaskar Saha, Abhinav Saxena, and George J. Vachtsevanos*
- 377 Integrated Fatigue Damage Diagnosis and Prognosis Under Uncertainties  
*Tishun Peng, Jingjing He, Yongming Liu, Abhinav Saxena, Jose R. Celaya, and Kai Goebel*
- 388 A Cost-Benefit Approach to Evaluating Engine Health Monitoring Systems  
*Grant Gordon, Chris Hickenbottom, and Dinkar Mylaraswamy*

- 396 An Approach to Prognostic Decision Making in the Aerospace Domain  
*Edward Balaban and Juan J. Alonso*
- 416 An Integrated Framework for Model-Based Distributed Diagnosis and Prognosis  
*Anibal Bregon, Matthew Daigle, and Indranil Roychoudhury*
- 427 Uncertainty Representation and Interpretation in Model-based Prognostics Algorithms based on Kalman Filter Estimation  
*Jose R. Celaya, Abhinav Saxena, and Kai Goebel*
- 437 Prognostic-Enabling an Electrohydrostatic Actuator (EHA) System  
*Sonia Vohnout, David Bodden, Byoung U. Kim, Neil Kunst, Patrick Edwards, Bill Gleeson, Dennis Cascio, Steve Brzuskiewicz, Roy Wagemans, Matthew Rounds, and N. Scott Clements*

## Poster Papers

- 450 Performance Based Anomaly Detection Analysis of a Gas Turbine Engine by Artificial Neural Network Approach  
*Amar Kumar, Alka Srivastava, Avisekh Banerjee, and Alok Goel*
- 458 Health Monitoring of a Power Supply Using Multivariate Regression  
*Leonardo R. Rodrigues, Joao P. P. Gomes, Takashi Yoneyama, and Roberto K. H. Galvao*
- 464 Failure Prognostics of a Hydraulic Pump Using Kalman Filter  
*Joao P. P. Gomes, Bruno P. Leao, Wlamir O. L. Vianna, Roberto K. H. Galvao, and Takashi Yoneyama*
- 469 Fleet Wide Asset Monitoring: Sensory Data to Signal Processing to Prognostics  
*Preston Johnson*
- 476 Prognostics Assessment Using Fleet-Wide Ontology  
*Gabriela Medina-Oliva, Alexandre Voisin, Maxime Monnin, Flavien Peysson, and Jean-Baptiste Leger*
- 487 **Author Index**



# Full Papers

# Anomaly Detection based on Information-Theoretic Measures and Particle Filtering Algorithms

Marcos E. Orchard<sup>1</sup>, Benjamín Olivares<sup>1</sup>, Matías Cerda<sup>1</sup> and Jorge F. Silva<sup>1</sup>

<sup>1</sup>*Electrical Engineering Department, Universidad de Chile, Santiago 8370451, Chile  
morchard@ing.uchile.cl*

## ABSTRACT

This paper presents an anomaly detection module that uses information-theoretic measures to generate a fault indicator from a particle-filtering-based estimate of the posterior state pdf of a dynamic system. The selected measure allows isolating events where the particle filtering algorithm is unable to track the process measurements using a predetermined state transition model, which translates into either a sudden or a steady increment in the differential entropy of the state pdf estimate (evidence of an anomaly on the system). Anomaly detection is carried out by setting a threshold for the entropy value. Actual data illustrating aging of an energy storage device (specifically battery state-of-health (SOH) measurements [A-h]) are used to test and validate the proposed framework.

## 1. INTRODUCTION

Anomaly detection modules (Zhang *et al.*, 2011; Orchard *et al.*, 2011) play an important role within Prognostics and Health Management (PHM) systems since they constitute the first step in the implementation of fault diagnosis and failure prognosis schemes (Orchard and Vachtsevanos, 2009). In most real applications, the anomaly detector requires to perform its task simultaneously minimizing both the false alarm rate and detection time (early detection). The latter is of paramount importance since the setup of online prognostic algorithms, and particularly those based on particle filtering algorithms (Orchard *et al.*, 2008; Orchard *et al.*, 2009), requires a proper characterization of the initial state pdf to provide adequate estimate of the remaining useful life (RUL) of monitored equipment.

Classical anomaly detection methods rely on a model of the system to measure a discrepancy between the actual measurements and a predetermined pattern of operation. A variety of techniques have been proposed to achieve this task, including tools from estimation theory, failure sensitive filters, multiple hypothesis filter detection, generalized likelihood ratio tests, and model-based approaches (Isermann and Balle, 1997; Ayhan *et al.*, 2006;

Zhou *et al.*, 2008; Khan and Rahman, 2009). Other methods focused on statistical analysis techniques, reasoning tools, spectral methods and information theory (Tolani *et al.*, 2005; Zhou *et al.*, 2008; Ibrahim *et al.*, 2008).

In the particular case of the battery state-of-health (SOH) monitoring and prognosis (Orchard *et al.*, 2010; Orchard *et al.*, 2011), there are still issues regarding the proper representation of regeneration (self-recharge) phenomena. Self-recharge phenomena are characterized by sudden, momentary, and occasionally considerable regeneration of the battery capacity that tends to fade in time faster than the typical SOH degradation time constant. These changes, related to physicochemical aspects and temperature/load conditions during charge and discharge cycles, are particularly important in the case of Li-Ion batteries because they often alter the trend of the SOH prediction curve, thus affecting the performance of prognostic modules that depend on Bayesian estimation algorithms to compute initial conditions for their associated predictive models.

This paper presents a solution for this problem that is based on a combination of a PF-based state estimators and information-theoretic measures that allows to detect rare events within the evolution of the fault condition under analysis. The paper is structured as follows: Section 2 introduces the basics on particle filtering (PF) anomaly detection modules, as well as information-theoretic measures applied to sequential Monte Carlo algorithms. Section 3 focuses on describing the case study that is used in this research to illustrate and validate the potential of the proposed detection approach, which corresponds to the analysis of capacity regeneration phenomena in a set of data depicting the battery state-of-health (SOH, [A-h]) degradation. Section 4 presents the proposed anomaly detection scheme and the results obtained for the case study of interest. Finally, Section 5 states the main conclusions.

Marcos E. Orchard *et al.* This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

## 2. PARTICLE FILTERING, ANOMALY DETECTION AND INFORMATION-THEORETIC MEASURES

Nonlinear filtering is defined as the process of using noisy observation data  $Y = \{y_t, t \in \mathbb{N}\}$  to estimate at least the first two moments of a state vector  $X = \{x_t, t \in \mathbb{N}\}$  governed by a dynamic nonlinear, non-Gaussian state-space model.

From a Bayesian standpoint, a nonlinear filtering procedure intends to generate an estimate of the posterior probability density function  $p(x_t | y_{1:t})$  for the state, based on the set of received measurements. Particle Filtering (PF) is an algorithm that intends to solve this estimation problem by efficiently selecting a set of  $N \gg 1$  particles  $\{x_t^{(i)}\}_{i=1 \dots N}$  and weights  $\{w_t^{(i)}\}_{i=1 \dots N}$ , such that the state pdf may be approximated (Doucet, 1998; Doucet *et al.*, 2001; Andrieu *et al.*, 2001; Arulampalam *et al.*, 2002) by the empirical distribution:

$$\tilde{\pi}_t^N(x_t) = \sum_{i=1}^N w_t^{(i)} \delta(x_t - x_t^{(i)}), \quad (1)$$

and the values of the particles weights  $w_t^{(i)} \propto w(x_t^{(i)})$  can be computed by:

$$w(x_t) = \frac{\pi_t(x_t)}{q_t(x_t)} \propto \frac{p(y_t | x_t) p(x_t | x_{t-1})}{q_t(x_t | x_{t-1})}, \quad (2)$$

$$w(x_t^{(i)}) = w_{t-1}^{(i)} \cdot \frac{p(y_t | x_t^{(i)}) p(x_t^{(i)} | x_{t-1}^{(i)})}{q_t(x_t^{(i)} | x_{t-1}^{(i)})} \quad \forall i \in \{1, \dots, N\}$$

where  $q_t(x_t)$  denotes the importance sampling density function (Arulampalam *et al.*, 2002; Doucet *et al.*, 2001). The choice of this importance density function  $q_t(x_t)$  is critical for the performance of the particle filter scheme. In the particular case of nonlinear state estimation, the value of the particle weights  $w_t^{(i)}$  is computed by setting the importance density function equal to the *a priori* pdf for the state, i.e.,  $q_t(x_t | x_{t-1}) = p(x_t | x_{t-1})$  (Arulampalam *et al.*, 2002). Although this choice of importance density is appropriate for estimating the most likely probability distribution according to a particular set of measurement data, it does not offer a good estimate of the probability of events associated to high-risk conditions with low likelihood. In this sense, this paper explores the possibility of using information-theoretic measures to analyze PF-based estimates of the state pdf in a dynamic system, with the purpose of detecting this type of events in a timely manner.

### 2.1 Particle Filtering for Anomaly Detection

PF-based anomaly detection modules (Kadirkamanathan *et al.*, 2002; Verma *et al.*, 2004; Orchard and Vachtsevanos, 2009; Zhang *et al.*, 2011; Orchard *et al.*, 2011) have been used in the past to identify abnormal conditions in

nonlinear, non-Gaussian dynamic systems. The objective in this type of implementations is to fuse the information that is available at a feature vector (measurements) to generate estimates of the *a priori* state pdf that could be helpful when determining the operating condition (mode) of a system and deviations from desired behavioral patterns. This compromise between model-based and data-driven techniques is accomplished by the use of a PF-based module built upon the nonlinear dynamic state model (3):

$$\begin{cases} x_d(t+1) = f_b(x_d(t) + n(t)) \\ x_c(t+1) = f_t(x_d(t), x_c(t), \omega(t)) \\ \text{Features}(t) = h_t(x_d(t), x_c(t), v(t)) \end{cases}, \quad (3)$$

where  $f_b$ ,  $f_t$  and  $h_t$  are non-linear mappings,  $x_d(t)$  is a collection of Boolean states associated with the presence of a particular operating condition in the system (normal operation, fault type #1, #2),  $x_c(t)$  is a set of continuous-valued states that describe the evolution of the system given those operating conditions,  $\omega(t)$  and  $v(t)$  are non-Gaussian random variables that characterize the process and feature noise signals, respectively. Since the noise signal  $n(t)$  is a measure of uncertainty associated with Boolean states, it is recommendable to define its probability density through a random variable with bounded domain. For simplicity,  $n(t)$  may be assumed to be zero-mean i.i.d. uniform white noise.

PF-based detection modules provide a framework where customer specifications (such as false alarm rate and desired probability of detection) can be easily managed and incorporated within the algorithm design parameters. However, the analysis of the relationship that exists between the number of particles and the detection time still depends on general guidelines inspired in empirical experience (for example, “the more particles are used, the longer is the detection time”).

The problem of early detection using PF-based approaches has also been discussed in (Orchard *et al.*, 2008), where a Risk-Sensitive PF (RSPF) framework complements the benefits of the classic approach by representing the probability of rare and costly events within the formulation of importance density function to generate more particles in high-risk regions of the state-space. Mathematically, the importance distribution is set as:

$$q(d_t, x_t | d_{t-1}^{(i)}, x_{t-1}^{(i)}, y_{1:t}) = \gamma_t \cdot r(d_t) \cdot p(d_t, x_t | y_t), \quad (4)$$

where  $d_t$  is a set of discrete-valued states representing fault modes,  $x_t$  is a set of continuous-valued states that describe the evolution of the system given those operating conditions,  $r(d_t)$  is a positive risk function that is dependent on the fault mode, and  $\gamma_t$  is a normalizing constant.

Although the approach presented offered better performance in terms of the detection time, it still required the definition of a risk importance sampling distribution.



In this sense, the use of information-theoretic measures offers an interesting alternative that complements the paradigm of PF-based anomaly detection modules, under the assumption that an anomaly should affect the qualitative behavior of the state pdf estimate. The following section focuses on the most important concepts that need to be taken into account when implementing these measures to analyze and characterize sampled versions of the *posterior* distribution.

## 2.2 Information-Theoretic Measures Applied to Particle-filtering Algorithms

Several examples that incorporate information-theoretic measures to analyze the outputs of particle filtering algorithms can be found in literature (Ajgl and Šimandl, 2011; Lanz, 2007; Boers *et al.*, 2010; Skoglar *et al.*, 2009). Most of those are related to uncertainty characterization, optimality testing, and evaluation of control strategies. In particular, this research focuses on the widely known differential entropy measure (Cover and Thomas, 1991).

Entropy is a measure of uncertainty that is associated to a probability measure. In particular, the differential entropy  $H$  of a probability density function  $p(x)$  is given by:

$$H(p) \triangleq -\int p(x) \log(p(x)) dx \quad (5)$$

Entropy-related applications for particle filtering algorithms generally aim at evaluating how many i.i.d. samples does the filtering algorithm require to represent regions of the state space that accumulate the majority of the probability mass, for a given state pdf estimate  $p(x)$ . For example, in (Liverani *et al.*, 2006) the authors propose the use of entropy to evaluate the pertinence of resampling procedures in a particle filtering algorithm aimed at estimating the states of a partially observed Markov chain. Instead, it is sought to generate an average weight for sampled particles, which depends on the distance that exists between the estimated and the actual value of the states.

In other applications, such as in (Ryan, 2008), the authors formulate a control strategy for a mobile sensor that intends to track an object, where the merit function depends on particle-filtering-based estimates and information-theoretic measures. Basically this approach uses entropy to characterize the uncertainty of the estimated pdf, and proposes a resampling method that intends to minimize the conditional entropy between the state of the tracked object and observed data, for a given control strategy.

Although the definition of differential entropy introduced in (5) allows straightforward computation in most cases, few considerations are required when trying to compute it in the case of particle-filtering-based estimates of the conditional state pdf's. Indeed, using (5), the differential entropy of the conditional state pdf estimate, given a set of measurements  $y_1, \dots, y_t$ , is defined as:

$$H(p(x_t | y_t)) = -\int p(x_t | y_t) \log(p(x_t | y_t)) dx_t, \quad (6)$$

where the *a posteriori* state pdf estimate can be inferred from the likelihood of measurement  $y_t$ , the *a priori* state estimate  $p(x_t | y_{t-1})$ , and the probability of acquiring the current measurement using Bayes Theorem:

$$p(x_t | y_t) = \frac{p(y_t | x_t)}{p(y_t | y_{t-1})} p(x_t | y_{t-1}). \quad (7)$$

Thus, replacing (7) in (6) and applying properties of the logarithm, it is possible to write:

$$H(p(x_t | y_t)) = \log(p(y_t | y_{t-1})) + \dots - \int p(x_t | y_t) [\log(p(y_t | x_t)) + \log(p(x_t | y_{t-1}))] dx_t \quad (8)$$

In addition, given that in this specific case all distributions correspond to particle-filtering estimates, both the *a priori* state estimate and the probability of measured data can be approximated by their corresponding sampled versions, as in (9)-(10):

$$p(x_t | y_{t-1}) \approx \sum_{i=1}^N w_{t-1/t-1}^{(i)} p(x_t^{(j)} | x_{t-1}^{(i)}), \quad (9)$$

$$p(x_t | y_t) \approx \sum_{i=1}^N w_{t/t}^{(i)} \delta(x_t - x_t^{(i)}), \quad (10)$$

where  $w_{t-1/t-1}^{(i)}$  and  $w_{t/t}^{(i)}$  are the *a priori* and posterior weight of the particle ( $i$ ), respectively. After using (9)-(10) in (8):

$$H(p(x_t | y_t)) = \log(p(y_t | y_{t-1})) + \sum_{j=1}^N w_{t/t}^{(j)} [\log(p(y_t | x_t^{(j)})) + \dots + \log\left(\sum_{i=1}^N w_{t-1/t-1}^{(i)} p(x_t | x_{t-1}^{(i)})\right)] \quad (11)$$

The term  $p(y_t | y_{1:t-1})$  in (11) can be computed through its sampled version:

$$p(y_t | y_{t-1}) \approx \sum_{i=1}^N w_{t/t-1}^{(i)} p(y_t | x_t^{(i)}), \quad (12)$$

where  $w_{t/t-1}^{(i)}$  are the particle weights. As a final result, the differential entropy of the particle-filtering estimate of the posterior state pdf can be computed as in (13) (Orguner, 2009):

$$H(p(x_t | y_t)) = \log\left(\sum_{i=1}^N w_{t/t-1}^{(i)} p(y_t | x_t^{(i)})\right) + \sum_{j=1}^N w_{t/t}^{(j)} \left[ \log(p(y_t | x_t^{(j)})) + \log\left(\sum_{i=1}^N w_{t-1/t-1}^{(i)} p(x_t^{(j)} | x_{t-1}^{(i)})\right) \right] \quad (13)$$

The latter expression will be of use when evaluating the uncertainty associated to online estimates in dynamic processes.

### 3. CASE STUDY: PF-BASED SELF-RECHARGE DETECTION IN LITHIUM-ION BATTERIES

An appropriate case study has been selected to demonstrate the efficacy of an anomaly detection module based on a PF state estimator and information-theoretic measures. Consider the case of energy storage devices, particularly of Li-Ion batteries, where continuous switching between charge and discharge cycles may cause momentary increments in the battery SOH (capacity regeneration). These sudden increments directly affect RUL estimates in classic prognostic schemes since the state pdf estimate has to be adjusted according to new measurements (thus modifying long-term predictions), while the observed phenomenon typically disappears after a few cycles of operation. Particularly in the case of Li-Ion batteries, the regeneration phenomena can produce an unexpected short-term increment of the battery SOH of about 10% of the nominal capacity.

The analysis of the aforementioned phenomena will be done using data registering two different operational profiles (charge and discharge) at room temperature. On the one hand, charging is carried out in a constant current (CC) mode at 1.5[A] until the battery voltage reached 4.2[V] and then continued in a constant voltage mode until the charge current dropped to 20[mA]. On the other hand, discharge is carried out at a constant current (CC) level of 2[A] until the battery voltage fell to 2.5[V]. Impedance measurements provide insight into the internal battery parameters that change as aging progresses. Repeated charge and discharge cycles result in aging of the batteries. Impedance measurements were done through an electrochemical impedance spectroscopy (EIS) frequency sweep from 0.1[Hz] to 5[kHz]. The experiments were stopped when the batteries reached end-of-life (EOL) criteria, which was a 40% fade in rated capacity (from 2[A-h] to 1.2[A-h]).

Two main operating conditions are thus distinguished: the *normal* condition reflects the fact that the battery SOH is slowly diminishing as a function of the number of charge/discharge cycles; while the *anomalous* condition indicates an abrupt increment in the battery SOH (regeneration phenomena). These phenomena, which are characterized by sudden, momentary, and occasionally considerable regeneration of the battery capacity, are related to physicochemical aspects and temperature/load conditions during charge and discharge cycles. In the case of Li-Ion batteries, the detection of such events is extremely important for a proper implementation of prognostic schemes since they often alter the trend of the SOH prediction curve, thus affecting the performance of prognostic modules based on Bayesian algorithms to estimate the initial conditions of their predictive models.

The study of battery SOH involves the analysis of many different factors, but this research is focused on one of the most critical features associated to it: the life cycle. Life

cycle models usually consider a specific term that aims to incorporate part of the phenomenology that is present in the battery degradation process. This term is the Coulomb efficiency,  $\eta_c$ , which is a measure for the amount of usable energy that is expected for the discharge cycle in progress, compared to the capacity exhibited by the battery during the previous discharge cycle (Orchard *et al.*, 2010). Equations (14)-(15) show how this term can be included in a nonlinear dynamic model that can be used for SOH estimation purposes:

$$\begin{cases} x_1(k+1) = \eta_c x_1(k) + x_2(k)x_1(k-1) + \omega_1(k) \\ x_2(k+1) = x_2(k) + \omega_2(k) \end{cases} \quad (14)$$

$$y(k) = x_1(k) + v(k), \quad (15)$$

where  $k$  is the cycle index;  $x_1$  is a state representing the battery SOH;  $x_2$  is a state associated with an unknown model parameter that is required to explain minor differences with respect to the expected behavior (which are specific to the monitored battery);  $y(k)$  is the measured SOH;  $\omega_1$ ,  $\omega_2$  and  $v$  are non-Gaussian noises.

Although model (14)-(15) enables the implementation of Bayesian filtering techniques to monitor degradation processes in Li-Ion batteries, it results inadequate when trying to detect and isolate the short and long-term effect of regeneration (self-recharge) phenomena. This fact motivates the development of anomaly detection modules, either based on PF-algorithms as in (Orchard *et al.*, 2011), or information-theoretic measures as the present research proposes.

### 4. ANOMALY DETECTION MODULE BASED ON INFORMATION-THEORETIC MEASURES AND PARTICLE FILTERING ALGORITHMS

The primary concept behind the proposed anomaly detection scheme is that any sudden abnormal behavior in the system should affect the distribution of the PF-based posterior state estimate. This is caused by the fact that, under abnormal operating conditions, the system model no longer represents the best choice for the importance sampling distribution. As a consequence, the weights associated to particles with low-likelihood undergo strong corrections, increasing the differential entropy of the aforementioned conditional state pdf.

In this sense, the proposed detection module considers a particle filtering algorithm based on model (14)-(15), as state estimator module, and a stage where expression (13) is used to compute the differential entropy of the posterior pdf estimate. The resulting entropy (which is computed at each cycle of operation) corresponds to the output of the detection module. Anomaly detection is carried out by setting a threshold for the entropy estimate. It is of special interest to isolate events where the entropy increases in a

sudden manner, or where steadily increases since in both cases it evidences that the particle filtering algorithm is unable to track the process measurements using the predetermined state transition model.

Validation of the proposed scheme is performed on SOH degradation data from an accelerated test at Prognostic Center of Excellence at NASA Ames, where it is of particular interest to detect the moments when battery SOH measurements evidence the existence of capacity regeneration (also known as “self-recharging”) phenomena (Orchard *et al.*, 2010). Furthermore, as an additional contribution of the analysis, we will assess what is the actual impact (in terms of early detection) that is associated to an increment in the number of particles in the PF state estimator; taking into account the performance of the proposed entropy-based indicator as the filter uses more particles.

Figure 1 shows the actual SOH degradation data and the results obtained by the proposed detection scheme when 30 particles are used in the implementation of the PF algorithm. In particular, Figure 1 a) illustrates on the difficulty the PF estimator undergoes when the *a priori* transition model (14)-(15) is used to track the degradation of battery capacity in the presence of self-recharge phenomena (for example at the 19<sup>th</sup>, 30<sup>th</sup>, and 47<sup>th</sup> cycles of operation). As it has been mentioned before, the concept behind the entropy-based detection module is to recognize these issues, providing in those cases an indicator that may be used as alarm signal.

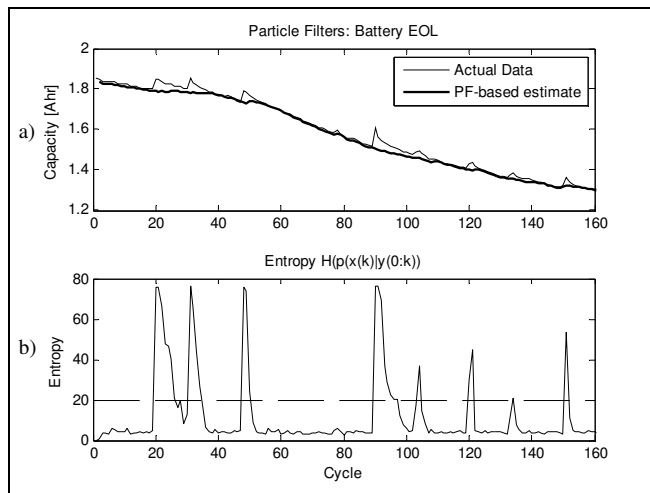


Figure 1. Evolution of the entropy of the posteriori state pdf, using 30 particles within the implementation of the particle filtering algorithm

Figure 1b) depicts the evolution in time of the entropy of the posterior PF-based estimate, for the case of battery SOH degradation. On the one hand, it is important to note that the entropy of the *posterior* state pdf, in absence of self-recharge phenomena, tends to stabilize until it almost behaves like a constant function of time. This stabilization

value directly depends on the variance of process and observation noise kernels in equations (14) and (15), which are the actual sources of uncertainty within the implementation of the particle-filtering-based estimator. On the other hand, Figure 1b) also shows that the entropy-based indicator experiences strong modifications on its value in the event of a self-recharge phenomenon (more than eight times in some cases, as in the 19<sup>th</sup>, 30<sup>th</sup> and 47<sup>th</sup> cycle of operation). This fact validates the use of the proposed approach for anomaly detection purposes, triggering the alarm whenever the differential entropy of the posterior state pdf is bigger than a given threshold (e.g., twice the stabilization value for the entropy of the estimate in the absence of capacity regeneration phenomena). However, it is still not clear if an increment on the number of particles would allow computing a lower threshold for the detection module, while simultaneously avoiding the generation of false alarms.

Figure 2 and Figure 3 provide critical information to answer the latter inquire. On the one hand, Figure 2 depicts the obtained results when using  $N=100$  particles in the PF-based estimator, which implies that the computational complexity of the algorithm increases more than three times. On the other hand, Figure 3 shows the case when 500 particles are used.

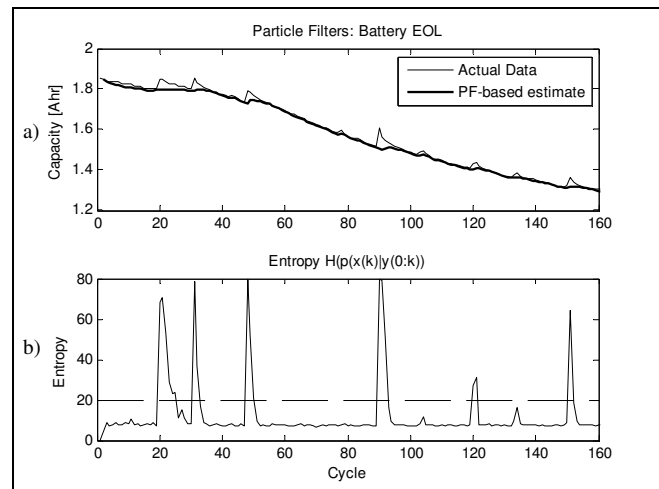


Figure 2. Evolution of the entropy of the posteriori state pdf, using 100 particles within the implementation of the particle filtering algorithm

Although an increment in the number of particles  $N$  reduces the amount of time that is required to reach a stabilization value for the entropy of the *posterior* pdf, it does not necessarily increase the capability of the filter to track the evolution of the system in the event of capacity regeneration. As a consequence, the proposed anomaly indicator improves its detection capability (and reduces the probability of false alarms) as the number of particles increases. Moreover, the resulting fault feature (either in the case of  $N=30$  or  $N=100$  particles) allows to easily

implement an anomaly detection module based a PF-based detection module (Orchard *et al.*, 2011), which uses the entropy indicator to perform the hypothesis testing and declare the anomaly, for a given false alarm rate.

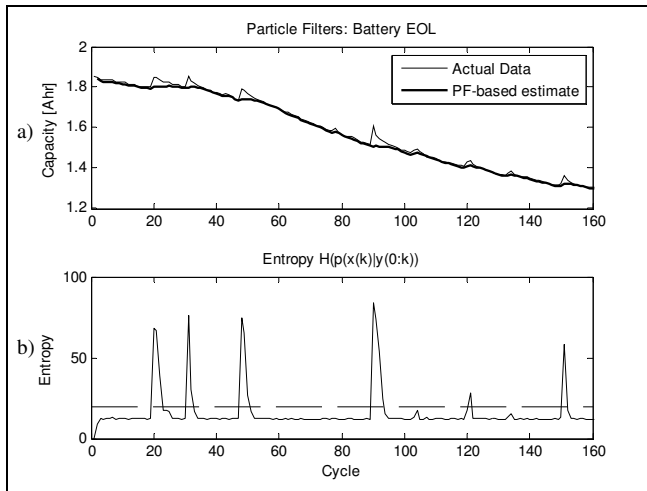


Figure 3. Evolution of the entropy of the posteriori state pdf, using 500 particles within the implementation of the particle filtering algorithm

Finally, it is important to note that a drastic increment in the number of particles (as shown in Figure 3) does not necessarily imply equivalent improvements in the capability of the anomaly detector. Furthermore, this research shows that using less than 100 particles is enough to achieve adequate performance both in terms of detection capabilities and computational effort for the estimation algorithm.

## 5. CONCLUSION

This paper presents an anomaly detection module that is based on a PF state estimator and information-theoretic measures, which aims at isolating self-recharge phenomena within the SOH degradation process of an energy storage device (Li-Ion battery). From obtained results, we surmise that the proposed anomaly detection approach, which computes a fault indicator from the entropy of the PF-based *posterior* state pdf estimate, is capable of isolating rare and sudden events –such as self-recharge phenomena in the degradation curve– in a simple and efficient manner. Empirical analysis on actual data from acceleration test shows that although an increment in the number of particles within the proposed scheme does improve the detection capability of the proposed approach (also reducing the probability of false alarms), although it does not necessarily compensate the raise on the computational cost of the estimation algorithm. As a result of the aforesaid analysis, an appropriate range for  $N$  (number of particles) is defined for the case study hereby described.

## ACKNOWLEDGEMENT

The authors want to thank CONICYT for its financial support through the project FONDECYT #1110070.

## REFERENCES

- Ajgl J., and M. Šimandl, (2011). “Particle Based Probability Density Fusion with Differential Shannon Entropy Criterion” Proceeding of the 14th International Conference on Information Fusion ISIF, pp. 803-810, Chicago, Illinois, USA.
- Andrieu, C., A. Doucet, E. Puskaya, (2001). “Sequential Monte Carlo Methods for Optimal Filtering,” in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds. NY: Springer-Verlag.
- Arulampalam, M.S., S. Maskell, N. Gordon, T. Clapp, (2002). “A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking,” *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174 – 188.
- Ayhan, B., M.Y. Chow, M.H. Song, (2006). “Multiple discriminant analysis and neural network based monolith and partition fault detection schemes for broken rotor bar in induction motors,” *IEEE T. Industrial Electronics*, Vol. 53, No. 4, pp 1298-1308.
- Boers, Y., J.N. Driessen, A. Bagchi, and P.K. Mandal, (2010). “*Particle Filter Based Entropy*,” Proceedings of FUSION 2010, Edinburgh, UK.
- Cover, T., and J. Thomas, (1991). “Element of Information Theory,” Wiley Interscience, New York, 1991.
- Doucet, A., (1998). “On sequential Monte Carlo methods for Bayesian Filtering,” Technical Report, Engineering Department, Univ. Cambridge, UK.
- Doucet, A., N. de Freitas, N. Gordon, (2001). “An introduction to Sequential Monte Carlo methods,” in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds. NY: Springer-Verlag.
- Ibrahim, A., M. El Badaoui, F. Guillet, F. Bonnardot, (2008). “A new bearing fault detection method in induction machine based on instantaneous power factor,” *IEEE T. Industrial Electronics*, Vol. 55, No. 12, pp 4252-4259.
- Isermann, R., and P. Balle, (1997). “Trends in the Application of Model-Based Fault Detection and Diagnosis of Technical Processes,” *Control Eng. Practice*, vol. 5, no. 5, pp. 709-719.
- Kadirkamanathan, V., P. Li, M.H. Jaward, and S.G. Fabri, (2002). “Particle filtering-based fault detection in nonlinear stochastic systems,” *International Journal of Systems Science*, 33(4), pp. 259-265.
- Khan, M. and M. Rahman, (2009). “Development and implementation of a novel fault diagnostic and protection technique for IPM motor drivers,” *IEEE T. Industrial Electronics*, Vol. 56, No. 1, pp. 85-92.

- Lanz, O., (2007). "An information theoretic rule for sample size adaptation in particle filtering," Proceedings of the 14th International Conference on Image Analysis and Processing (ICIAP), pp. 317–322.
- Liverani, S., and A. Papavasiliou, (2006). "Entropy Based Adaptive Particle Filter," 2006 IEEE Nonlinear Statistical Signal Processing Workshop.
- Orchard, M., G. Kacprzynski, K. Goebel, B. Saha, G. Vachtsevanos, (2008). "Advances in Uncertainty Representation and Management for Particle Filtering Applied to Prognostics," 2008 *International Conference on Prognostics and Health Management PHM 2008*, Denver, CO, USA.
- Orchard, M., (2009). *On-line Fault Diagnosis and Failure Prognosis Using Particle Filters. Theoretical Framework and Case Studies*, Publisher: VDM Verlag Dr. Müller Aktiengesellschaft & Co. KG, Saarbrücken, Germany, 108 pages. Atlanta: The Georgia Institute of Technology, Diss., 2007.
- Orchard, M. and G. Vachtsevanos, (2009). "A Particle Filtering Approach for On-Line Fault Diagnosis and Failure Prognosis," *Transactions of the Institute of Measurement and Control*, vol. 31, no. 3-4, pp. 221-246.
- Orchard, M., L. Tang, K. Goebel, G. Vachtsevanos, (2009). "A Novel RSPF Approach to Prediction of High-Risk, Low-Probability Failure Events," First Annual Conference of the Prognostics and Health Management Society, San Diego, CA, USA.
- Orchard, M., Tang, L., Saha, B., Goebel, K., and Vachtsevanos, G., (2010). "Risk-Sensitive Particle-Filtering-based Prognosis Framework for Estimation of Remaining Useful Life in Energy Storage Devices," *Studies in Informatics and Control*, vol. 19, Issue 3, pp. 209-218.
- Orchard, M., L. Tang, and G. Vachtsevanos, (2011). "A Combined Anomaly Detection and Failure Prognosis Approach for Estimation of Remaining Useful Life in Energy Storage Devices," *Annual Conference of the Prognostics and Health Management Society 2011*, Montreal, QB, Canada.
- Orguner, U., (2009). "Entropy Calculation in Particle Filters," Signal Processing and Communications Applications Conference.
- Ryan, A., (2008). "Information-theoretic tracking control based on particle filter estimate," AIAA Guidance Navigation and Control Conference
- Skoglar, P., U. Orguner, and F. Gustafsson, (2009). "On information measures based on particle mixture for optimal bearings-only tracking," Proceedings of IEEE Aerospace Conference 2009, Big Sky, Montana, USA
- Tolani, D., M. Yasar, S. Chin, A. Ray, (2005). "Anomaly Detection for Health Management of Aircraft Gas Turbine Engines," 2005 American Control Conference, pp. 459-464, Portland, OR, USA.
- Verma, V., G. Gordon, R. Simmons, S. Thrun, (2004). "Particle Filters for Rover Fault Diagnosis," *IEEE Robotics & Automation Magazine*, pp. 56 – 64.
- Zhang, B., T. Khawaja, R. Patrick, M. Orchard, A. Saxena, G. Vachtsevanos, (2009). "A Novel Blind Deconvolution De-Noising Scheme in Failure Prognosis," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 2, pp. 303-310.
- Zhang, B., C. Sconyers, C. Byington, R. Patrick, M. Orchard, and G. Vachtsevanos, (2011). "A Probabilistic Fault Detection Approach: Application to Bearing Fault Detection," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 5, pp. 2011-2018.
- Zhou, W., T. Habetler, R. Harley, (2008). "Bearing fault detection via stator current noise cancellation and statistical control," *IEEE T. Industrial Electronics*, Vol. 55, No. 12, pp 4260-4269, Dec. 2008.

# Application of Inductive Monitoring System to Plug Load Anomaly Detection

Christopher Teubert<sup>1</sup>, Scott Poll<sup>2</sup>

<sup>1</sup> *NASA USRP Intern, Iowa State University, Ames, IA, 50011, USA*  
*teubert@gmail.com*

<sup>2</sup> *NASA Ames Research Center, Moffett Field, 94035, USA*  
*scott.poll@nasa.gov*

## ABSTRACT

NASA Ames Research Center's Sustainability Base is a new 50,000 sq. ft. LEED Platinum office building. Plug loads are expected to account for a significant portion of the overall energy consumption. This is because building design choices have resulted in greatly reduced energy demand from Heating, Ventilation, and Air Conditioning (HVAC) and lighting systems, which are major contributors to energy consumption in traditional buildings. In anticipation of the importance of plug loads in Sustainability Base, a pilot study was conducted to collect data from a variety of plug loads. A number of cases of anomalous or unhealthy behavior were observed including schedule-based rule failures, time-to-standby errors, changed loads, and inter-channel anomalies. These issues prevent effective plug load management; therefore, they are important to promptly identify and correct. The Inductive Monitoring System (IMS) data mining algorithm was chosen to identify errors. This paper details how an automated data analysis program was created, tested and implemented using IMS. This program will be applied to Sustainability Base to maintain effective plug load management system performance, identify malfunctioning equipment, and reduce building energy consumption.

## 1. INTRODUCTION

Over the past several years there has been tremendous interest in green technologies and sustainable practices within the building industry. As technology improvements have reduced energy consumption from Heating, Ventilation, and Air Conditioning (HVAC) and lighting systems, plug loads constitute larger percentages of a building's total load. Managing plug loads can lead to dramatically reduced building energy consumption (Lobato, Pless, Sheppy, & Torcellini, 2011;

Christopher Teubert et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Kaneda, Jacobson, & Rumsey, 2010).

In preparation for deploying a plug load management system in Sustainability Base, which was not yet occupied at the time of this investigation, a pilot study was conducted in another office building on the NASA Ames campus (Poll & Teubert, 2012). The system monitored and controlled plug loads through the use of smart power strips, each of which had four channels (receptacles) for devices to be plugged into.

Over the course of the pilot study several issues were observed. Most serious of these were (i) failure of schedule-based plug load management rules to go into effect, (ii) failure of a device to go to low-power or standby mode, (iii) changing a device plugged into a channel, and (iv) inter-channel load relationship anomalies. These issues prevent effective plug load management; therefore, they are important to promptly identify and correct. We describe each of these issues in greater detail in the following paragraphs.

Schedule-based rule failures occur when rules to turn devices off or on at specified times, as commanded by the plug load management system, fail to go into effect. This could happen as a result of loss of communication or faulty hardware. These failures reduce the effectiveness of active plug load management, thereby increasing energy waste.

Time-to-standby failures are when a device fails to enter a low-power mode. This error, which can be symptomatic of a device malfunction, leads to greatly increased energy consumption.

Changed loads refers to a configuration change of the devices plugged into a power strip. Usually this means that a device has either been replaced with a newer model or that a different device has been plugged into that channel. A configuration change such as this is only an issue if the system administrators are not notified of the change. For example, changing loads without updating the associated schedule-based rules could lead to data loss or damage if a computer is inadver-

tently de-energized.

Inter-channel anomalies refer to a situation where the relationship between two channels is undesirable. One example of such an anomaly would be if a monitor is in active mode while the computer is off. The error could be symptomatic of malfunctioning equipment. In the future, this could also indicate failure of load-sensing control, which turns off devices based on the behavior of a 'master' device. For example, a rule could be created so that when a computer is off, the peripherals (speakers, printer, monitor) would be powered down as well. Load-sensing control was not investigated for the pilot study but it will be in Sustainability Base.

Developing a model-based system to identify the aforementioned anomalies for each channel would be labor intensive and would not scale to a plug load management system for an entire building with hundreds or thousands of loads. Therefore, it was decided to use a data-driven approach to do automated analysis. Data-driven algorithms (Kantardzic, 2011) are capable of analyzing vast amounts of data to pick out unusual or unhealthy behavior and therefore lend themselves nicely to building plug load management at NASA Ames' Sustainability Base.

The Inductive Monitoring System (IMS) tool (Iverson, 2004) was chosen for this application because of its ability to learn healthy behavior without having to create a complex model for each channel. IMS creates a knowledge base of nominal behavior from judiciously chosen training data sets. New plug-load data are then compared to healthy behavior to pick out anomalies. If not addressed, these anomalies could lead to increased power consumption, decreased effectiveness of the plug load management system, or even damage to plug load devices. Once an anomaly is identified, building personnel are automatically notified so that they may address the issue.

The Sustainability Base IMS application uses device power draw data collected by plug load monitoring power strips located in copy rooms, break rooms and at workstations. The power strips measure and transmit power draw once per second to a cloud-based data service which records minimum, mean, and maximum power draw at one minute intervals. The volume of the data (1440 records per device each day) makes it necessary to implement automated analysis.

The main contributions of this research are (i) observation of potential issues with plug load management, (ii) definition of raw and derived plug load parameters that identify different anomalies, (iii) application of Inductive Monitoring System to identify faulty plug load devices or improper usage, and (iv) development of an automated program to process plug load data and notify appropriate personnel of problems that require attention.

## 2. INDUCTIVE MONITORING SYSTEM

Inductive Monitoring System is a data mining algorithm designed to detect deviation from healthy system behavior. The first step in using IMS is off-line learning, or the establishment of a knowledge-base of healthy behavior. To do this a series of vectors of data previously determined to be healthy, or training data, are fed in one-by-one to the program. K-means clustering (Bradley & Fayyad, 1998) is used to group data into multi-dimensional clusters; different regions of the cluster space may represent different operating modes of the system. If the vector is determined to be close to one of the existing clusters, the cluster is expanded to include it. If the vector is too far from the clusters it becomes the beginning of a new cluster. Parameters are used to control how the clusters are expanded or created; the default IMS parameters were used in this study.

Once the healthy clusters are fully formed, new data sets are then analyzed. Each vector of the testing data is compared with the formed clusters, and the closest cluster is determined for comparison. The composite score is defined as the Euclidean distance between the vector and the closest point on the nearest cluster in multidimensional space. IMS also calculates the contribution, or local score, of each individual parameter to the composite score.

The IMS tool has been used in a number of complex systems. Following the Columbia (STS-107) accident in 2003, IMS was used to analyze the telemetry from four temperature sensors located in each wing of the orbiter. IMS analyzed data from launch/ascent and on-orbit and was able to detect anomalies much earlier than the monitoring systems used in mission control (Iverson, 2004).

An IMS based program has been used by the International Space Station (ISS) flight control team in mission control to monitor operations the Control Moment Gyroscopes (CMGs) and External Thermal Control System (ETCS). This program has successfully identified multiple anomalies in these systems (Iverson, Spirkovska, & Schwabacher, 2010).

### 2.1. Sustainability Base Plug Load IMS Application

Sustainability Base IMS will be used as an important tool for the building's health management. Output from IMS will allow operators to identify and address unhealthy plug load behavior promptly, thereby increasing the effectiveness of the plug load management system and maintaining high system efficiency. The most critical element in ensuring useful results from the IMS algorithm is the definition of input vectors. Sample training vectors are shown in Table 1. Each row is an input vector whose parameters are defined by the column headers. Combining raw and derived quantities is essential to permit visibility of different failure types. For Sustainability Base three parameters were chosen:

- Raw power draw: This is employed to discover cases of changed power loads. Changed power draw could also be symptomatic of a larger problem, or an example of normal behavior that has not been previously observed.
- Consecutive minutes in the idle mode power range, as applicable: This feature is meant to find cases where a device fails to go to standby or low-power mode. The ranges for idle modes were defined a priori.
- Piecewise function corresponding to the time when the device is drawing power, and zero, otherwise: These values are used by IMS to find cases of schedule-based rule failures, or times when the device is drawing power when it should not.

Power Draw (W)	Idle Time (hrs)	On Time (hrs)
100	0.54	10.33
102	0.56	10.35
160	0.00	10.36
102	0.02	10.38
...	...	...

Table 1. Sample Training Vectors

The power strips transmit power draw measurements every second to a cloud-based data service, which records data at one minute intervals. During mode transitions a device will sometimes spend parts of a minute in different modes, causing the system to record an average power draw value in a range where the device does not typically operate. If included in the training data, these transitory values can prevent the IMS system from detecting anomalies when a changed device draws steady-state power in these intermediate ranges. Recall from the input vector that duration is only captured for the idle mode in the second parameter, the first parameter of the input vector is only raw power draw.

In order to eliminate this phenomenon the raw power draw is filtered to remove intermediate values. Values where the relative difference is greater than 10% are filtered out, where the relative difference is defined as the change in power draw divided by the average, as shown in Eq. (1). The raw power draw of an example load before and after applying the filter is shown in Figures 1a and 1b, respectively.

$$RD = \frac{P_{i+1} - P_i}{(P_{i+1} + P_i)/2} \quad (1)$$

The formed vectors are used as input to the IMS algorithm and the resulting three local scores are used for post processing. Each local score corresponds to the distance, expressed as a percentage, from an input vector parameter to the closest cluster of the healthy training data. A simple filter is applied to remove occasional misleading spikes in the local score that do not correspond to legitimate errors.

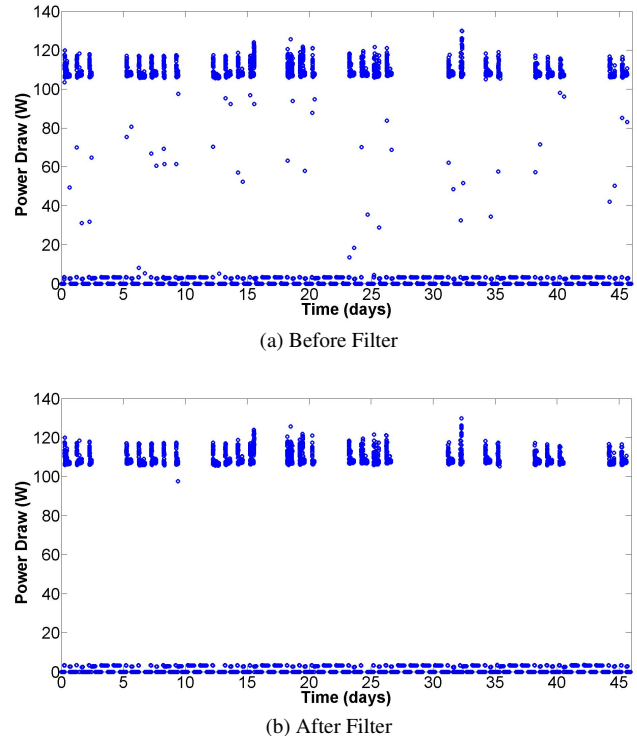


Figure 1. Results of Power Draw Filter

An error is indicated when there are more than 5 consecutive minutes in which the resulting changed power load local score (corresponding to first parameter of input vector) is above 3%, time-to-standby local score (second parameter) above 10%, or rules local score (third parameter) is above 5%. Thresholds for individual IMS local score parameters were obtained by observing the noise fluctuations in the three parameters. Adjustments were made manually until the thresholds were at a level where the IMS program reliably filtered out noise while still detecting anomalies.

These thresholds will likely have to be adjusted for the Sustainability Base deployment. Monitoring the plug loads as described above allows the appropriate personnel to be notified of errors so that they may be corrected, thereby preventing power waste, optimizing plug load management system performance, and possibly extending the life of the devices in question.

In the case where the day's local scores are all below 1% (i.e., the system never deviated more than 1% from nominal behavior) the day's behavior is considered healthy and is added to the training vector for processing the next day of data. This allows the program to better define healthy behavior as time goes on, and prevents the system from picking up deviations in power draw that come from gradual normal system behavior changes.



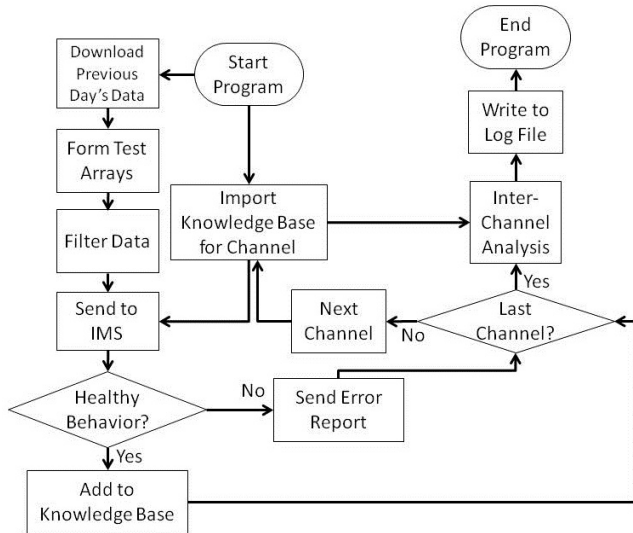


Figure 2. Sustainability Base IMS Program Flow

A separate IMS build may be used to monitor other Sustainability Base subsystems. For example, sensor measurements from the lighting, ground source heat pump, underfloor air distribution, and photovoltaic systems can be used to create additional knowledge bases. The IMS capability will be expanded to incorporate additional systems as they are introduced to the building and as the needs of Sustainability Base change.

### 2.2. Automated Sustainability Base Plug Load Monitoring System

An automated IMS system analyzing plug load system health (see Figure 2) will be implemented by running the Perl program every morning on a server to analyze the past day's data. It will then find unhealthy behavior and notify building personnel by email as necessary. The program is modular, allowing for future additions as new needs arise.

The first task that the program executes is downloading the previous day's data from the cloud server using the system API. Each channel's plug load information is imported and parsed into the test vectors.

The stored knowledge base of nominal behavior is read from files for each channel. The test vectors and nominal clusters are fed into IMS. An additional test is done to find cases of missing data using the timestamps included in the raw data.

If all local scores are below a certain threshold the vectors are appended to the training vectors for use in future days' analyses. This allows the system to learn so that it may better characterize nominal behavior, thereby both reducing false positives and more accurately recognizing errors. Additionally, allowing the individual to mark the identified false positives,

---

### Plug Load Monitoring Weekly Report

---

November 8, 2011

#### ERRORS

##### High Priority:

- Nodes 1-5: Continued communication error 11/6/11 - 11/13/11
- Workstation Rm 288/Ch3.0, Desktop Computer: Sustained unusual behavior from 11:36 11/8/11 - 15:22 11/8/11

##### Low Priority:

- Copy Rm 287/Ch5.0, Shared Copier: Failure to reach standby mode from 06:00 11/7/11 - 22:00 11/7/11

#### STATISTICS

Total Energy Use: 3243 kWh  
 Last Week's Energy Use: 3254 kWh  
 Energy Difference: 11 kWh

Report generated automatically at 02:33 on November 8, 2011

---

Figure 3. Sample Weekly Report

and having the system then add the marked data to the training vectors could lead to a greater reduction in the frequency of false positives.

The resulting local score vectors are processed using the methods described in Section 2.1. Errors are then separated into three categories based on priority of notification. For high priority errors, such as prolonged communication errors or drastically changed loads, a notification is emailed immediately to the system administrators so that the issue may be resolved. Medium priority errors, such as rule failures, are saved as part of a weekly report emailed to the system administrators. Low priority errors, such as short time-to-standby delay or short-lived communication errors, are saved in a log file located on the server.

A sample weekly report generated using pilot study data is shown in Figure 3. Such a report tells the contact what errors are occurring in the system, where they are occurring, and when they occurred. Errors are sorted by seriousness of the anomaly. The report also includes some statistics on power use and difference from the previous week so that the user may better understand their environmental impact.

Note that we deliberately chose a weekly report that showed several alert types (communication failure, failure to reach standby mode, etc.). Consequently, the energy difference in Figure 3 does not represent the energy savings from a week that employed rules versus a week that did not. In fact, in

this report the week of November 8 and the previous week had no rules in place, hence the similar energy consumption. Nor does it represent the difference in energy consumption between a week in which the identified anomalies were remediated and a week where the anomalies were not remediated. Assessment of remediation of identified anomalies was beyond the scope of this study. As the system is deployed in Sustainability Base, each identified anomaly and potential remediation will be individually assessed so as to not adversely affect system operations in the event that the identified anomaly is insignificant.

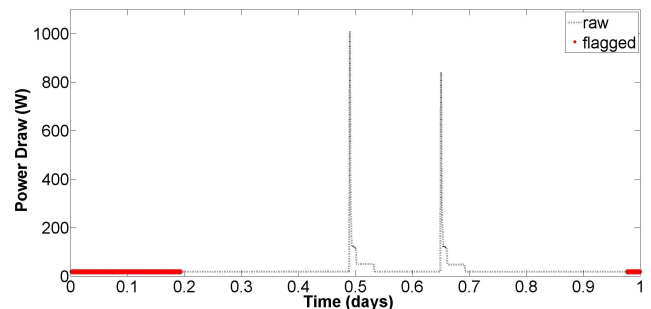
This program was found not to require a large amount of time or processing ability. However, significant data storage is necessary to cache plug load data, training vectors, and program logs. Each channel requires on average 126 kB of storage each day for the plug load data at one minute resolution. System cost and impact can be reduced by employing a multi-purpose server, running additional programs for other functions, rather than a dedicated server.

### 3. RESULTS

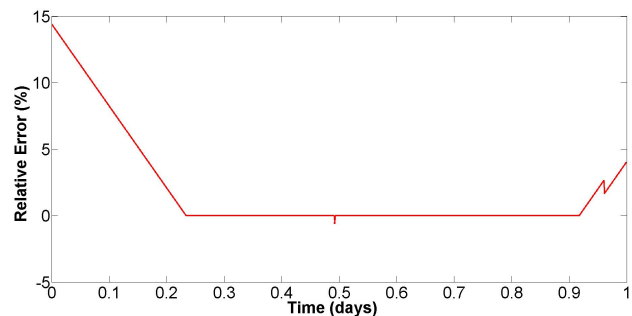
Findings of applying IMS to pilot study plug loads are discussed in this section. During the pilot study examples of five types of errors were found: changed loads, time-to-standby errors, rule failures, inter-channel anomalies, and missing data. For missing data it was not necessary to use IMS so it was analyzed using a simple anomaly detection scheme.

These errors are discovered without defining a priori what types of errors to expect or any model attributes such as rule times, time to transition to standby mode, or normal power draw. The only information that is needed is a sample of healthy data and a range of idle mode power draw for each channel, as applicable. The IMS program is then able to find unusual behavior by comparing new data to the knowledge base learned from the sample data. IMS is capable of discovering errors that have never been seen before in the system. Additional investigation of such cases can determine whether the new behavior is detrimental or insignificant to system operation.

Examples of schedule-based rule failures, time-to-standby errors, changed loads and inter-channel anomalies are provided in the following subsections. In each case the raw data are plotted together with an indication of the points that have been flagged by IMS as being abnormal. The dominant IMS local score output is also presented in a corresponding plot as percentage relative error, which is a normalized distance from the relevant test vector parameter to the nearest cluster in the knowledge base.



(a) Power Draw



(b) On Time (Third Vector Parameter) Local Score

Figure 4. Case of Rule Failure

#### 3.1. Schedule-Based Rule Failures

The first error revealed during the plug load management trial was the occasional failure for schedule-based rules to go into effect. This is likely because of loss of communication or a malfunctioning channel.

An example of this was observed with a printer. Schedule-based rules were set up to turn off the printer between the hours of 10PM and 6AM to conserve power, but failed to go into effect because of communication issues. Figure 4a shows the power draw of the printer. The black dotted line is the raw power draw for the printer and the red points have been flagged by IMS for schedule-based rule failures; note that because of the threshold applied to the IMS local score output, not all points from 10PM (0.92) to 6AM (0.25) are highlighted. The local score over the same time period in Figure 4b increases the longer it has been since the device was supposed to be powered down.

The Sustainability Base IMS found when rule failures occurred for all cases tested. Fixing this type of error would ensure that the plug load management system eliminates power consumption during non-business hours, when the device should be turned off, and maintain the system effectiveness.

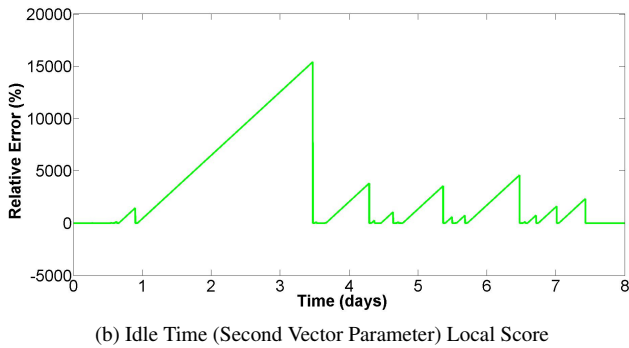
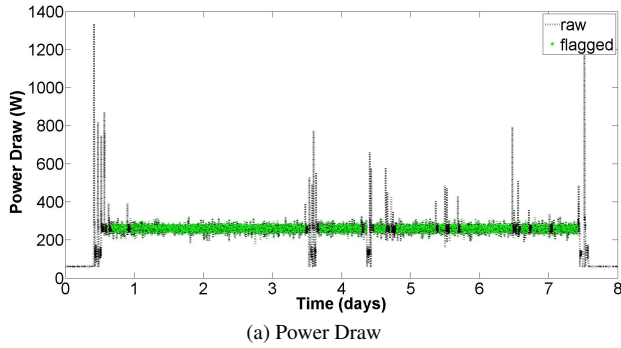


Figure 5. Case of Time-to-Standby Error

### 3.2. Time-to-Standby Errors

Time-to-standby errors were observed in a couple of cases. They can occur as a result of device malfunctions and often require equipment maintenance or possibly replacement.

The power draw for a malfunctioning copier from the pilot study can be seen in Figure 5a. The black dotted line is the raw power draw for the copier and the green points correspond to instances where IMS has found time-to-standby errors. Note that the device fails to enter low-power mode for several days. These cases correspond to an increased local score during the same time periods as seen in Figure 5b.

IMS was able to reliably pick out when time-to-standby errors occurred in all cases tested. Using built-in low power functionality was found in the pilot study to be one of the most effective methods of reducing power draw. Therefore, it is important to identify cases of this anomaly and correct them.

### 3.3. Changed Loads

During the pilot study there were occasions when occupants changed the devices connected to the power strip. They did this in order to replace or change the location of their devices. This can result in data loss or damage if a computer with particular shutdown procedures is plugged into a controlled outlet instead of an uncontrolled outlet.

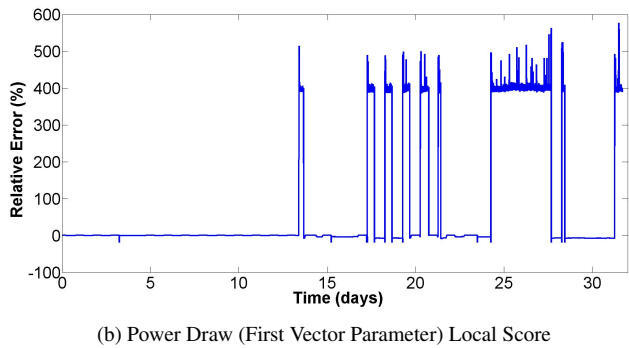
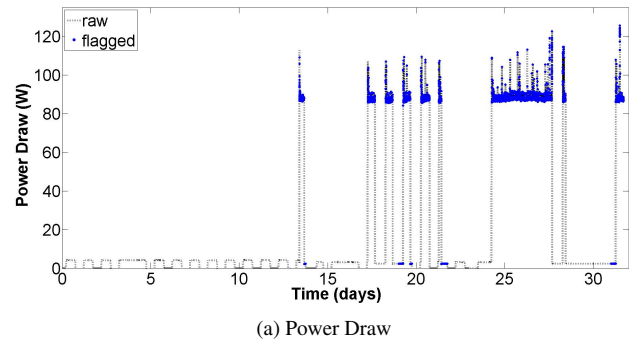


Figure 6. Case of Changed Load

IMS revealed several cases where a device was swapped with another device. The power draw for one such channel can be seen in Figure 6a. This channel originally had a set of speakers (days 1-13), but they were replaced with a computer (days 13-31). The data points that IMS has picked to be the changed load have been marked with blue points. Figure 6b is the local score of that channel during the time period; the output prior to day 13 is near zero, meaning that IMS had seen similar data before.

IMS was able to pick out that the active mode of the computer was a changed load, but the phantom load of the computer was too close to the active mode of the speakers and therefore was not detected. Similarly, the only case that IMS was unable to detect was when a computer was replaced with speakers. The active power draw of the speakers matched the phantom load of the computer previously on that channel. This points to the need of active configuration management. It is likely that additional derived quantities could be used to find the changed load in this case. This is discussed further in the Conclusion Section.

### 3.4. Inter-Channel Anomalies

We also did a preliminary investigation of detecting inter-channel anomalies. These anomalies could indicate that a device is malfunctioning or that load-sensing control rules have failed to go into effect.

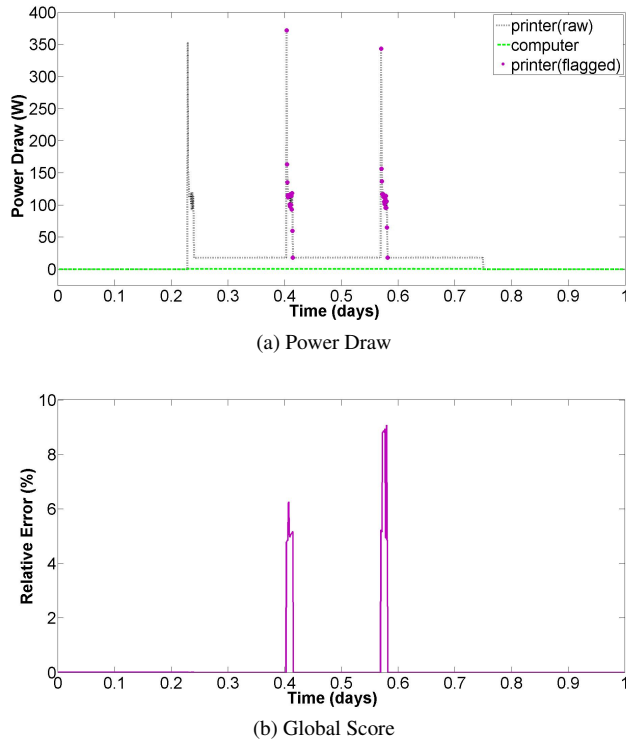


Figure 7. Case of Inter-Channel Anomaly

An example of this can be seen in Figure 7a, which shows the power draw of a printer. In this example the printer (the black dotted line) has three large spikes in its power draw. The first spike occurs when the device is re-energized, while the second and third are from print jobs. In this case the printer was receiving print jobs while the associated computer (the green dashed line) was off. The IMS algorithm saw this as unusual and flagged the points noted in magenta.

Figure 7b shows the IMS global score during that same time period. Note that IMS was able to distinguish between the typical start-up load and the anomalous print jobs. The time stamp in the third parameter of the IMS vector allowed IMS to distinguish between the start-up power spike, which occurred at the same time every day, and potentially anomalous behavior.

Note that the data for these channels considered individually are normal, it is the correlation between the channels that has changed relative to the knowledge base. Detecting these inter-channel errors can uncover abnormal device usage or behavior. For the example shown here the behavior actually reflects the fact that this was a shared printer which received a print job from another computer, but it was presented to IMS in such a way as to make it a test case for inter-channel anomalies.

#### 4. CONCLUSION

These results from the pilot study have proven the effectiveness of IMS for plug load health monitoring. Sustainability Base IMS successfully detected schedule-based rule, time-to-standby, changed load and inter-channel errors in the system. Such a system is expected to be an effective aid in preventing energy waste, improving plug load management system effectiveness, and avoiding system damage. The IMS is currently being deployed to Sustainability Base. Some fine tuning is expected to strike the right balance between flagging irrelevant issues and missing relevant ones.

The Sustainability Base IMS system will provide support to facility managers and occupants to identify usage anomalies. For this study it was not directly tied into the plug load management system, which employed only schedule-based rules to change the on/off state of the channels. The commercial plug load management system was not able to identify the anomalies noted by the IMS system and so was unaware of operational faults that negatively impact energy usage and system usability.

The results from the pilot study (Poll & Teubert, 2012) show that proper setup of device power management settings lead to significant energy savings. The IMS can be employed to find cases of incorrect setup or malfunctioning equipment. Additional analysis would be required to accurately estimate the magnitude of this savings.

Additional research is planned to extend the Sustainability Base IMS to detect other types of errors. The programs were created in a modular fashion to allow for such expansions. These could include additional derived quantities in order to better pick out device changes where the power draw closely matches that of the previous device, or to detect new errors as the needs of the system change.

Another future addition to the Sustainability Base IMS program is the ability to do real time analysis. This will require a fast and reliable method of accessing plug load data. Enabling such a system will increase its effectiveness by notifying system administrators and occupants sooner of potential problems and will allow for some real-time system health statistics to be displayed for Sustainability Base.

#### ACKNOWLEDGMENT

We thank the personnel of the Intelligent Systems Division who participated in the plug load pilot study. We extend our gratitude to Bryan Matthews for his help with the IMS algorithm. Finally, we thank members of the Diagnostics and Prognostics Group in the Intelligent Systems Division for helpful discussions and feedback.

## REFERENCES

- Bradley, P. S., & Fayyad, U. M. (1998). Refining initial points for k-means clustering. In *Proceedings of the fifteenth international conference on machine learning* (pp. 91–99).
- Iverson, D. L. (2004). Inductive system health monitoring. In *Proceedings of the 2004 international conference on artificial intelligence*. CSREA Press.
- Iverson, D. L., Spirkovska, L., & Schwabacher, M. (2010). General purpose data-driven online system health monitoring with applications to space operations. In *Proceedings of the fifty-third annual isa powid symposium*. Research Triangle Park, NC: International Society of Automation.
- Kaneda, D., Jacobson, B., & Rumsey, P. (2010). Plug load reduction: The next big hurdle for net zero energy building design. In *Proceedings of 2010 aceee summer study on energy efficiency in buildings*. Washington, D.C.: American Council for an Energy Efficient Economy.
- Kantardzic, M. (2011). *Data mining: Concepts, models, methods, and algorithms*. Hoboken, New Jersey: John Wiley and Sons, Inc.
- Lobato, C., Pless, S., Sheppy, M., & Torcellini, P. (2011). *Reducing plug and process loads for a large scale, low energy office building: Nrel's research support facility* (Tech. Rep. No. NREL/CP-5500-49002). National Renewable Energy Laboratory.
- Poll, S., & Teubert, C. (2012). Pilot study of a plug load management system: Preparing for sustainability base. In *Proceedings of 2012 ieee green technologies conference*. Institute of Electrical and Electronic Engineers, Inc.

# Health Assessment of Composite Structures in Unconstrained Environments Using Partially Supervised Pattern Recognition Tools

Emmanuel Ramasso<sup>1</sup>, Vincent Placet<sup>2</sup>, Rafael Gouriveau<sup>3</sup>, Lamine Boubakar<sup>4</sup>, and Nouredine Zerhouni<sup>5</sup>

<sup>1,3,5</sup> *Department of Automation and MicroMechatronics Systems*  
*surname.name@ens2m.fr*

<sup>2,4</sup> *Department of Applied Mechanics*  
*surname.name@univ-fcomte.fr*

<sup>1,2,3,4,5</sup> *FEMTO-ST Institute, UMR CNRS 6174 - UFC / ENSMM / UTBM, 25000, Besançon, France*

## ABSTRACT

The health assessment of composite structures from acoustic emission data is generally tackled by the use of clustering techniques. In this paper, the K-means clustering and the newly proposed Partially-Hidden Markov Model (PHMM) are exploited to analyse the data collected during mechanical tests on composite structures. The health assessment considered in this paper is made difficult by working in unconstrained environments. The presence of the noise is illustrated in several examples and is shown to distort strongly the results of clustering. A solution is proposed to filter out the noisy partition provided by the clustering methods. After filtering, the PHMM provides results which appeared closer to the expectations than the K-means. The PHMM offers the possibility to use uncertain and imprecise labels on the possible states, and thus covers supervised and unsupervised learning as special cases which makes it suitable for real applications.

## 1. INTRODUCTION

### 1.1. Context and motivation

The use of organic matrix composite materials has seen considerable growth in many industry sectors in the last decade. The very high specific strength and stiffness and the low-weight of carbon fibres composites has catapulted the use of these materials into the aeronautic market. Their use is also increasing in automotive and railway industries. However, challenges remain to predict their durability, their multiple failure modes over long in-service conditions, to assess the remaining lifetime and to detect damages requiring immediate repair in mobile structures. The main issue is to better

understand the damage mechanisms and kinetics.

Typically the observed failure consists of inter-fibre matrix cracking, fibre breakage and a variety of interfacial failure (like fibre-matrix debonding, splitting or inter-ply delamination). These damages are almost always accompanied by releases of heat and stress-wave propagation due to microstructural changes.

Detection and analysis of acoustic emissions are powerful means for identification of damage phenomena and monitoring of their evolution (Huguet, Godin, Gaertner, Salmon, & Villard, 2002; Barr & Benzeggagh, 1994; Huguet, 2002; Momon, Godin, Reynaud, RMili, & Fantozzi, 2012; Momon et al., 2010) and generally, a standard method based on only one or several features of waves is inadequate for composite materials submitted to complex loadings (Momon et al., 2012). Pattern recognition techniques are thus considered to identify distinct types of AE-signals based on a large number of features obtained from recorded wave hits. However, the formation of AE-signal clusters highly depends on:

- The experimental configuration,
- The material,
- The geometry of the specimen,
- The existence of AE-sources not correlated to specimen failure,
- The criterion to identify the particular failure mechanisms from AE-clusters.

Recent works clearly show the assets of supervised techniques to better identify AE signals regardless of test conditions (Momon et al., 2012).

This work deals with the health assessment of tubular composite structures based on data-driven approaches. Such structures are used in many application fields, such as speed

Emmanuel Ramasso et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

rotors, flywheels, pressure vessels, transportation systems and so on. In these applications, many sources related to the operation of the structure can generate an acoustic activity in addition to the material deformation and degradation. The surrounding electric and electronic appliances can also generate spurious signals at level of acquisition cards used with acoustic sensors. Moreover, the stress state in these composite structures is most of the time complex (multiaxial and heterogeneous) due to the combination of in-service loads and environments. These conditions make particularly difficult the prediction of damage occurrence.

### 1.2. Related work on data-driven approaches for damage phenomena identification and monitoring

State of the art data-driven approaches for tackling the problem of identification and monitoring of damage phenomena can be found in (Momon et al., 2012) where two families of pattern recognition approaches are considered:

- Unsupervised approaches, in particular a K-means clustering algorithm.
- Supervised approaches, in particular the K-Nearest Neighbours classifier.

In unsupervised approaches, the data feed a clustering algorithm which finds out the underlying data structure. The term *unsupervised* specifies that no prior is available concerning the potential membership of a datum to a particular class (also called clusters or group). Generally, unsupervised techniques require to tune the number of possible clusters, either empirically or automatically based on a given criterion (Momon et al., 2012). Many unsupervised techniques have been developed, in particular the K-means algorithm used in (Momon et al., 2012) where the optimisation relies on the Euclidean distance which defines circle-shaped clusters. Other algorithms exist, and some are able to manage the number of clusters sequentially while relying on the Mahalanobis distance to fit ellipsoid-shaped clusters with any orientation (Serir, Ramasso, & Zerhouni, 2012).

In supervised techniques, each datum is accompanied by a label which represents the class. Data and labels are then used to train a classifier which roughly consists in finding the frontiers between the classes. Given a testing instance, the classifier infers membership degrees to all possible classes, and the class with the maximum membership can be selected. Many classifiers have been proposed in the literature, in particular the K-Nearest Neighbours (KNN) (Momon et al., 2012) which is a model-free classifier which assigns to the testing instance the most similar datum found in the training dataset.

### 1.3. Problem statement and contribution

Instead of considering a binary situation where one has or does not have labels of damage, we consider a more general case called *partially-supervised modelling* which consists in

assuming that the prior on labels is now encoded with imprecision and uncertainty. The introduction of this soft computing part enables one to cope with variable situations:

- If the labels are precise and certain for the whole dataset, then one retrieves the *supervised* case.
- If the labels are fully imprecise and certain for the whole dataset, then one retrieves the *unsupervised* case.
- If the labels are fully imprecise and certain only for a part of the dataset but precise and certain for the other part, then one is facing a *semi-supervised* case.
- If the labels takes the form of an uncertainty measure such as a probability, a possibility or a belief mass distribution (Klir & Wierman, 1999) over the set of labels, then one is facing a *partially-supervised* case.

Therefore, the partially-supervised case is the most general situation and enlarges the binary situation (supervised or unsupervised) by considering a *soft* case. It paves the way for the development of algorithms able to manage supervised or unsupervised cases.

Partially-supervised learning was considered in several algorithms such as the K-nearest neighbours classifier (Denoeux, 1995), decision trees (Vannoorenberghe & Denoeux, 2002) and mixture models (Vannoorenberghe & Smets, 2005; Come, Oukhellou, Denoeux, & Aknin, 2009; Denoeux, 2011). In these algorithms, the uncertainty measure is represented by belief functions which generalise probabilities and set-membership approaches (Dempster, 1967; Shafer, 1976; Smets & Kennes, 1994; Smets, 1994). For prognostics and health management applications, it was exploited more recently in (Ramasso, Rombaut, & Zerhouni, 2012) where the authors proposed to combine discrete and continuous predictions.

More recently, the procedures of inference and training in Hidden Markov Models (Rabiner, 1989) were extended to the partially-supervised case and the resulting model, called a Partially-Hidden Markov Model (PHMM) (Ramasso, Denoeux, & Zerhouni, 2012), is able to perform clustering and classification. This model appears to be well-suited for data-driven identification and monitoring of damages in composites, in particular because:

**PHMM manages uncertainties on labels:** It is a partially-supervised technique, managing uncertainties using belief functions which is an original approach compared to the literature on composites analysis and health assessment.

**PHMM considers time-dependency of features:** The underlying statistical modelling takes explicitly into account the relationship between consecutive data, in opposite to previous approaches. This specificity, combined with the Evidential Expectation-Maximisation algorithm

(E2M) used to estimate the parameters in the PHMM, makes it practically more robust to initial conditions than usual methods.

**PHMM is flexibility to represent distributions:** The distributions of features conditionally to the damages can take various forms, in particular the mixture of Gaussians which allows to represent ellipsoid-shaped relationships between features with different orientations, in opposite to usual approaches of clustering based on the Euclidean distance.

**PHMM can be used for prognostics:** Usual HMMs were already exploited for prognostics and the extension to PHMM can be straightforward. Note that prognostic is not considered in this paper.

Usual probabilistic HMM were exploited for damage modelling in some recent works, in particular in (Zhou et al., 2008, 2009). To our knowledge, the most related work can be found in (Wang, 2011) where imprecise probabilities are used for composite materials analysis based on HMM. Belief functions are different from imprecise probabilities (Shafer, 1976; Smets, 1994), the only mathematical link holds in the lower probability measure which represents a belief measure. The tools developed in the the theory of belief functions as considered in Shafer and Smets’ mind (Shafer, 1976; Smets & Kennes, 1994) are now well recognized are suitable ones for pattern recognition and are generally developed to alleviate probabilistic assumptions.

The contribution of this paper lies in two main aspects:

- The introduction of PHMM for health assessment of composites (with the aforementioned assets).
- The proposition of a filtering approach adapted for clustering methods.
- The testing and validation of PHMM and filtering results on a real-world case.

A plot chart of the proposed methodology is given in Figure 1.

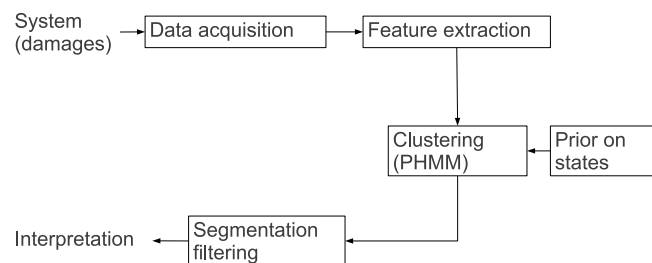


Figure 1. Plot chart of the proposed methodology.

## 2. EXPERIMENTS

### 2.1. The material

Health was assessed on composite split disks when submitted to mechanical loading. The tests were performed according to ASTM D2290 “Apparent hoop tensile strength of plastic or reinforced plastic pipe by split disk method” (Figure 2).

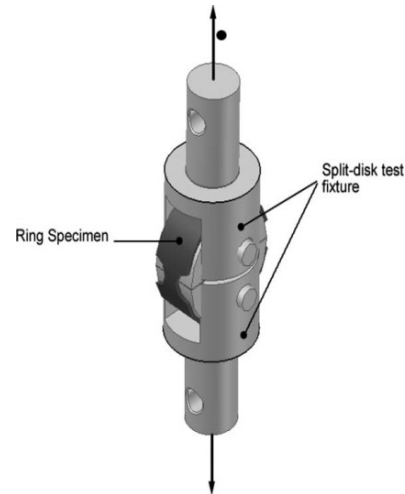


Figure 2. Experimental set-up for tensile test on split-disk specimen.

Rings were produced by cutting and machining filament-wound carbon fibre reinforced epoxy tubular structures intended for the manufacturing of flywheel rotors. Three different lay-up configurations, namely  $[(90^\circ)_6]$ ,  $[(90^\circ)_2 / \pm 20^\circ / (90^\circ)_2]$  and  $[(90^\circ)_2 / \pm 45^\circ / (90^\circ)_2]$  lay-ups were studied.

The transient elastic waves were recorded at the material surface using a multi-channels data acquisition system from EPA (Euro Physical Acoustics) corporation (MISTRAS Group). The system is made up of miniature piezoelectric sensors ( $\mu - 80$ ) with a range of resonance of 250 – 325 kHz, preamplifiers with a gain of 40 dB and a 20 – 1000 kHz filter, a PCI card with a sampling rate of 1 MHz and the AEWin software. The sensors were coupled on the specimen faces using a silicon grease. The calibration of the system was performed after installation of the transducers on the specimen and before each test using a pencil lead break procedure. Ambient noise was filtered using a threshold of 40 dB. The acquisition parameters: PDT (Peak Definition Time) =  $60\mu\text{sec}$ ; HDT (Hit Definition Time) =  $120\mu\text{sec}$  and HLT (Hit Lock Time) =  $300\mu\text{sec}$  were identified using preliminary measurements.



## 2.2. Identification of damage mechanisms and their happening under mechanical loading using Non Destructive Technique (NDT)

The damages of ring specimens caused by the mechanical loading were first identified by cautiously inspecting the inner, outer and width surfaces of the specimen following the mechanical tests. Pictures in Figure 3 and their legends precisely detail the identified damage mechanisms.

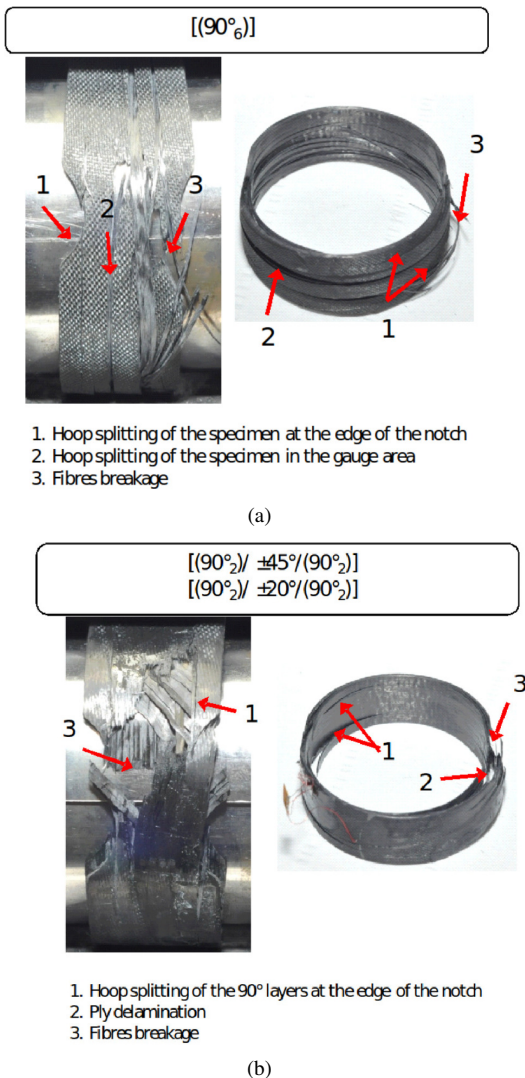


Figure 3. Photograph of the observed macro-damage on split-disk specimen after failure. Evidence of splitting in UD structures lay-up, ply delamination in multidirectional structures and fibre failures.

Some of them could have been induced by the very high level energy release at the specimen failure. To avoid any confusion and misinterpretation, some tests were also stopped just before the specimen failure (at approximately 95% of the ultimate tensile stress), and the composite observed using an op-

tical microscope (Nikon Eclipse LV 150). These microscopic observations allowed the damage mechanisms observed from fractography images to be confirmed (Figure 4).

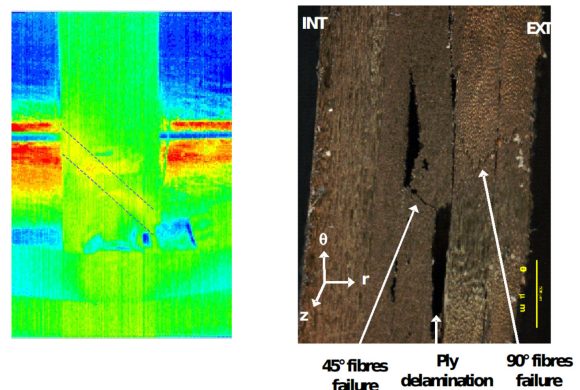


Figure 4. Micrograph of the surface of a  $[(90^\circ)_2/\pm 45^\circ/(90^\circ)_2]$  specimen (thickness side) loaded at 95% of the maximum strength. Evidence of Matrix cracking, ply delamination and fibre breakage.

The occurrence of the damages according to the stress level were identified for each lay-up configuration using a combination of information obtained from NDT techniques, i.e. AE, IR (Infrared) thermography. A CCD Kodak Megaplug 4.2 camera was also used to record specimen pictures with a frame rate of 1 frame per second. Infrared thermography provided high resolution thermal maps as a function of loading time and allowed the damages, characterized by a heat release on the surface of the specimen, to be located (Figure 5).

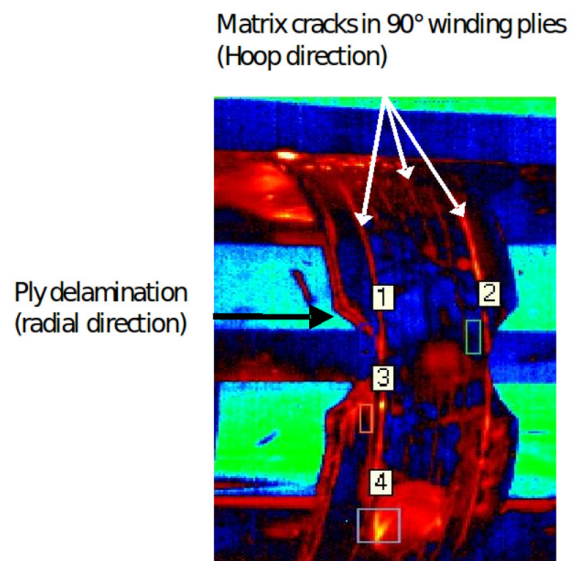


Figure 5. Infrared map during split-disk tests (specimen  $[(90^\circ)_2/\pm 45^\circ/(90^\circ)_2]$ ). Evidence of matrix cracks (splitting) and radial delamination.

The used apparatus was a Mid-Wavelength InfraRed camera

with 3 – 5 $\mu$ m spectral response (MWIR3 JADE from CEDIP Company). It has a focal plane array detector with 320 × 256 (InSb) and provides high sensitivity of less than 20mK at 300K at high speed frame rate of 150 Hz. Altair software was used to control the camera and acquire and process the infrared images. A calibration was done on the specimens in order to be able to determine absolute temperatures from the radiative measurements. It takes into account the emissivity of the material at ambient temperature and the radiation conditions of the environment. Figure 6 provides an example of the signals recorded on a [(90°)<sub>2</sub>/ ± 45°/(90°)<sub>2</sub>] specimen.

Since approximately 550MPa (20 seconds), IR images clearly show the emergence of matrix cracks, growing in the hoop direction up to the outer 90° layers splitting. A good correlation is observed between acoustic activity and temperature increase. Each crack propagation can easily be located on the outer surface of the specimen using IR images. A temperature increase of several degrees goes with the crack propagation in this sample area. A significant increase in the absolute energy from AE of a magnitude order of 109 aJ is recorded. Since approximately 850MPa (52 seconds), a damage increase, traduced by a quick increase in temperature and in absolute energy (of about 1010aJ order) was recorded. By means of macrography images, this damage was identified to be due to bundles of fibres fracture.

The evolution of the increase in the acoustic activity and temperature at the outer surface of the specimen and the optical images allowed the different damage modes to be identified during tensile loading for each specimen. Tables 1, 2, 3 synthesised the results for each specimen family.

In the first table (Tab. 1), the damage modes and onsets as a function of mechanical stress of the UD specimen are listed. In addition to fibres breakage just before the final failure, macrocracks with specimen hoop splitting occur since 450 MPa.

The second table (Tab. 2) and the last table (Tab. 3) show that for the others stacking sequences, in addition to the hoop splitting of UD layers, inter-laminar ply delamination is detected. Experimental data do not provide enough information to clearly distinguish the onsets of each variety of interfacial failure. Ely and Hill (Ely & Hill, 1995) showed that the signal amplitude can be used as a filter criteria to distinguish fibre breakage and longitudinal splitting. Unfortunately, analysis based on only one or several of the most used parameters, such as amplitude, energy and duration as a filter criteria do not allow hoop splitting and ply delamination to be discriminated.

### 2.3. Identification of damage mechanisms based on acoustic emission data using pattern recognition tools

The following tests are focused on the configuration [(90°)<sub>6</sub>].

[(90°) <sub>6</sub> ]		
Event	Loading onset (MPa)	Detected damage mode
1	450 – 480	Specimen hoop splitting
3	1300 – 1610	Fibres breakage
4	1340 – 1660	Final failure

Table 1. Damage onsets and modes as a function of lay-up configuration [(90°)<sub>6</sub>] determined during quasi-static SD tests using NDT.

[(90°) <sub>2</sub> / ± 45°/(90°) <sub>2</sub> ]		
Event	Loading onset (MPa)	Detected damage mode
1	470 – 550	Hood splitting in 90° layers
2	?	Ply delamination
3	1230 – 1310	Fibres breakage
4	1250 – 1380	Final failure

Table 2. Damage onsets and modes as a function of lay-up configuration [(90°)<sub>2</sub>/ ± 45°/(90°)<sub>2</sub>] determined during quasi-static SD tests using NDT.

#### 2.3.1. Features extraction

Acoustic emission signals obtained from the acquisition board can be transformed into features which provides relevant information about the wear mechanisms.

Several features can be extracted and we refer to (Hadzor et al., 2011) for a detail overview. Figure 7 (taken from (Huang et al., 1998)) summarized the most relevant ones. There are about 22 well-known features and the most used ones are in particular: amplitude (in decibels), duration, rise time, strength, energy, counts, counts to peak, and some frequency-based features.

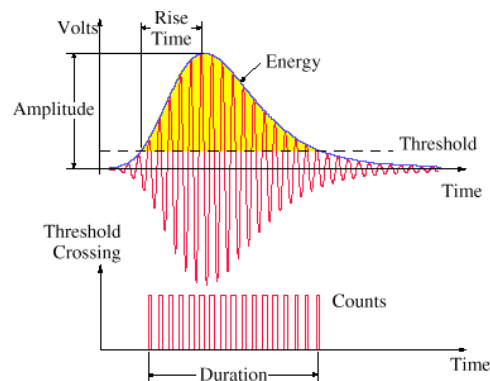


Figure 7. Features of typical AE signal.

The principal components of features were studied via a PCA with the aim of reducing the number of features by selecting the most relevant ones. Using the PCA results, 7 features were selected, accounting for 95% of the variance. An example of PCA is given in Figure 8. As underlined in several papers on AE-based materials analysis, this figure depicts that frequency-based features as well as signal energy, duration and amplitude brings most of the variance and appear as rel-

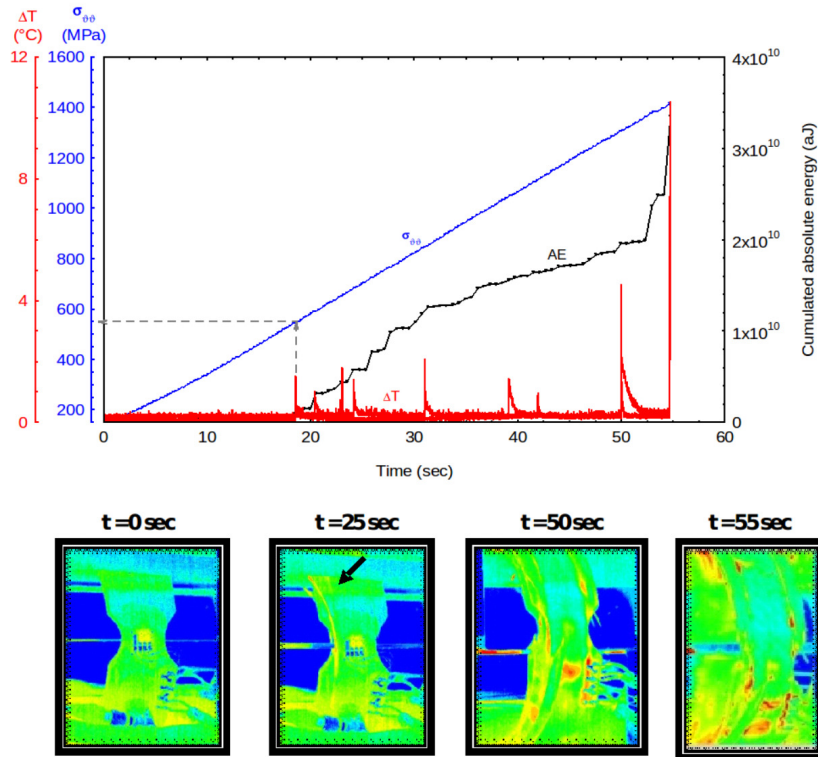


Figure 6. Evolution of the damage as a function of time during tensile tests on rings. Lays-up:  $[(90^\circ)_2 / \pm 45^\circ / (90^\circ)_2]$ . Hoop stress, acoustic activity and temperature increase vs. time and infrared maps at different times.

[(90°) <sub>2</sub> / ± 20° / (90°) <sub>2</sub> ]		
Event	Loading onset (MPa)	Detected damage mode
1	480 – 520	Hoop splitting in 90° layers
2	?	Ply delamination
3	850 – 1020	Fibres breakage
4	900 – 1120	Final failure

Table 3. Damage onsets and modes as a function for lay-up configuration  $[(90^\circ)_2 / \pm 20^\circ / (90^\circ)_2]$  determined during quasi-static SD tests using NDT.

evant features for composites analysis.

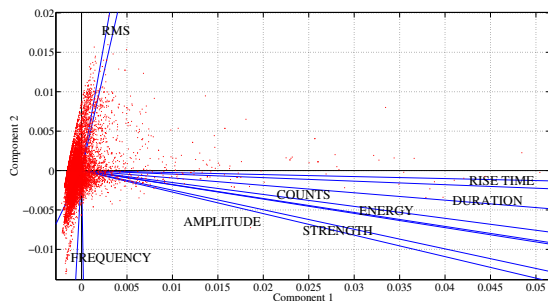


Figure 8. The principal components.

### 2.3.2. Noise removal

**Need of noise removal** When the pre-processing of data is completed (features extracted followed by a PCA), clustering algorithms can be applied in order to identify families of degrading modes. However, due to the unconstrained environment, a lot of waveforms can be registered, among which only a few represent damage mechanisms in the composite structure.

To illustrate the amount of waveforms which can be considered as false events, we simply applied the K-means clustering algorithm with  $K = 10$ . To cope with the dependency on the initial conditions of the K-means clustering, 200 iterations and 10 different initialisations were considered and the partition minimizing the average within-cluster distance was chosen.

The evolution of the logarithm of the cumulated sum of cluster appearance ( $\log\text{CSCA}$ ) along time is represented in Figure 9(a) for a configuration  $[(90^\circ)_6]$ . In this configuration, three main families of damage is generally encountered, but it is expected that each family can be represented by members with particular properties, accounting for the choice 10 clusters. This number also well illustrates the influence of noise as described hereafter.

The  $\log\text{CSCA}$  is defined as follows. Let  $\Omega =$

$\{1, 2, \dots, k, \dots, K\}$  be the set of clusters, the following matrix:

$$S(t, k) = 1 \quad \text{if cluster } k \text{ is activated at sample } t \quad (1)$$

where  $t$  is the sample index, represents the cluster appearance. The logCSCA can be computed as:

$$\log \text{CSCA}(t, k) = \log_{10} \sum_{x=1}^t S(x, k) \quad (2)$$

The result is depicted in Figure 9(a). This result is far from one can expect. Actually, the number of clusters located at the beginning of the experiment can not reflect a damage mechanism because this area corresponds only to the seating of the specimen in the grips. The initial portion of the stress-time curve is clearly non-linear. The material is not significantly stressed, the applied load at this point just allows the grips to be straightened. Moreover, only a few phenomenons are detected after the half of the duration of the experiment which is also unexpected since this area should be one of the most active ones.

A simple noise removal procedure enables one to obtain a much more satisfying result as depicted in Figure 9(b). The procedure is detailed below.

**Noise removal procedure** Since it appears difficult to automatically identify a noise waveform from other types of waveforms directly from the features, the noise removal procedure is applied after the clustering phase, i.e. in the clusters space. The noise is removed by a simple majority voting scheme in a given window with size  $W$ . The basic assumption is to consider that a mechanical phenomenon implies several consecutive waveforms with similar properties. Let  $S_f$  be the filtered matrix of cluster appearance:

$$S_f(t, k^*) = 1 \quad (3)$$

where

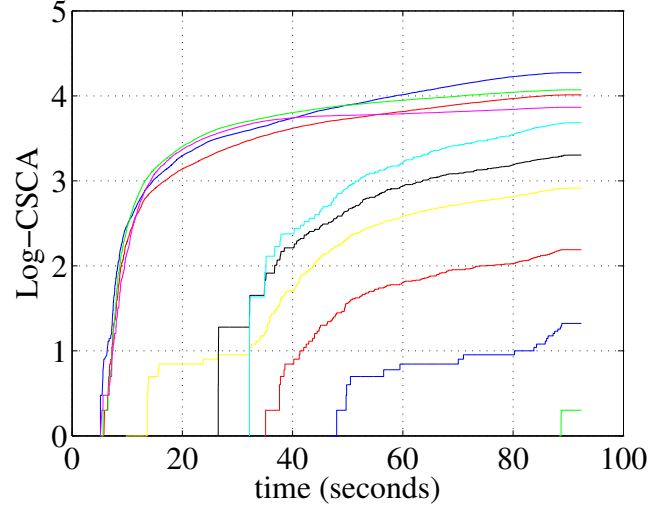
$$k^* = \underset{j=1}{\text{argmax}}^K C_j \quad (4)$$

and

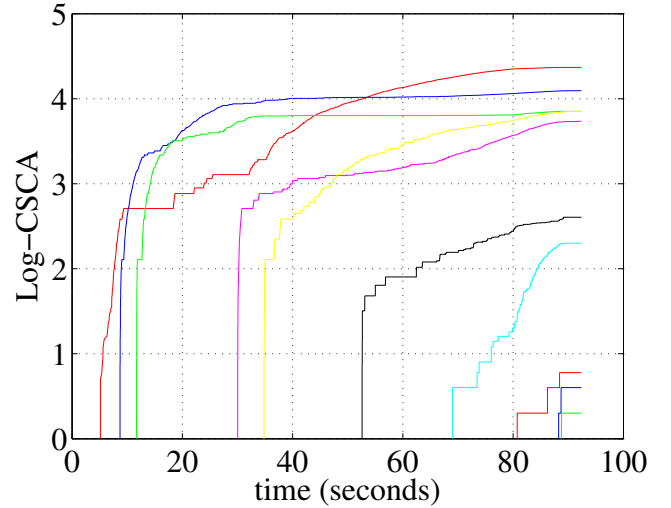
$$C_j = \sum_{i=t-\frac{W}{2}}^{t+\frac{W}{2}} S(i, k) \quad (5)$$

To speed up the processing, the sliding window of size  $W$  can be moved by step equals to  $P = W/2$ . In this case  $S_f(t', k^*) = 1, \forall t' \in [t, t + P]$ .

**Setting up the window size** At the beginning of an experiment, most of waveforms come from the straightening of the grips. This phase generates a lot of waveforms (about 20%) with small energy and small amplitude. Therefore, in this



(a) K-means: Before filtering.



(b) K-means: After filtering.

Figure 9. Evolution of the logarithm of the cumulated sum of cluster appearance along time using the K-means with 10 clusters.

phase, the window size can be large (here chosen equal to 256 points). Then, the specimen start to be notably loaded which induces more complex mechanical phenomenons generating waveforms with particular properties until the final failure. We assume that the damages appearing along time have an increasing importance as the load increases. Therefore, in order to not filter out these potentially important phenomenons, the window size decreases along time. The decreasing rate of the window size is, for the moment, set manually, as depicted in Figure 10.

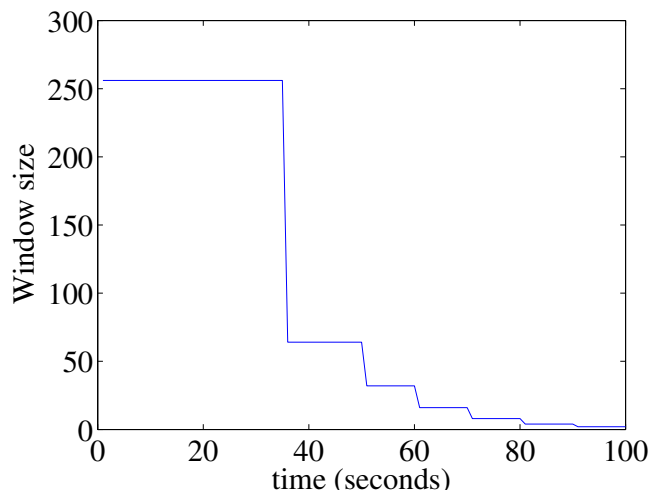


Figure 10. Evolution of the size of the window used in the noise removal procedure. The time axis represents the percentage of experiments' duration.

This noise removal procedure is illustrated in Figures 9(a)-9(b) for the K-means algorithm.

### 2.3.3. Clustering results

**Test on  $[(90^\circ)_6]$  with 10 states** For comparison purpose (based on the aforementioned K-means results), the Partially-Hidden Markov Model (PHMM) was tuned with 10 states, 1 component for each state, using mixture of Gaussians to represent the distribution of features conditionally to the states. 10 different initialisations were used and the PHMM with the highest likelihood was selected. On the selected model, the Viterbi decoder was applied to obtain the sequence of states along time. The state sequence is then processed by the noise removal procedure. In the PHMM, two sets of partial labels were used:

- A set of labels in the 3 first seconds of the experiment, which correspond to the seating of the specimen in the grips and to the straightening of the grips.
- A set of labels in the 3 last seconds of the experiment, which correspond to the fibre breakage.

For these two areas, the plausibility of the two states was set to 1. Since other mechanical phenomenons can appear

in these areas, a random noise drawn uniformly in  $[0, 1]$  with  $\sigma = 0.1$  was added on the labels, followed by a normalisation of the plausibilities which must be in  $[0, 1]$ .

Figures 11(a)-11(b) pictorially depict the clustering result before and after noise removal.

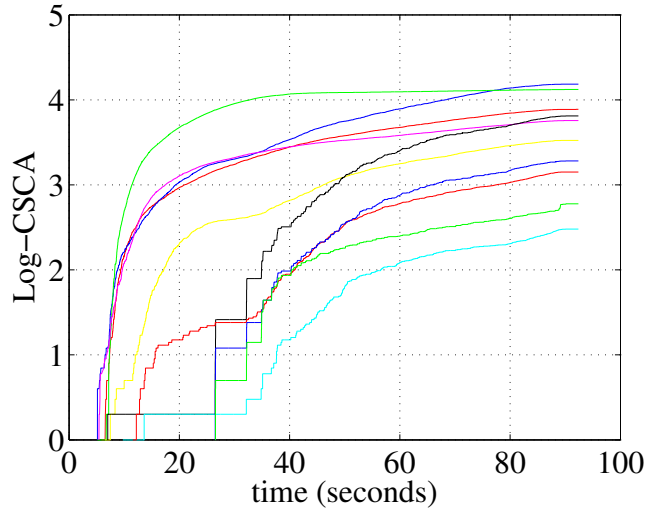
The result provided by the (post-processed) PHMM appears to be different from the K-means ones. The cluster of noise, which starts around  $t = 5$  for PHMM, is well isolated by this method. This result is much more satisfying than the one obtained by the K-means which is more affected by the noise since several clusters starts very soon while there is almost no constraint. Compared to the K-means, the cluster of noise detected by the PHMM also depicts a stationary regime (around  $t = 40$ ) which is attributed to the end of the seating of the specimen in the grips.

Moreover, the K-means seems to provide clusters with quite periodical starting points (the period seems to be close to 35 seconds on Figure 9(b)) which does not reflect the real behavior of the material. In comparison, the PHMM emphasizes that in some intervals of time, several damages appear, which seems more credible. For example, in the last 30%, several clusters have been identified, which is coherent with the fact that the material is approaching the fibres breakage and the final failure.

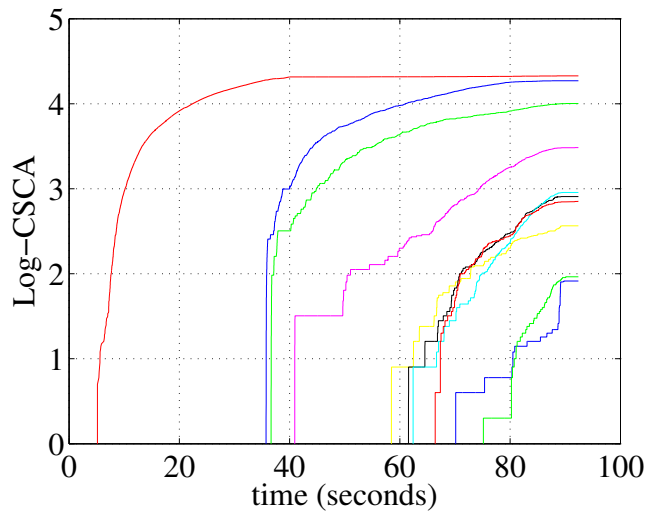
The results provided by the PHMM are also closer to the expectations detailed in Table 1 since three main families appear: around  $t = 5$ ,  $t = 30$  and for  $t > 60$ . The following tests emphasize this conclusion.

**Test  $[(90^\circ)_6]$  with 3 states** The PHMM was run on the configuration  $[(90^\circ)_6]$  with 3 states and 2 components per state and using a similar labeling process as proposed in the previous test. The number of states corresponds here to the number of expected families described in Table 1 and the number of components to different possible members to these families. Figure 12 represents the logCSDA criterion along time for this set-up. The noise removal procedure again improves the detection by separating the clusters. The first cluster still presents the stationary regime around  $t = 40$  and could also include the signature of the matrix microcracking. The second cluster may represent the hoop splitting with propagation of macrocracks, while the third may characterize the fibre breakage which, as expected, seems to generate some "jumps" in the evolution of the logCSDA.

Figure 13 also presents the results of clustering jointly with report to the mechanical stress and the cumulated energy. The positioning of the clusters is here well justified: the first cluster gathers the waveforms corresponding to the seating of the specimen in the grips where the stress is minimal, the second to possible splittings and the third to fibre breakage.

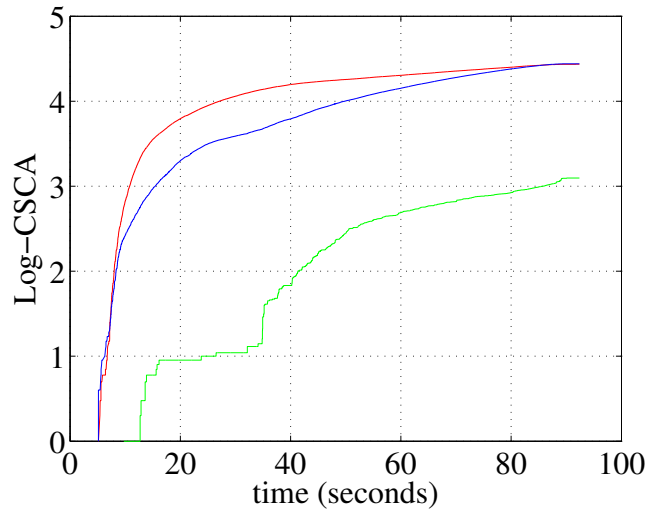


(a) PHMM: Before filtering.

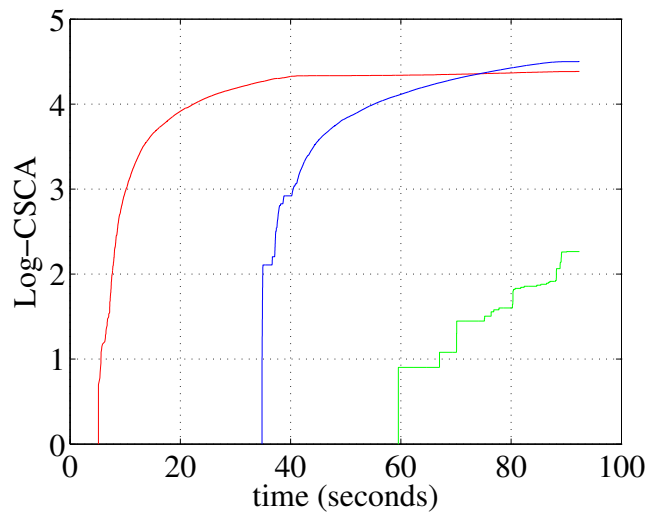


(b) PHMM: After filtering.

Figure 11. Evolution of the logarithm of the cumulated sum of cluster appearance along time using the PHMM with 10 clusters and 1 component.



(a) PHMM: Before filtering.



(b) PHMM: After filtering.

Figure 12. Evolution of the logarithm of the cumulated sum of cluster appearance along time using the PHMM with 3 clusters and 2 components.

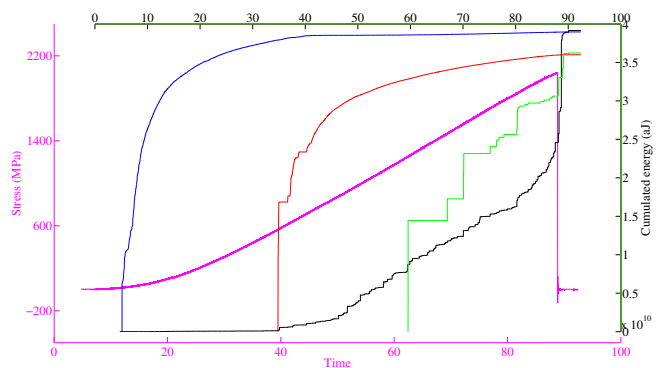


Figure 13. Viewing the results presented in Figure 12(b) jointly with the mechanical stress and the cumulated energy.

### 3. CONCLUSION

In this paper, we considered the problem of health assessment of composite structures in applications where the noise is present and can strongly influence the results. Considering the noise in composite structure analysis is a difficult task due to the number of data to process and to the lack of knowledge about the damage appearance and evolution. To cope with this problem, a filtering procedure working in the space of the clusters (which can be estimated by any clustering approach) is developed and applied on the K-means and on the newly proposed Partially-Hidden Markov Model (PHMM). The results of the latter are shown to be closer to the expectations than the former.

As perspectives, several tracks are currently tackled:

- How to automatically set-up the window's size in the cluster filtering process? Various experiments let us think that one solution is to consider the level of energy released during the experiment. To make it more reliable, a combination with other features should be used.
- How to obtain a better description of some particular damages and a better discrimination of the mechanisms of a same damage family, such as ply delamination and splitting, in order to improve the interpretation of the damage process? The labelling process allowed in PHMM can be exploited to emphasize some kind of "macro-states" and a hierarchical processing of these macro-states could give some useful information about finer degradations.
- On the opposite to the previous item, given a large number of clusters representing micro-states, how to gather them into families of damages? Partially supervised learning in PHMM can be exploited to assign uncertain and imprecise prior on some waveforms about their membership to some predefined macro-states. The PHMM could then be applied to estimate if "components" have to be gathered (Serir, Ramasso, & Zerhouni, 2011).
- How to exploit the model-based approach (Rabiei, Modarres, & Hoffman, 2011) jointly with data-driven ones using possibly distributed sensors (Daigle, Bregon, & Roychoudhury, 2011) for a better health assessment and prognostics (Kessler, Flynn, Dunn, & Todd, 2011)? Many information fusion tools were developed in the literature (Ramasso & Jullien, 2011; Ramasso, Rombaut, & Zerhouni, 2012) and experiments are now necessary for a validation on composite structure analysis presented in this paper.

### REFERENCES

- Barr, S., & Benzeggagh, M. (1994). On the use of acoustic emission to investigate damage mechanisms in glass-

fibre-reinforced polypropylene. *Composite Science Technology*, 52, 369-376.

- Come, E., Oukhellou, L., Denoeux, T., & Akin, P. (2009). Learning from partially supervised data using mixture models and belief functions. *Pattern Recognition*, 42(3), 334-348.
- Daigle, M., Bregon, A., & Roychoudhury, I. (2011). Distributed damage estimation for prognostics based on structural model decomposition. In *Annual conference of the prognostics and health management society* (Vol. 2).
- Dempster, A. (1967). Upper and lower probabilities induced by multiple valued mappings. *Annals of Mathematical Statistics*, 38, 325-339.
- Denoeux, T. (1995). A k-nearest neighbor classification rule based on Dempster-Shafer theory. *IEEE Trans. on Systems, Man and Cybernetics*, 5, 804-813.
- Denoeux, T. (2011). Maximum likelihood estimation from uncertain data in the belief function framework. *IEEE Transactions on Knowledge and Data Engineering*.
- Ely, T., & Hill, E. (1995). Longitudinal splitting and fiber breakage characterization in graphite epoxy using acoustic emission data. *Mater. Eval.*, 53, 369-376.
- Hadzor, T. J., Barnes, R. W., Ziehl, P. H., Xu, J., & Schindler, A. K. (2011, June). *Development of acoustic emission evaluation method for repaired prestressed concrete bridge girders* (Tech. Rep. No. FHWA/ALDOT 930-601-1). 238 Harbert Engineering Center, Auburn, AL 36849: Auburn Highway Research Center, Department of Civil Engineering.
- Huang, M., Jiang, L., Liaw, P., Brooks, C., Seeley, R., & Klarstrom, D. (1998). Using acoustic emission in fatigue and fracture materials research. *Journal of Materials*, 50(11), 1-12. (The Minerals, Metals & Materials Society (TMS))
- Huguet, S. (2002). *Application de classificateurs aux données d'émission acoustique: identification de la signature acoustique des mécanismes d'endommagement dans les composites à matrice polymère*. Unpublished doctoral dissertation, Institut national des sciences appliquées (Lyon), Groupe d'Études de Métallurgie Physique et de Physique des Matériaux. (in French)
- Huguet, S., Godin, N., Gaertner, R., Salmon, L., & Villard, D. (2002). Use of acoustic emission to identify damage modes in glass fibre reinforced polyester. *Composite Science Technology*, 62, 1433-1444.
- Kessler, S. S., Flynn, E. B., Dunn, C. T., & Todd, M. D. (2011). A structural health monitoring software tool for optimization, diagnostics and prognostics. In *Annual conference of the prognostics and health management society* (Vol. 2).
- Klir, G., & Wierman, M. (1999). Uncertainty-based information elements of generalized information theory.

- In (chap. Studies in fuzzyness and soft computing). Physica-Verlag.
- Momon, S., Godin, N., Reynaud, P., RMili, M., & Fantozzi, G. (2012). Unsupervised and supervised classification of ae data collected during fatigue test on cmc at high temperature. *Composites Part A: Applied Science and Manufacturing*, 43, 254-260.
- Momon, S., Moevus, M., Godin, N., RMili, M., Reynaud, P., Fantozzi, G., et al. (2010). Acoustic emission and lifetime prediction during static fatigue tests on ceramic-matrix composite at high temperature under air. *Composites Part A: Applied Science and Manufacturing*, 41, 913-918.
- Rabiei, M., Modarres, M., & Hoffman, P. (2011). Structural integrity assessment using in-situ acoustic emission monitoring. In *Annual conference of the prognostics and health management society*.
- Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77, 257-285.
- Ramasso, E., Denoeux, T., & Zerhouni, N. (2012). Partially-Hidden Markov Models. In *International conference on belief functions*. Compiègne, France. (Accepted in February 2012)
- Ramasso, E., & Jullien, S. (2011). Parameter identification in Choquet integral by the Kullback-Leibler divergence on continuous densities with application to classification fusion. In *European society for fuzzy logic and technology* (p. 132-139). Aix-Les-Bains, France.
- Ramasso, E., Rombaut, M., & Zerhouni, N. (2012). Joint prediction of observations and states in time-series based on belief functions. *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*. (Accepted (<http://dx.doi.org/10.1109/TSMCB.2012.2198882>))
- Serir, L., Ramasso, E., & Zerhouni, N. (2011). Time-sliced temporal evidential networks: the case of evidential hmm with application to dynamical system analysis. In *Ieee int. conf. on prognostics and health management* (p. 1-10). Denver, CO, USA.
- Serir, L., Ramasso, E., & Zerhouni, N. (2012). Evidential evolving gustafson-kessel algorithm for online data streams partitioning using belief function theory. *International Journal of Approximate Reasoning*, 5, 747-768.
- Shafer, G. (1976). *A mathematical theory of Evidence*. Princeton University Press, Princeton, NJ.
- Smets, P. (1994). What is Dempster-Shafer's model ? In I. R. Yager, M. Fedrizzi, & J. Kacprzyk (Eds.), *Advances in the dempster-shafer theory of evidence* (p. 5-34). J.Wiley & Sons.
- Smets, P., & Kennes, R. (1994). The Transferable Belief Model. *Artificial Intelligence*, 66(2), 191-234.
- Vannoorenberghe, P., & Denoeux, T. (2002). Handling uncertain labels in multiclass problems using belief decision trees. In *Information processing and management of uncertainty in knowledge-based systems* (p. 1919-1926).
- Vannoorenberghe, P., & Smets, P. (2005). Partially supervised learning by a Credal EM approach. In *Europ. conf. on symbolic and quantitative approaches to reasoning with uncertainty* (Vol. 3571, p. 956-967).
- Wang, Y. (2011). Multiscale uncertainty quantification based on a generalized hidden markov model. *Journal of Mechanical Design*, 133, 031004(1-10).
- Zhou, W., Kovvali, N., Reynolds, W., Papandreou-Suppappola, A., Chattopadhyay, A., & 20, J. . D. Cochran vol., pp. 1271-1288. (2009). On the use of hidden markov modeling and time-frequency features for damage classification in composite structures. *Journal of Intelligent Material Systems and Structures, Special Issue on Information Management in Structural Health Monitoring*, 20, 1271-1288.
- Zhou, W., Reynolds, W., Moncada, A., Kovvali, N., Chattopadhyay, A., Papandreou-Suppappola, A., et al. (2008). Sensor fusion and damage classification in composite materials. In *Proc. spie* (Vol. 69260N).



# Use of COTS Functional Analysis Software as an IVHM Design Tool for Detection and Isolation of UAV Fuel System Faults

Octavian Niculita<sup>1</sup>, Phil Irving<sup>2</sup>, and Ian K Jennions<sup>3</sup>

<sup>1,2,3</sup> *Integrated Vehicle Health Management Centre, Cranfield University, Cranfield, Bedford MK43 0FQ, UK*

[o.niculita@cranfield.ac.uk](mailto:o.niculita@cranfield.ac.uk)

[p.e.irving@cranfield.ac.uk](mailto:p.e.irving@cranfield.ac.uk)

[i.jennions@cranfield.ac.uk](mailto:i.jennions@cranfield.ac.uk)

## ABSTRACT

This paper presents a new approach to the development of health management solutions which can be applied to both new and legacy platforms during the conceptual design phase. The approach involves the qualitative functional modelling of a system in order to perform an Integrated Vehicle Health Management (IVHM) design – the placement of sensors and the diagnostic rules to be used in interrogating their output. The qualitative functional analysis was chosen as a route for early assessment of failures in complex systems. Functional models of system components are required for capturing the available system knowledge used during various stages of system and IVHM design. MADe™ (Maintenance Aware Design environment), a COTS software tool developed by PHM Technology, was used for the health management design. A model has been built incorporating the failure diagrams of five failure modes for five different components of a UAV fuel system. Thus an inherent health management solution for the system and the optimised sensor set solution have been defined. The automatically generated sensor set solution also contains a diagnostic rule set, which was validated on the fuel rig for different operation modes taking into account the predicted fault detection/isolation and ambiguity group coefficients. It was concluded that when using functional modelling, the IVHM design and the actual system design cannot be done in isolation. The functional approach requires permanent input from the system designer and reliability engineers in order to construct a functional model that will qualitatively represent the real system. In other words, the physical insight should not be isolated from the failure phenomena and the diagnostic analysis tools should be able to adequately capture the experience bases. This approach has been verified on a laboratory bench top

test rig which can simulate a range of possible fuel system faults. The rig is fully instrumented in order to allow benchmarking of various sensing solution for fault detection/isolation that were identified using functional analysis.

## 1. INTRODUCTION

Initial research on development of IVHM solutions focused on fault identification during operations and maintenance. This research direction culminated with the development of model-based reasoning software tools capable of comparing the observed behaviour with the expected behaviour of the system in order to identify abnormal conditions and eventually performing run-time repairs. Livingstone and its extension L2, HyDe from NASA, Rodon™ from Combitech (2012), TFPG FACT, ReasonPro™ (2012) from Impact Technologies are just a few COTS and open source model-based reasoners that use system configuration and qualitative or quantitative behavioural models for developing fault detection/isolation procedures. A second research direction in developing IVHM solutions is supported by design analysis. These types of analysis are typically carried out for either investigations regarding system's potential to realize the health management goals – as a system design aid. Many different software tools like eXpress™ from DSI Int. (DSI, 2012), TEAMST™ from Qualtech Systems Inc. (QSI, 2012), ADVISE, Design PHM™ from Impact Technologies have been developed to aid the design of IVHM solutions, to minimize diagnostic ambiguity and to optimize diagnostic tests for sensitivity and accuracy (Keller, Baldwin Ofsthun, Swearingen, Vian, Wilmering & Williams 2007). The reasoning and testability tools can be utilized during the Detailed Design phase of a system.

Octavian Niculita et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

The first response in the research community to shift the assessment of failures and associated risks into the Preliminary Design phase focused on employing safety and

reliability analysis. System safety analyses typically include Fault Tree analysis, Event Tree analysis, and Probabilistic Risk Assessments. The most common of these is the Fault Tree Analysis (FTA), which is constructed to perform a Functional Hazard Assessment (FHA). In an FHA, safety engineers identify unacceptable design features and propose design or usage changes that will mitigate the failure effects and satisfy safety requirements. FTA identifies system high-level failures that could result in loss of life or costly system equipment. For each of these critical functions, the safety engineer performs a functional analysis of how the system works in normal operating conditions, then determines which failure modes could lead to the occurrence of the critical system functional failure. FTA are characterized by a top-down approach, which starts with the identification of a high-level failure event continued by the revealing of the contributing events that could lead to the occurrence of that high-level event. Software tools supporting the Safety Analysis process include but are not limited to CAFTA™, Fault Tree +™, and Sapphire™. The most frequently types of reliability analysis are carried out using Failure Modes and Effects (Criticality) Analysis and System Reliability Predictions. FMEA/FMECA typically starts with the lowest level hardware (piece-part FMEA/FMECA) or system functions (functional FMEA/FMECA), by determining the fundamental failure modes that have a direct effect on those piece-parts or functions. The next FMEA step is the representation of system functional effects at successive indenture levels of the system. A FMECA adds the notion of failure criticality to a FMEA. The failure rates of the associated failures are often taken into consideration. Software tools supporting the Reliability Analysis process include but are not limited to OCAS™, Isograph™, Relex™.

Over the last decade, industry and academia have tried to integrate conceptual system design and diagnostic design into a common platform, in order to improve the overall system performance and availability (Brignolo, Cascio, Console, Dague, Dubois, Dressler, Millet, Rehfus & Struss 2001; Kurtoglu, Johnson, Barszczm Johnson, & Robinson 2008; Glover, Cross, Lucas, Stecki, and Stecki, 2010). A necessary condition for creating this platform is the adoption of a recognised common ontology (Wilmering, 2008). Wilmering highlighted that the two main challenges in the development of this ontology:

- Information re - use;
- Integration of the tools.

Each manufacturer of complex systems seems to have developed their own methods and apparatus for integration of system engineering, testability, diagnosability and reasoning tools.

Since most of the IVHM related research focuses on fault detection and isolation during system operation and system maintenance, approaching the IVHM design stage is still in

its early beginnings (Walker & Kapadia, 2010). Existing tools use different techniques and methods for system representation and diagnostic development and have a wide range of capabilities and performance. There is still the need to develop reliable benchmarks to quantitatively assess tool performance and effectiveness. The Diagnostic Competition defined by NASA Ames Research Centre is a good example of how to find the best diagnostic solution among tools generally used during the Detailed Design phase (Kurtoglu, Narasimhan, Poll, Garcia, Kuhn, de Kleer, van Gemund, & Feldman, 2009). Our research is trying to complement the above mentioned research, by evaluating various COTS software tools capable of integrating IVHM development process into Conceptual Design of a high-tech high-value system.

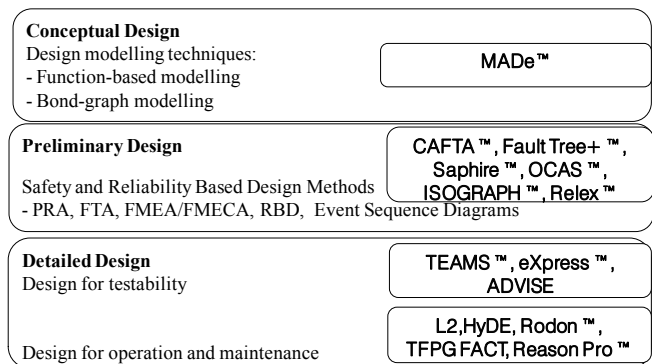


Figure 1. COTS IVHM Design software tools

This paper focuses on the use of functional analysis for the development of IVHM solutions. This type of analysis enables the integration of system failures analysis (traditionally carried out during the Preliminary Design phase through safety and reliability analysis) into the Conceptual Design phase for a new system. This approach allows for a better understanding of the failure mechanism, for a more precise identification of fault propagation paths throughout the system, and for a better system design against the faults that might arise once the system is deployed into operations.

When developing a reliable functional model for a complex system it is necessary to have access to a rigorously defined taxonomy. The origins of the functional approach used in design dates back to the mid '80s (Hubka, V. & Ernst Eder, W., 1984) (Pahl, G. & Beitz, W., 1988) and it is attributed to the European schools of design as identified by Stone & Wood (2000) and Malin, & Throop (2007). Pahl states that the quality of a system has to be built-in from the beginning of the design process and maintained throughout the whole production process. This will insure a high degree of availability, once the system is deployed into its operational environment He goes even further; emphasizing that up to 80% of all system faults can be traced back to insufficient

planning and design work. Assuming this, it can be stipulated that initiating a failure analysis from the conceptual design phase will have a positive impact on the quality of the system being designed. Later on, this approach generated different views (Ulrich & Eppinger, 1995; Ullman, 1997) in the USA. A consistent review of the ontology of functions, entities and problems, relevant to the engineering world and used by FMECA was carried out by Rudov-Clark, Stecki, and Stecki (2009) and it was concluded that the taxonomy built by Stone was the most generic taxonomy, suitable for a wide range of engineering applications, so it was used in supporting the functional analysis inside MADe™. It has been demonstrated in the last decade that functional modelling is a key step in the system conceptual design process, whether original or redesign (Stone & Wood, 2000). Stone's original taxonomy was considerable extended in the last years by exploiting component function-based behavioural modelling as part of the system engineering process (Hutchenson, McAdams, Tumer, 2012). Functional representations can be identified in both domains of software and hardware safety and reliability. The many forms of system safety and reliability analysis such as Fault Tree, Event Tree, FMECAs and Probabilistic Risk Assessments are important stages during system development as part of the Preliminary Design stage. The main drawback of these methods is that they cannot be applied at the Conceptual Design stage, since at this point models are not fully documented, the knowledge repository is not available, and accurate probability numbers are not defined with confidence. The function based design paradigm was further explored by Tumer, as part of the research focused on failure analysis of complex systems. She emphasized the integration of failure analysis for software driven hardware systems into the Conceptual Design stage (Tumer & Smidts, 2011). Kurtoglu and Tumer's research formed the baseline for the creation of a unified functional fault identification and propagation framework as part of early design stage of a new generation of high tech high value systems (Kurtoglu & Tumer, 2008).

The use of system functional analysis as part of the system design can enhance the confidence of safety analysis at the early stages and aid throughout the development of system health management capability. Health management design is generally undertaken in order to support fault detection strategies, fault isolation strategies and design of testability solutions. Fault detection analysis calculates the percentage of system faults that can be detected by defined tests. Fault isolation analysis determines the failure ambiguity groups that will result from exercising the defined tests over the fault universe. Testability analysis sometimes associated with sensor set definition and optimization will determine the optimal sequence of tests to be implemented based on the fault space, defined tests, and other optimization criteria (practicality, cost, weight, reliability). As designs become more complex, defining and implementing a testability

solution becomes more challenging. Ideally, health management capability must be developed concurrent with the design itself. Current practice does not facilitate an automatic feedback loop between test engineers and system design engineers. This feedback can be achieved through the incorporation of health management development process in the early design stage of the asset.

The functional modelling approach uses functions and flows to describe the system. Clear ontology should be provided with each functional model in order to ensure others can read it, as they might represent a blueprint of the system using a different ontology.

MADe™ software tool was selected for being the only COTS software that employs functional analysis as a method to design IVHM solution during the Conceptual Design phase of a new asset. The software provides a clear ontology, which can be used simultaneously by system designers, reliability-availability-maintainability engineers and IVHM designers in modelling the real asset and running specific type of analysis for each individual field. Another reason behind the selection of this package is the built-in connection between functional modelling and behavioural modelling within the same model. This connection allows for a fast and robust identification of the fault propagation paths throughout the system. Additional reasons behind the selection of this tool were the early validation and reusability of models, its ability to model systems/sub-systems/components/parts and the automated support for safety/reliability analysis.

This paper is the first of a series which highlights the pros and cons of existing COTS software tools employed during the health management development process. The objectives of this particular research are to establish the capability and utility of the MADe implementation of functional analysis in conceptual design of a health monitoring system for a laboratory based fuel system typical of that found on UAV aircraft. The items considered are: i) development of the fuel system functional and behavioural model ii) sensor set optimization for identification of selected faults and iii) challenges encountered throughout the implementation of the health management solution.

This paper contributes to the transformation of state of the art software tools into state of practice by identifying the advantages and shortcomings in building a functional model to aid the development and integration of IVHM Design into the Conceptual Design phase of a complex system.

## 2. FUEL SYSTEM TEST BED

Integrated systems such as those supplying fuel to aircraft engines are evolving to higher complexity with each generation and their faults are consequently becoming more

difficult to diagnose. On the other hand, the demands for extreme reliability imposed by the regulators and maximized availability imposed by the operators are continuously increasing.

We aim to take a relative simple fuel system, to illustrate the key steps of the diagnostic analysis using functional analysis and to implement the output of this analysis within an IVHM solution which meets the initial fault detection and isolation requirements. A schematic diagram of the fuel system is presented in Figure 2a. The fuel system contains a motor driven external gear pump with internal relief valve, a shut off valve, one filter, two tanks (main tank and sump tank, the last one emulating the engine), non-return valve, three-way valve to switch between recirculation and engine-feed mode, variable restrictor to simulate engine injection and back pressure when partially closed. The fuel system is representative of a small UAV engine feed. The diagnostic analysis will focus on the filter, pump, shut-off valve, pipes and nozzle failure modes. Five failure modes that are emulated on the rig are: filter clogging from foreign matter, pump degradation, valve stuck in a midrange position, leak in the main line, and a clogged nozzle.

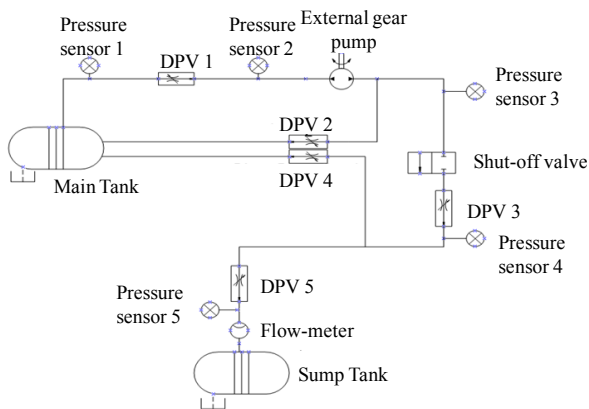


Figure 2a. Fuel system schematic

The fuel rig can accommodate various faults with different degrees of severity. When a filter clogs, the flow through the filter reduces and the pressure difference measured across the filter increases. The filter failure was emulated by replacing the filter component with a Direct-acting Proportional Valve (DPV1). Valve position fully open is equivalent to a healthy filter; partially closed being equivalent to a clogged filter with a particular degree of severity. Various degrees of severity of this fault can be simulated by varying the DPV position. In this manner, incipient, slow progression, cascading and abrupt types of faults can be simulated on the rig and the ability of the functional approach to model and address such conditions can be assessed. The physical implementation of the fuel system test bed is depicted in Figure 2b.

The physical system allows the testing and validation of various IVHM models and the assessment of the analyses

carried out using such models. Prior the construction of the physical system, a physical simulation model was developed during the fuel system design phase using a CAE COTS software tool: SimulationX™ from ITI. This modelling phase encompasses basically the sensibility studies carried out during the fuel system design phase in order to specify in a correct manner the components/system performance in order to meet the specified system requirements.

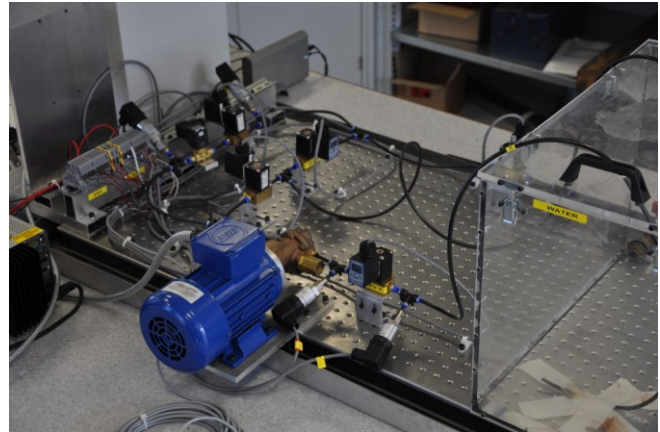


Figure 2b. Fuel system test bed

The fuel system designer is the one who can get the first insights into the system failure world, by using physical simulation models (as fluid-flow SimulationX™ model) to recreate various faulty scenarios. This knowledge should be incorporated into the diagnostic analysis models in order to ensure the consistency and accuracy of such models. The novelty of this work resides in the ability of simultaneously simulating various fuel system faults on the real system, in a simulation environment and in a functional model. The next section will describe the development of a functional model capable of aiding the development of the fuel system IVHM capability.

### 3. MADE FUNCTIONAL MODELLING FRAMEWORK APPLIED TO THE FUEL SYSTEM

Functional modelling makes use of a system model which decomposes the main system function(s) into smaller functions which are well defined for each component. This enables the assessment of the correct functionality of the system. The taxonomy used by this software package has nine major classes of functions, presented in Table 1. Each class further expands to other functions.

A primary element of any functional modelling approach is the representation of real world information corresponding to the input and output for the previously defined functions. These elements are represented by flows.

Three categories of flows, namely *energy*, *signal* and *material* were employed to capture system's characteristics. Some of the previously defined functions can accept any category of flow but others can deal only with particular

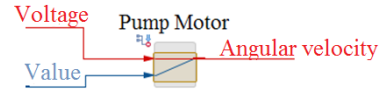
types (e.g. all three functions - to transfer, to transmit, and to transport belong to the same functional class - to channel, but, the *transfer* function accepts all type of flows; the *transmit* deals with energy or signal types of flow, while the *transport* can be used only in conjunction with material flow). Energy flows are depicted with red, signals with blue and material with green in later diagrams.

For the functional layer to be completed according to the method described in section 3, the links between all components have to be defined. This is achieved by translating the real world information exchange into distinct type of flows and by associating these as inputs and outputs for the functions defined already. These flows represent a measurable characteristic for the function (Kirschman, 1996) and are used during the Preliminary Design phase. The flow related information is mainly useful in supporting Failure Mode, Effects and Criticality Analysis (FMECA), where failures are related to the discrepancy in flow properties.

Class	Description
Branch	To cause a material or energy to no longer be joined or mixed
Channel	To cause a material or energy to move from one location to another
Connect	To bring two materials or energies together
Control	To alter or govern the size or amplitude of material, signal or energy
Convert	To change from one form of material or energy to another
Provide	To accumulate or provide material or energy
Signal	To provide information
Stop	To cease or prevent the transfer of material, signal or energy
Support	To firmly fix a material into a defined location or to secure an energy into a specific course

Table 1: Function classes provided by MADE

Figures 3a and 3b are two snapshots of the fuel system functional model. Figure 3a describes the function of the pump motor: to convert the electric energy and a specific analogue value into mechanical rotational energy. Components can be fully described following this functional approach by a single function (e.g. gear pump motor) or a combination of function (see the shut-off valve functions) as described in Figure 3b.



To convert – to change from one form of energy (*electrical energy*) or material to another form of energy (*mechanical – angular velocity*)

Figure 3a. Functional model schematic for gear pump motor component

The gear pump supplies the system with the hydraulic energy necessary to provide the flow rate and to transfer the potential contamination down the line.

The pipe between the pump and the shut-off valve *transmits* the hydraulic energy received from the pump while the shut-off valve has a suite of two different functions as follows: *to channel* the hydraulic energy down the line in the next pipe and *to regulate* the hydraulic energy in response to the information received from the solenoid. Both components (the pipe and the shut-off valve) have an extra function: *to transport* the potential contamination in the system, if such type of analysis is to be taken into consideration.

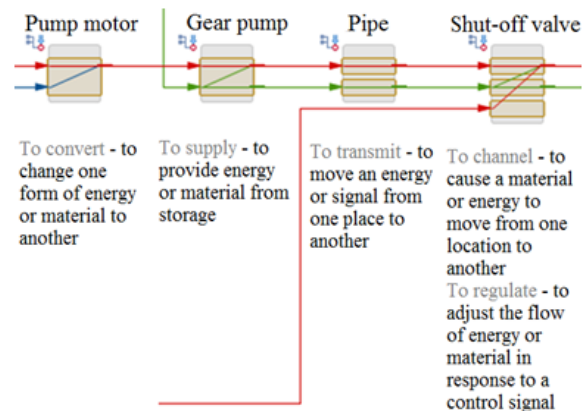


Figure 3b. Functional model schematic for gear pump motor, gear pump, pipe and shut-off valve components

Within a functional model, the inputs and output are connected inside each component by a causal link. The model can capture the polarity of these causal links depending on the effect of the input on the output (Figure 4a-Figure 4d). Each component functional model encapsulates the causality connection between two different types of flow. The pump motor functional model contains a positive connection between the voltage (the input flow presented as electrical energy) and the angular velocity (output flow presented as mechanical rotational energy). The same positive connection is found in the second connection (value-angular velocity).

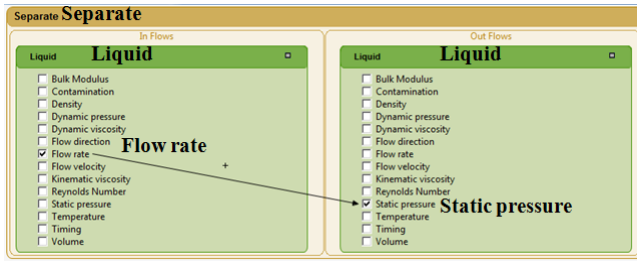


Figure 4a. Causal links between inputs and outputs flows for filter component

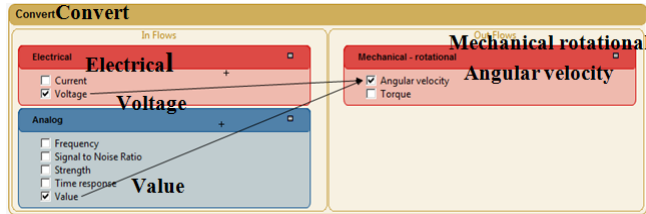


Figure 4b. Causal links between inputs and outputs flows for pump motor component

The function of the pump will be to supply flow rate as hydraulic energy by making use of angular velocity (provided by the motor) and the flow rate (provided by the inlet pipe). In order to obtain a consistent model, the output flows of a component have to match the input flows of the component positioned downstream.

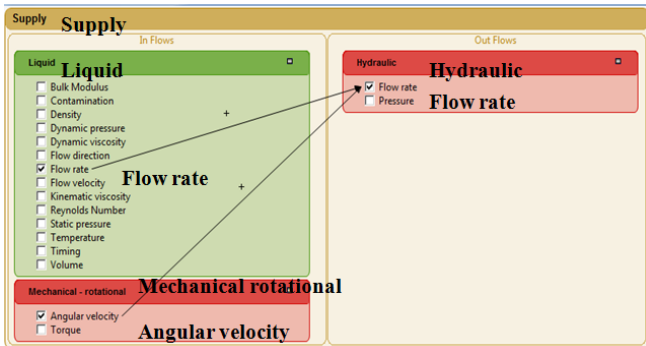


Figure 4c. Causal links between inputs and outputs flows for external gear pump component

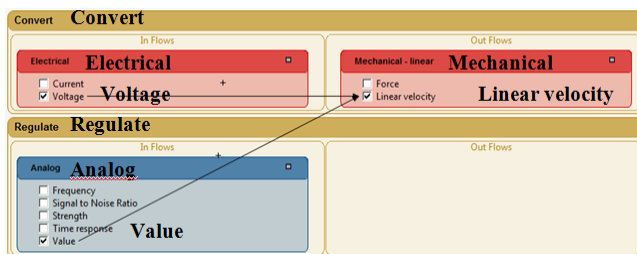


Figure 4d. Causal links between inputs and outputs flows for valve solenoid component

Figure 5 presents the full functional model at the system level and also presents the exchange of information between components using specific types of flow.

The reticence in using this tool is the fact that requires a change in failure addressing approach from physical to functional. Therefore it requires a fully adoption of its functional taxonomy in order to be able to emulate the real system into viable models.

At this point, a functional model can answer the questions related to fault propagations paths throughout the system (Glover, 2010). However some queries still remain about the causes of failures, how a system can fail, how critical each failure is and about the interaction between failures and their impact on the overall system capability. The answers to these questions can be revealed by populating the functional model with failure modes for all components and by focusing the analysis on the most critical or ones likely to be the most frequent. The functional model was populated with failure diagrams for five of the fuel system components: the filter, the gear pump, the shut-off valve, the pipe 04, and the nozzle. Once all the failure modes are defined, the system functional model is considered as being complete.

Our main goal is to design an IVHM solution capable of discriminating between nominal/faulty cases and also capable of isolating all five faults. Since the fault universe is relatively small (five faults) we aim for 100% fault detection and isolation (no ambiguity groups).

Often the failure modes are confused with failure mechanisms or the failure causes. To overcome this, the software under investigation uses a rigid terminology for failure descriptions: these must be characterized by causes, mechanism, faults, and symptoms. These four elements form a low level of abstraction regarding system knowledge, while the functional failure mode is considered to be a high level and will define the basis of the functional analysis. For example: a pipe component can leak or be clogged. These two failure modes are captured by the behavioural taxonomy as shown in Figure 6a.

Causes are linked to mechanisms, which then lead into faults that are ultimately connected to functional failures. Mechanisms and faults can present particular symptoms and these are captured accordingly in the failure diagram. These symptoms are the expression of unintended/emerging behaviour of a faulty system. Figure 6b depicts the functional model of a pump, highlighting the inputs (speed, back pressure, and pump characteristic), the function (*to produce flow*), the intended output (the flow rate response including the normal/abnormal behaviour effects (flow rate OK, too high or too low)), the failure modes and their unintended/emerging behaviour (classified as symptoms). One of the failure modes affecting the pump is represented

through a failure diagram by defining the cause(s), the mechanism(s) and faults that can be linked to the functional failure (of not supplying the required amount of hydraulic energy as volumetric flow rate). Due to the restrictions of the physics for this failure mode, this flow indicator can display either OK or too low, hence the negative causality between the fault concept and the functional failure concept. Three types of failure side effects are captured by linking the symptom concepts to the functional failure diagram previously defined as in Figure 6c.

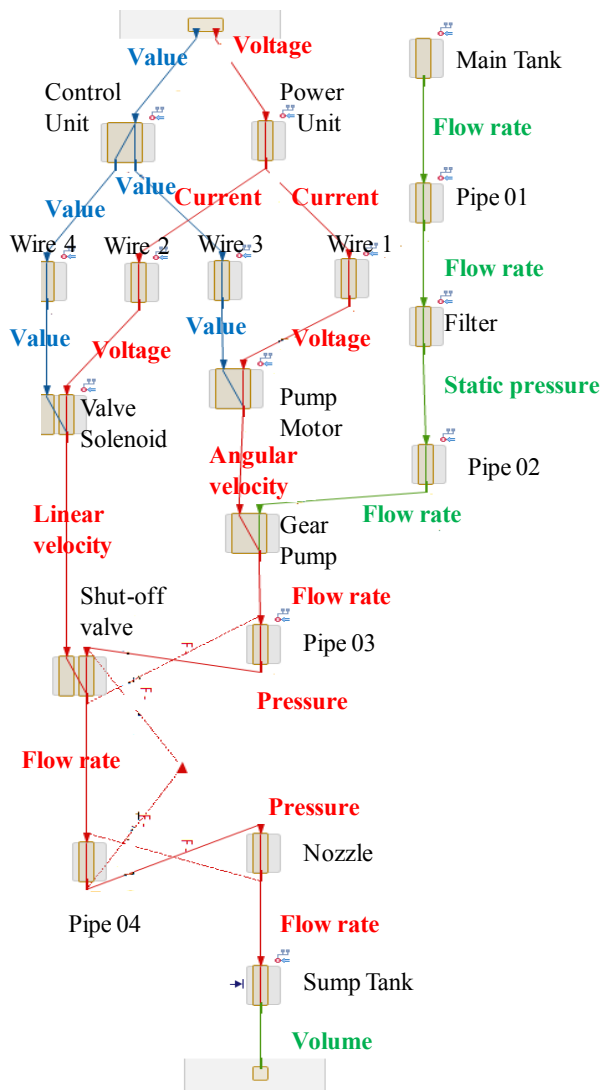


Figure 5. Fuel system functional model

The unintended/emerging behaviours complement the development of the health management solution, as they allow the user to define built-in tests for the components affected by such behaviours in order to obtain a higher probability of detection. These sensing capabilities will work in conjunction to the functional sensing capabilities identified using the functional analysis. Symptom driven

reasoning modules were proved to be extremely useful for discrimination between members of the same ambiguity group.

Causes, mechanisms and faults are all external inputs to the model and do not arise from the intrinsic functional representation of the components. Causes and faults selected for inclusion might be the ones which most frequently occur (based on service experience) or the ones which the safety analysis considers most dangerous.

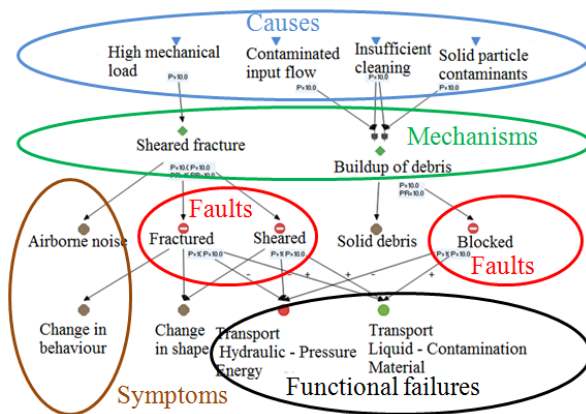


Figure 6a. Failure mode diagram, pipe example

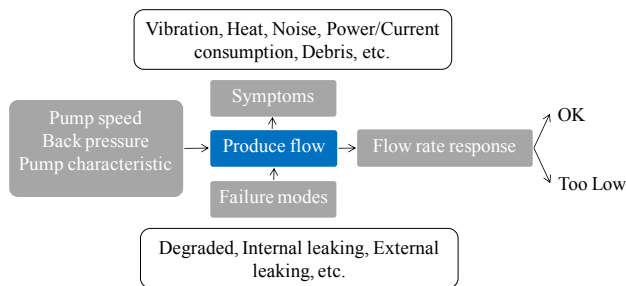


Figure 6b. Pump component, functional model

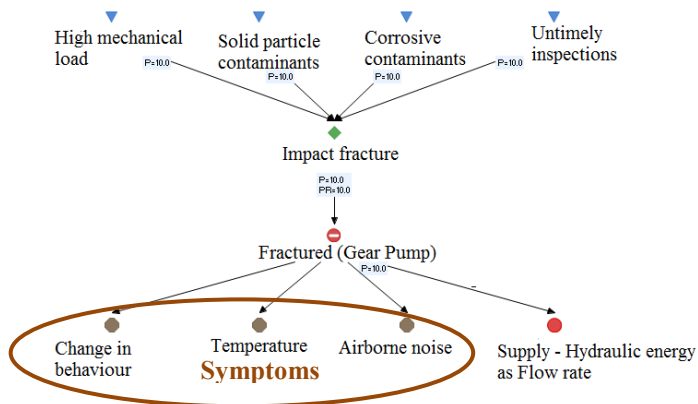


Figure 6c. Pump failure diagram, symptoms capturing

All elements of a failure diagram can be characterized by criticality numbers for difficulty of detection, occurrence, probability, progression rate and severity. We will first consider the functional failure of a pipe of *not being able to transport the hydraulic energy as pressure*. The criticality settings for this functional failure are shown in Figure 7. System Designers, Reliability Availability and Maintainability (RAM) analysts, IVHM designers and logisticians are able to populate the failure diagrams through criticality coefficients using their own experience and information expertise. Variability in detection difficulty can be represented within the functional model by choosing the functional failure to be detectable either during a flight test or a ground inspection or by setting it as a non-detectable type of failure. Occurrence, frequency and severity of a failure may be treated using the same approach. These settings will be directly fed into the calculation of Risk Priority Number (RPN) for each individual fault, enabling the realization of a complete map for safety, reliability, availability and IVHM design analysis. A real advantage of this procedure is the consistency obtained by using the same tool across the entire development process.

● Pressure			
General	Very high	High	Low
Response	Difficulty of Detection	Ground Inspection	5.0
Criticality	10		10.0
Reliability	ContinuouslyObservable	InFlightTest	Ground Inspection
	Occurrence	Reasonable Probable	6.6
	10		10.0
	ExtremelyUnlikely	Remote	Occasional
		ReasonablyProbable	Frequent
	Severity	Marginal	4.5
	10		10.0
	Minor	Marginal	Critical
			Catastrophic

Figure 7. Failure diagram - criticality settings

By accommodating this system knowledge, the model generates a propagation table describing the effect of all failure modes on the system. This operation is undertaken by propagating the erroneous output flow of the components through the software Fuzzy Cognitive Map (FCM) capability. The propagation table represents the baseline for generating the list of the functional failures. Previous research proved that the IVHM design process built on physical failure assessment is highly error prone as it does not take into account the impact of failures on other system components.

The functional model also supports safety and reliability analysis by performing the most common type of analysis: Fault Tree Analysis (FTA) and FMECAs. It is not meant to replace the detailed analysis carried out by dedicated safety

tools like: Isograph™, Relex™, CAFTA™, Sapphire™ or Fault Tree+ but having these features as part of a functional modelling tool enhances the health management development process.

Figure 8 illustrates the fault tree analysis for the fuel system with five faults: clogged filter, faulty gear pump, faulty shut-off valve, leaking pipe, clogged pipe. The five faults were propagated through the model using these failure diagrams in order to determine their end effects. The FTA incorporates a list of failure routes, which describes a sequence of events, eventually leading to a system failure. Figure 8a shows the fault tree display of a cut set with the failure of the filter marked in orange. The top element of the FTA is the function of the overall fuel system – to provide fuel – which in the case of a filter failure would be lost. The cut set represents the route through a fault tree between an event (system failure) and an initiator (component failure mode).

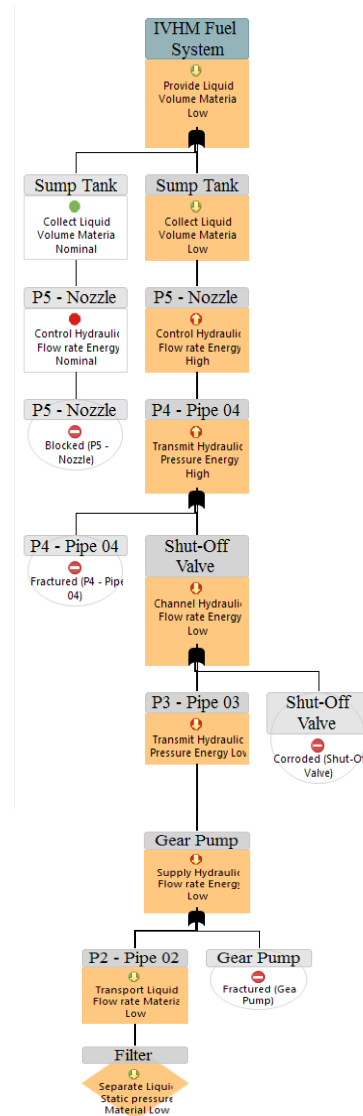




Figure 8a. Fault tree displaying the cut set for filter failure

The FTA is accompanied by the decomposition table of probability of occurrence ( $P(f)$ ) and relative importance of each individual failure mode ( $RI\%(F)$ ) as in Figure 8b.

Each component of the functional model is characterized from the criticality point of view by the duration of operation, mean time to repair and a failure distribution type (exponential or Weibull). A criticality analysis can be carried out by establishing a minimum threshold for criticality indicators or by activating the failure modes only for the components under investigation. The second option was selected for this study. Two types of reliability analyses were carried out using this software: functional reliability and hardware reliability. The results of the hardware reliability calculated using block diagrams are depicted in Figure 8c. Only the components that can be simulated on the test bed as being faulty were characterized through an exponential failure distribution defined by the part failure rate ( $\lambda$ ), mean part failure rate ( $\lambda \times 10^6$  hours) and standard deviation (failure rate).

Basic Event	P(f)	RI% (F)
IVHM Fuel System		
Filter		
Separate Liquid Static pressure Material Low	0.0001279	1.9
Gear Pump		
Fractured (Gear Pump)	0.0021822	32.4
P4 - Pipe 04		
Fractured (P4 - Pipe 04)	0.0000923	1.4
P5 - Nozzle		
Blocked (P5 - Nozzle)	0.0008919	13.3
Shut-Off Valve		
Corroded (Shut-Off Valve)	0.0035016	52

Figure 8b. FTA results

Within the same module of the software, the reliability analysis results can also be complemented by Monte Carlo simulations.

The functional analysis allows for the automatic identification of the required set of sensors and the associated logic capable of detecting and isolating each fault within the fault universe. In the case of this study, the fault universe is constituted by a clogged filter, a degraded gear pump, a shut-off valve stuck mid range, a leaking pipe and a clogged nozzle.

The sensor identification analysis enables the calculation of Fault Detection and Fault Isolation coefficients with reference to the entire fault universe. Fault Detection analysis calculates the percentage of system faults (defined as specific component failure modes) detected by the proscribed tests. Fault Isolation analysis determines the failure ambiguity groups that will result from exercising the tests over the fault universe.

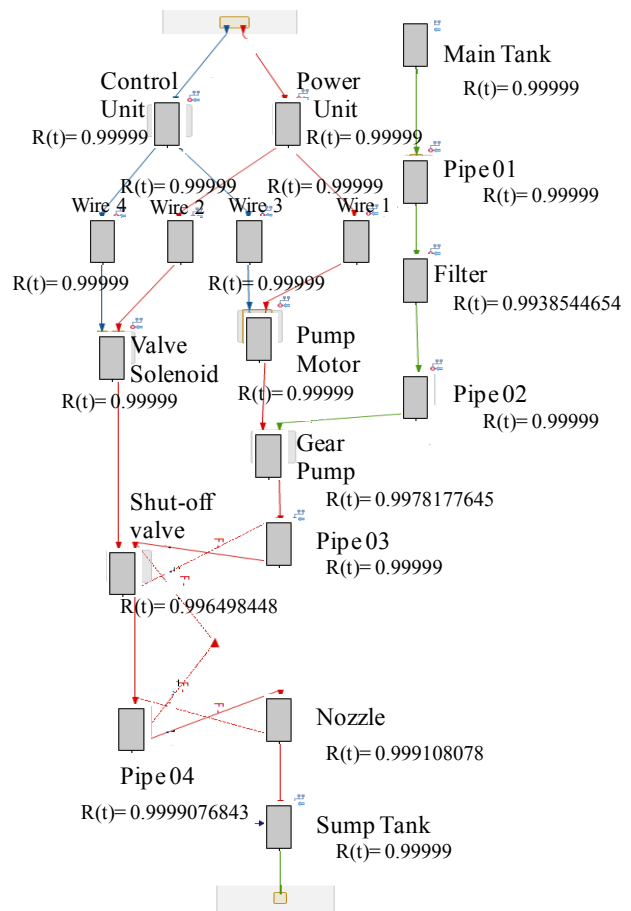


Figure 8c. Fuel system - Hardware reliability

IVHM designers require tools able to run what-if type of analyses in order to identify the optimum health management solution meeting the fault detection and fault isolation requirements. Due to different types of restrictions in terms of weight, costs, performance, probability of detection of the HM solution on one side and sensor reliability on the other side, FDI coefficients might have to be dropped below the required figures. Ambiguity groups - collections of failure modes with the same system response - will appear in such situations. Once sensors have been identified, the diagnostic analysis also provides the core elements for a reasoning capability/expert system. Once a deviation of a particular parameter outside its nominal range is detected, the function of that component will be evaluated by examining high-level functional failure mode layers and the causes, mechanisms, faults and symptoms described in the lower levels of the model.

#### 4. SENSOR SET OPTIMISATION

The fuel system sensor set optimization analysis is based on the functional model developed in the previous section. As

mentioned before, the overall function of the fuel system is to provide a volume of fuel when commanded by an electric signal assuming that it was powered up.

When each of the five faults is injected in the system through its own failure diagrams, the tool will generate a propagation path of the failure. The propagation paths are collated into a propagation table which will form the basis for the sensor set optimization process. Each line from the Table 2 represents the effects of the five failure modes on the output flow(s) of the other components.

The vector (F.G.P.P.P.S.S.) represents the Filter *pressure*, Gear pump *flow rate*, Pipe 02 *flow rate*, Pipe 03 *flow rate*, Pipe 04 *pressure*, Nozzle *flow rate*, Shut-off valve *flow rate*, Sump tank *volume* output parameters. The arrow-up and arrow-down symbols mark the deviation of these parameter outside normal boundaries when a particular failure mode was injected and propagated throughout the functional model.

For example, the first row of Table 2 shows the effects of a clogged filter on the system response. A clogged filter determines the flow rate in pipe 02 to decrease; this will impact the flow rate down the line in the gear pump, pipe 03, pipe 04, shut-off valve and the sump tank.

COMPONENT	FLOW	FAILURE	F	G	P	P	P	P	S	S
FILTER	PRESSURE	LOW	↓	↓	↓	↓	↓	↓	↓	↓
PUMP	FLOW	LOW	↓	↓	↓	↓	↓	↓	↓	↓
PIPE	PRESSURE	LOW	↓	↓	↓	↓	↓	↓	↑	↓
NOZZLE	FLOW	LOW	↑	↓	↑	↑	↑	↓	↓	↓
VALVE	FLOW	LOW	↑	↓	↑	↑	↓	↓	↓	↓

Table 2. Fuel system propagation table obtained using functional analysis

The functional modelling technique (input flows linked to output flows throughout the functional layer) might induce the designer to model only the downstream effects of a particular fault. The Fuzzy Cognitive Maps (FCM) method employed by this tool also allows for modelling of upstream effects. The dotted red line in Figure 9 is the causal connection representing the effect of a clogged nozzle on the upstream component (Pipe 04). If a nozzle is getting clogged, the output parameter (flow rate) decreases. The variation of this flow rate parameter has an inversely proportional effect on the Pipe 04 input flow rate parameter increasing afterwards the output pressure. This behaviour could not have been achieved without the negative feedback loop depicted in Figure 9. The same technique will be replicated upstream using iterative loops throughout the system until all the effects of a clogged nozzle would be captured by the model.

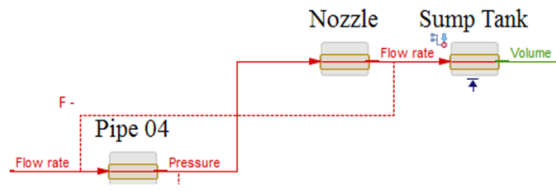


Figure 9. FCM Feedback loops

A functional model will therefore be consistent when the propagation table generated by the model will replicate from a qualitative point of view the physics of the system under similar faulty conditions.

The qualitative characteristics of each individual failure contained in the propagation table are processed by an optimization algorithm in order to identify the combination of elements which allow discriminating between them. The elements mentioned above are in fact the flows captured in the functional analysis and the type of flows will determine the type of sensors to be used to identify a particular fault. A detailed description of the sensor set discrimination analysis using this software is presented by Rudov-Clark (Rudov-Clark, 2009).

Functional analysis is a qualitative analysis. This type of qualitative analysis identifies the foundation of an HM solution for a given system for a known fault universe. As mentioned in the previous section, for this particular scenario of the fuel system, the fault universe is composed by five distinct faults. The optimization algorithm generates 6 sensor set solutions, with maximum coverage and no ambiguity groups. One of the solutions contains four sensors and is presented in Figure 10 and it comprises of:

- S1 - a sensor measuring the *static pressure* after the Filter,
- S2 - a sensor measuring the *flow rate* after the Gear pump,
- S3 - a sensor measuring the *flow rate* after the Shut-off valve
- S4 - a sensor measuring the *pressure* in the Pipe 04.

The components that need to be monitored are marked with green. The type of sensor is determined by the type of the output flow of those components. By making use of the output from these sensors, the analysis shows that it is possible to discriminate with 100% confidence between all five faults by using a particular diagnostic logic. Obviously, this was expected as the structure of the fuel system is quite simple, and there are only five faults.

Based on this analysis, the IVHM designer has information about the location of the sensors contained in each sensor set, and also information regarding the type of the monitored flow. All the sensor set solutions are complemented by fault detection, fault isolation and ambiguity group indicators (if any).

Each sensor set also contains the diagnostic rules to be implemented on the real system to enable the fault

identification. Most of the time, the diagnostic rules are associated with tests that are carried out during the diagnostic and troubleshooting procedures.

Functional models are typically used for either analysis of a system's potential to realize health management goals – as a design aid - or to support execution of fault isolation reasoning. The output of the functional analysis represents the input to other tools that actually create, or help direct the creation of, executable system diagnostics. The qualitative diagnostic layer produced by MADE will have to be complemented by a quantitative layer obtained by physical simulation of the system flows.

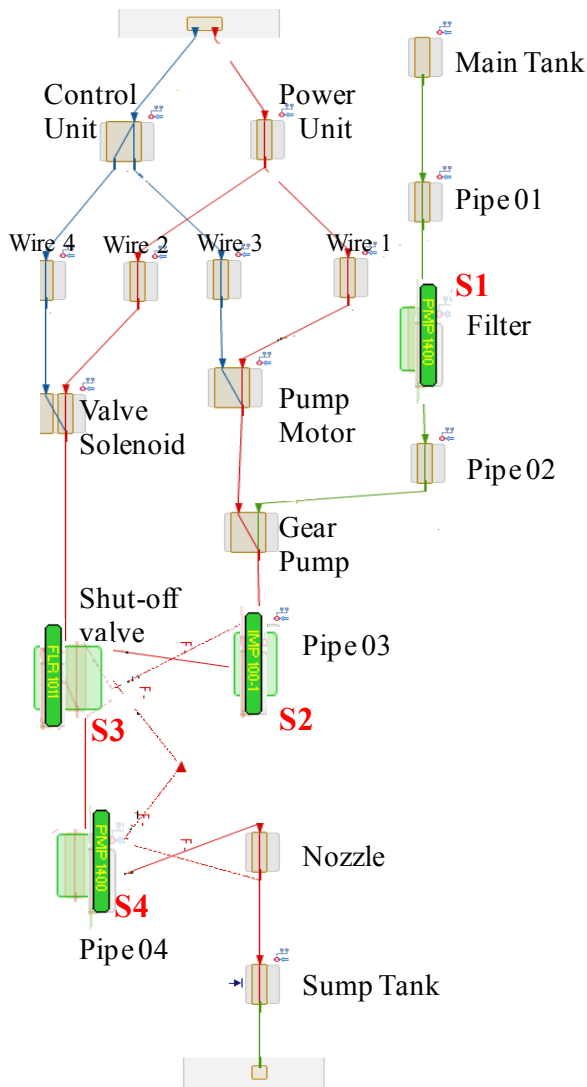
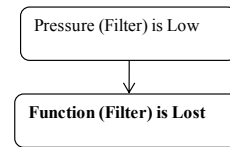


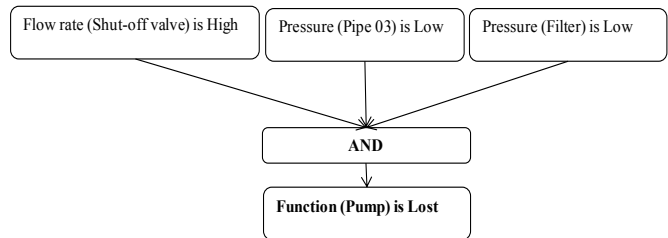
Figure 10. One of the functional sensor set solutions

Figure 11 shows the diagnostic rules associated with the sensor set solution from Figure 10.

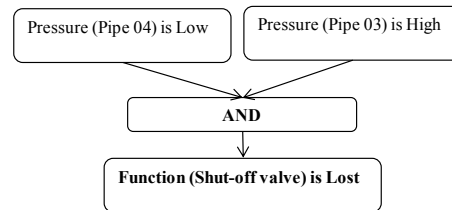
Clogged filter:



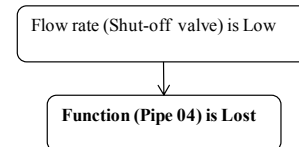
Degraded gear pump:



Shut-off valve stuck mid range:



Leaking Pipe:



Clogged nozzle:

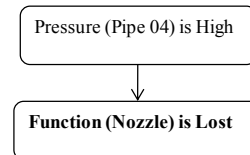


Figure 11. Diagnostic rules for the optimized sensor set solution depicted in Figure 10

These rules form the baseline for diagnostic engine of the fuel system. For example the function of the gear pump (to deliver a specific amount of flow rate) is classified as lost if the sensor monitoring the shut-off valve flow rate has an output value below nominal, the pressure in pipe 03 below nominal and the static pressure at the filter is below

nominal values. In all other cases, the function is considered to be operating normally. In a similar manner, the other four diagnostic rules describe the lost of filter, shut-off valve, pipe and nozzle’s functionality by making use of the output of the sensors (S1-S4).

Information on weight, cost and reliability can be associated with each sensor in order to get a clear image of the comparison between different IVHM solutions (as in Table 3). The software might further generate queries about various sensor set solutions as in Figure 12.

Name	Type	# of Sensors	Coverage	Possible Coverage	Cost	Weight (g)
Sensor Set 1	FCM	4	100%	100%	£910.00	175.00
Sensor Set 2	FCM	4	100%	100%	£1,300.00	100.00
Sensor Set 3	FCM	4	100%	100%	£1,010.00	150.00
Sensor Set 4	FCM	4	100%	100%	£1,200.00	125.00
Sensor Set 5	FCM	4	100%	100%	£500.00	30.00
Sensor Set 6	FCM	4	100%	100%	£1,080.00	115.00

Table 3. Sensor set additional information

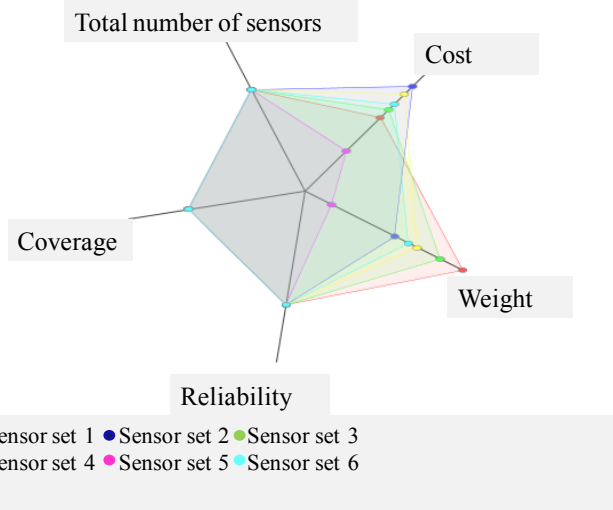


Figure 12. Sensor set comparison query

The information generated by functional analysis can be further used by the HM designer in developing the executable HM solution and also by the system designer in analyzing the impact on the overall design once this solution is integrated with the asset.

### 5. IVHM SOLUTION – VERIFICATION AND VALIDATION

The sensor set solutions identified and highlighted in the previous section were embedded in the physical test bed and the functional diagnostic rules were assessed against physical simulation results.

For each component, a physical degradation scenario was simulated and system behaviour in terms of pressure and volumetric flow rates were captured.

Figure 13 shows the variation of the pressure parameters in various points of the system vs. the degradation of the filter. As described in section 2, the filter degradation was emulated by gradually closing the DPV1 (see Figure 2a). Nine degrees of severity were generated corresponding to 100%, 90%, 80%, 70%, 60%, 50%, 40%, 30% and 20% valve opening. Position 100% valve open corresponds to a healthy filter, 20% valve open corresponds to an almost clogged filter. The mean values of the volumetric flow rate and pressure for each set of data (1 minute at 1kHz) were plotted against valve opening coefficients and the results were interpolated.

Although, a real operating environment might present more noise and the sensor sensitivity and accuracy as well as the parameters magnitude levels might be different on a real fuel system than in the test-bed, the effects of a failure mode on the real asset will be similar in both cases from a qualitative point of view.

Clogged filter diagnostic rule:

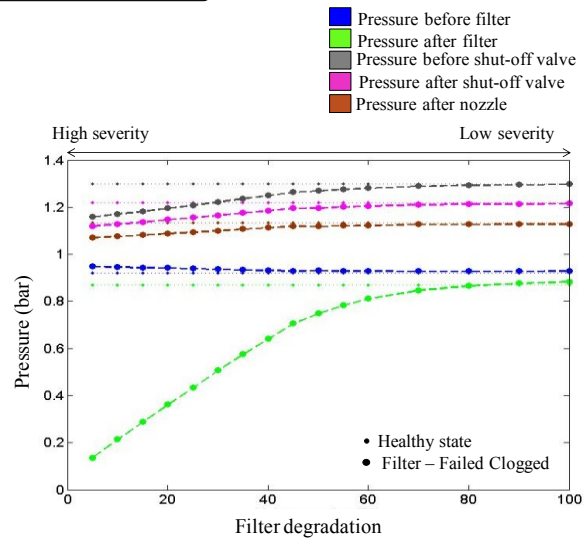
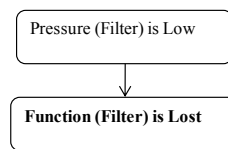


Figure 13. Pressure rates under a filter degradation scenario

From a diagnostic perspective, the pressure drop across the filter is generally used for the identification of a clogged filter (SAE J905, 2009). Functional approach demonstrated that by using strictly the pressure after the filter (the green signal in Figure 13), this failure can be fully detected and isolated. This quantifies in the reduction of the number of sensors required to identify a clogged filter. It is widely accepted that the reliance on one fault indicator (pressure after the filter) is not a robust detection method however one

potential solution to make the indicator more reliable might be the use of multiple redundant indicators that provide a means for resolving differences (e.g. by “voting”).

The degraded pump scenario was implemented by creating a leak after the pump component through the DPV2 (see Figure 2a). Initially, for the healthy situation, this valve was fully closed. Gradually, the DPV2 position was set to 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% and 100% open. The mean values of the volumetric flow rate and pressure for each set of data (1 minute at 1kHz) were plotted against valve opening coefficients and the results were interpolated. As in the previous case, difference between the emulation approach adopted here and the degradation of a real pump might exist however from a qualitative point of view, a degrade pump will provide a lower flow rate for the same load (system configuration) and same pump speed. On the test-bed, the pump controls have a feedback loop which keeps the pump speed constant at all time.

The diagnostic rule for detection and isolation of a degraded pump combines the variation of three parameters: the increase of the flow rate through the shut-off valve (equivalent to a decrease in pressure difference across the shut-off valve), the decrease in the pressure measured after the pump and the decrease in the pressure after the filter. The consistency of the diagnostic rule can easily be verified by observing the variation of parameters P2, P3 and the difference between P2 and P3 in Figure 14.

Degraded pump diagnostic rule:

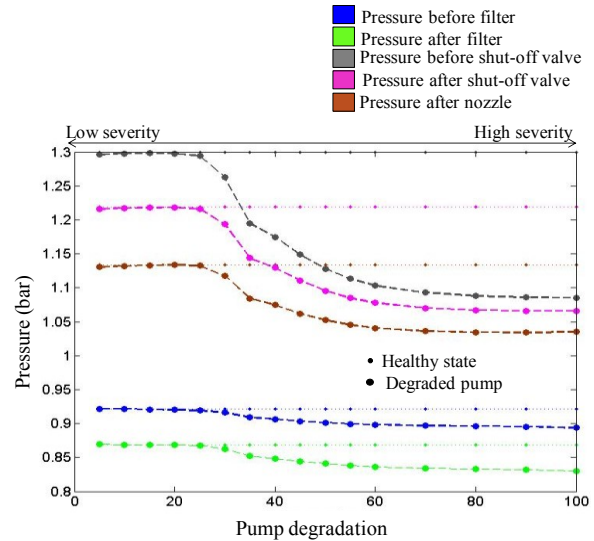
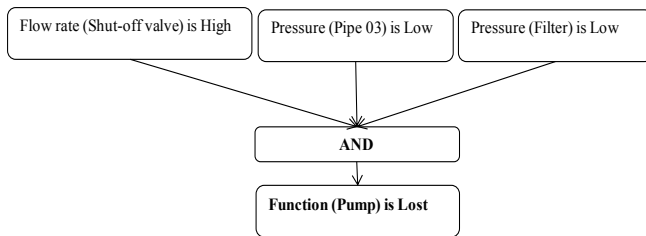
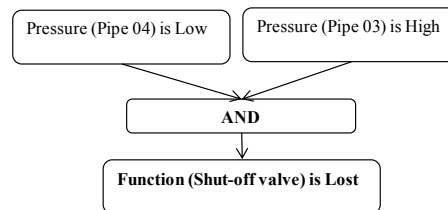


Figure 14. Pressure rates under a degraded pump scenario

The malfunction of the shut-off valve (stuck in a midrange position) was implemented by adding the DPV3 in line with the shut-off valve (see Figure 2a). Initially, for the healthy situation, the DPV3 was fully open. Gradually, the direct-acting proportional valve was closed and data was measured on the rig for 1 minute at 1kHz for the valve being 100%, 90%, 80%, 70%, 60%, 50%, 40%, 30%, and 20% open. The mean values of the volumetric flow rate and pressure for each set of data were plotted against valve opening coefficients and the results were interpolated.

The diagnostic rule for detection and isolation of a stuck shut-off combines the variation of two parameters: the increase in the pressure measured before the valve and the decrease in the pressure measured after the pump. The consistency of the diagnostic rule can easily be verified by observing the variation of parameters P3 and P4 in Figure 15.

Shut-off valve stuck mid range diagnostic rule:



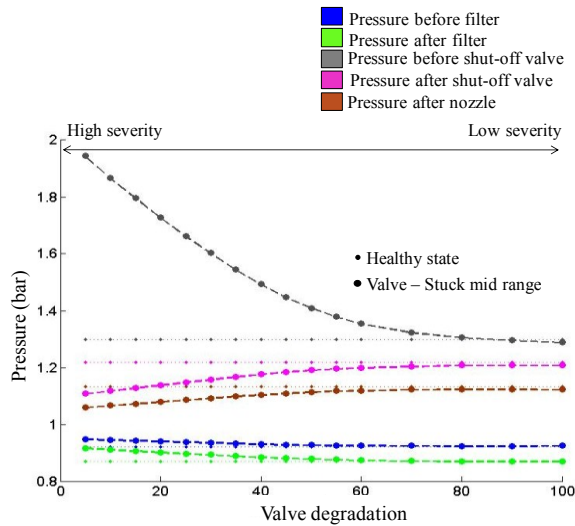


Figure 15. Pressure rates under a degraded shut-off scenario

The leaking pipe scenario was implemented by creating a leak after the pump component through the DPV4 (see Figure 2a). Initially, for the healthy situation, this valve was fully closed. Gradually, the DPV4 position was set to 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% and 100% open. The mean values of the volumetric flow rate and pressure for each set of data (1 minute at 1kHz) were plotted against valve opening coefficients and the results were interpolated.

The diagnostic rule identifying a leak in the Pipe 04 implies the drop in the flow rate through the shut-off valve (equivalent to the flow throughout the system). The consistency of flow in Figure 16a. The flow rate through the shut-off valve parameter can be interpolated from the pressure difference across the shut-off valve as per Figure 16b. The decrease of the flow rate through the shut-off valve is equivalent to the increase in the pressure drop across the valve (the difference between parameters P3 and P4).

Leaking pipe diagnostic rule:

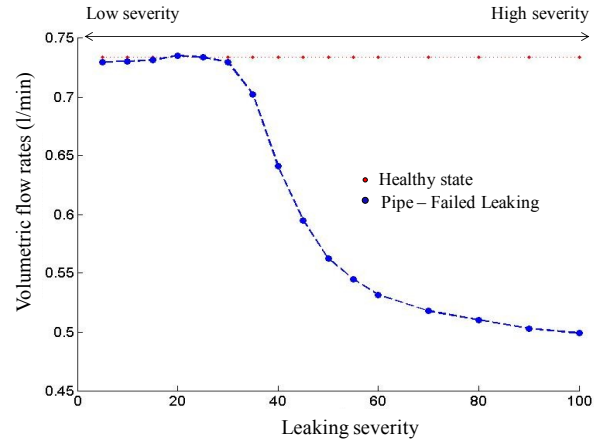
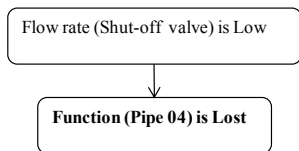


Figure 16a. Flow rates under a leaking pipe scenario

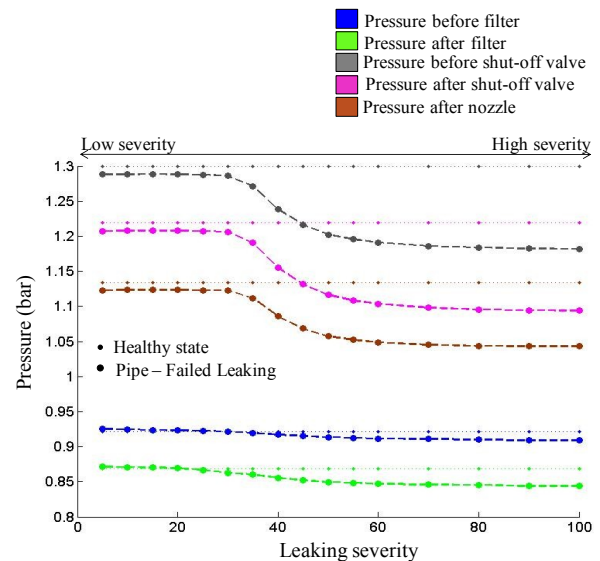


Figure 16b. Pressure rates under a leaking pipe scenario

The clogged nozzle scenario was implemented by replacing nozzle component with DPV5 (see Figure 2a). Initially, for the healthy situation, this valve was fully open. Gradually, the DPV5 was closed and data was measured on the rig for 1 minute for the valve being 100%, 90%, 80%, 70%, 60%, 50%, 40%, 30%, and 20% open. The mean values of the volumetric flow rate and pressure for each set of data were plotted against valve opening coefficients and the results were interpolated.

The diagnostic rule identified by the functional model as being capable of detecting and isolating the clogged nozzle involves the pressure measured before the nozzle. When this parameter increases above normal limits, the function of the nozzle is considered lost (equivalent to the nozzle being clogged). The consistency of this diagnostic rule can be verified by observing the variation of this parameter for all

degrees of severity of a clogged nozzle scenario (P4 in Figure 17).

Clogged nozzle diagnostic rule:

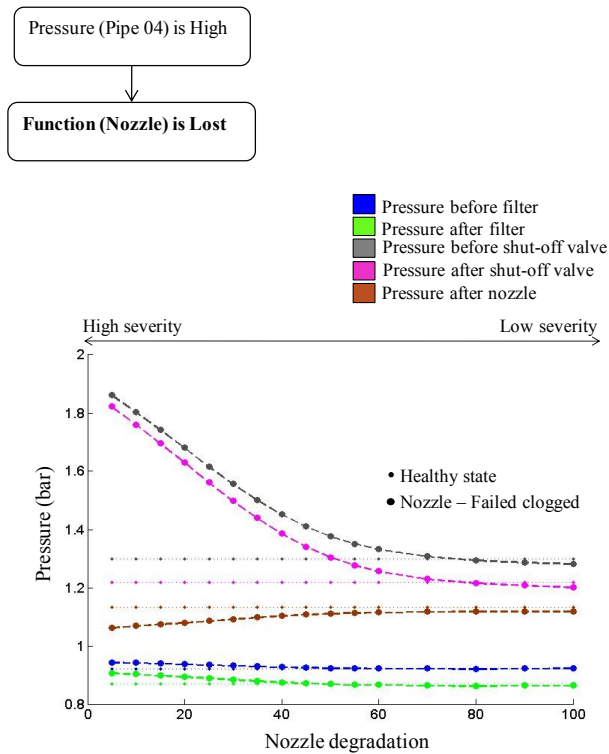


Figure 17. Pressure rates under a nozzle degradation scenario

To sum up, the sensor set solution identified by the functional analysis (four sensors – three pressure sensors and one flow meter) can be further minimized by inferring the flow rate from two of the existing pressure sensors. This second optimisation is strictly related to the implementation process of the sensor set solution on the real asset and is not part of the IVHM design process.

In order to incorporate the sensor set solution on a real asset and to deploy the diagnostic rules as part of the IVHM solution, several calibration procedures have been performed. As implied in Figure 11, diagnostic rules for a sensor set solution contain qualitative information only. Faults are indicated by deviations from nominal healthy values using *Low/High* indicators without a quantitative threshold defining the separation between “normal” and “low” conditions or between “normal” and “high” conditions.

## 6. DISCUSSION

The fuel system test case demonstrates that functional analysis can support the initial evaluation and assessment of functional-failure risks of physical systems. Functional

approach in general and MADe™ software in particular have interesting positive and negative characteristics. These are:

Pros:

1. Functional modelling brings a reasonable degree of methodology to the future automated processes of impact analysis. By integrating its rigorous taxonomy, functional modelling solves one of the shortcomings of the current IVMH design approach: the lack of a clear, unified and widely accepted ontology.
2. The whole reason for introduction of functional analysis was to offer designers a tool to be used during the conceptual design when there is no physical system. No previous information needs to be provided to the model regarding possible fault propagation routes. Diagnostic analysis tools like eXpress™ (Hilberth, 1995; Gould, 2004), ADVISE (Keller, 2007), TEAMS™ (QSI, 2012) require IVHM designers to explicitly build the fault propagation model by using causal relationships between part, component, sub-systems and systems. Regarding functional approach, only the information related to structural topology and system functionality are used as input parameters for configuring the model.
3. If combined with the appropriate information like difficulty of detection, progression rate, occurrence, probability, severity, functional analysis can complement and/or substitute safety and reliability analysis. It allows the designers to focus on functional failures rather than physical failures.
4. Another important feature of the functional analysis is its capability to identify end-effects (upstream and downstream) of a particular component failure mode. By employing FCM (Fuzzy Cognitive Maps) techniques, MADe is able to identify functional failures resulting from component interaction.
5. Functional approach can easily handle high complexity factors of the analysis. Any number of parts/components/sub-systems or system failures can be introduced in the model using failure diagrams (cause-mechanism-fault-functional failure). The advantages of modelling a system using a high degree of complexity remain though in discussion.
6. The health management solutions developed using this COTS functional analysis software become more reliable as they are constructed using models that incorporate simultaneously design, RAM and IVHM information.
7. Considering the multitude of multidisciplinary data incorporated into a functional model, it can be

concluded that functional analysis offers significant support for conceptual design analysis in vertical integration and system engineering.

Cons:

1. System designers might find the mindset shift from physical approach to functional approach difficult to adopt. The concepts underpinning the functional approach might be easier to be adopted by safety and reliability engineers.
2. Functional failure analysis provides only qualitative predictions of system parameter changes. Further tests and calibration procedures have to be performed in order to transform the functional analysis output into executable code.
3. It was demonstrated that the proposed instrumentation solutions obtained using functional approach are capable of detecting and isolating the faults defined in the fault universe. As they do not involve any physics, the authors have identified a drawback as no assessment can be undertaken regarding the efficiency of functional sensor set solutions for prognostic purposes at this stage.

## 7. CONCLUDING REMARKS

The paper has presented the results of an applied research project that combined functional analysis with behaviour simulation as a design aid for developing health management solutions.

First of all, a functional model of a fuel rig was developed using MADe™ and a series of analyses to aid the health management development process were carried out. The model of the system incorporates all its components, topology, functions, failure modes and failure diagrams, in order to simulate failure propagation paths and local, next and end-effects on other components. Care must be taken in setting up the functional model with the choice of component functions and inflow/outflow to these functions. The validation of the functional model consistency was made with reference to the physical simulation model of the same system. A functional model will be considered consistent when the functional propagation table incorporating failure effects on the system matches the physical propagation table containing the effects of similar failures.

Secondly, functional modelling for IVHM design and fuel system design cannot be done in isolation. This paper described the steps required to correctly develop a functional model that will reflect the physical knowledge inherently known about a given system. This means that during this development process, the model required several amendments including the use of several feedback loops to

replicate the upstream effects of a failure throughout the system in order to ensure good alignment with real physical behaviour. Since functional qualitative model makes explicit the essential distinction between normal and faulty conditions, it covers classes of components rather than individual ones, thus facilitating component libraries. The MADe™ framework employs modular, reusable function-component-behaviour models that can be integrated using an industrial standard.

Thirdly, the functional qualitative models offer means of expressing and exploiting approximate knowledge on a firm theoretical basis in a formal model. For instance, knowledge about qualitative deviations of component behaviour is used to identify the detection and isolation diagnostic rules.

If functional models are populated with reliability, availability and maintainability figures, the design engineers are able to systematically investigate functional-risks as early as possible during the design cycle. Simultaneously, the functional model increases the rate of success in deploying an IVHM solution with a reliable diagnostic logic. During the testing phase, the initial calibration of the diagnostic rules has to be done in conjunction with the physical simulation. This calibration will need further adjustments once the IVHM solution is fully deployed on the real system in real operating conditions.

Besides functional analysis, a further input to the PHM community is the construction of physical fuel system test-bed for assessment and validation of various IVHM design techniques. As future work, the diagnostic logic obtained using functional failure analysis will be tested on the real asset and the propagation of failures and assessment of their impact on the overall system capability will be compared with the results obtained by using other COTS IVHM design tools.

Areas identified for future work include the implementation of the proposed functional techniques on a large-scale, highly complex landing gear system and on a central frequency changer house, both systems having huge downtimes and high repair costs associated. These systems will allow investigations regarding the efficiency of the functional approach when used in various domains (hydraulic and electric).

## ACKNOWLEDGEMENT

This research is funded by the IVHM Centre, Cranfield University. The authors would like to thank K. Keller and T. Wilmering at the Boeing Company, St. Louis and J. Stecki at PHM Technology for valuable discussion and guidance during this project.



## REFERENCES

- Brignolo, R., Cascio, F., Console, L., Dague P., Dubois, P., Dressler, O., Millet, D., Rehfus, B., & Struss, P., (2001). Integration of Design and Diagnosis into a Common Process. In: *Electronic Systems for Vehicles, VDI Verlag, Duesseldorf*, pp. 53-73.
- Combitech, (2012), [www.combitech.se/sv/Tjanster/Informationsforsorjning/Produkter--verktyg/](http://www.combitech.se/sv/Tjanster/Informationsforsorjning/Produkter--verktyg/)
- DSI (2012). eXpress Diagnostic Software. <http://www.dsiintl.com/weblogic/Products.aspx>
- Hubka, V., & Ernst Eder, W., (1984). Theory of Technical Systems, *Springer-Verlag, Berlin*.
- Glover W., Cross J., Lucas A., Stecki C., & Stecki J., (2010). The use of prognostic health management for autonomous unmanned air systems, *Proceeding of International Conference on Prognostics and Health Management*, October 10-16, Portland, Oregon, USA.
- Gould E., (2004). Modelling it both ways: Hybrid diagnostic modelling and its application to hierarchical system design, DSI International, *AUTOTESTCON*, pp. 576-582.
- Hilberth, K., (1995). Dependency modelling as an input to MIL-STD TRD/TPS development. *AUTOTESTCON*, pp. 552-555.
- Hutcheson, R.S., McAdams, D.A. & Tumer I.Y., (2012). Function-based behavioural modelling. *Int. J. Computer Aided Engineering and Technologies*, Vol. 4, No. 3, pp. 250-271.
- Impact Technologies, (2012), <http://www.impacttek.com/Resources/SellSheetPDFs/ReasonPro.pdf>
- ITI, (2012). <http://www.itisim.com/simulationx/>
- Keller, K., Baldwin, A., Ofsthun, S., Swearingen, K., Vian, J., Wilmering, T., & Williams, Z., (2007). Health Management Engineering Environment and Open Integration Platform. *Proceedings of IEEE Aerospace Conference*, vol., no., pp.1-16 , paper no. 1319.
- Kirschman, C. F., Fadel, G. M., & Jara-Almonte, C. C., (1996). Classifying functions for mechanical design. *Proceedings of ASME Design Engineering Technical Conference and Computers in Engineering Conference*, Paper ID 96-DETC/DTM1504, Irvine, August 18-22.
- Kurtoglu, T., Johnson, S. B., Barszcz, E., Johnson, J. R., & Robinson, P. I. (2008). Integrating system health management into the early design of aerospace systems using Functional Fault Analysis, in *Proceedings. International Conference on Prognostics Health Management*, Denver, CO, pp. 1-11.
- Kurtoglu, T., & Tumer, I.Y., (2008). A graph-based fault identification and propagation framework for design of complex systems. *Journal of Mechanical Design*, vol. 130, no. 5, pp. 051401-1-051401-8.
- Kurtoglu, T., Narasimhan, S., Poll, S., Garcia, D., Kuhn, L., de Kleer, J., van Gemund, A., and Feldman, A., (2009). First international diagnosis competition - DXC'09, *In Proc. DX'09*, pp. 383-396.
- Malin, J. T., & Throop, D. R., (2007). Basic concepts and distinctions for an Aerospace ontology of functions, entities and problems. *in Proc. IEEAC*, pp 1-18.
- Pahl, G. & Beitz, W., (1988). Engineering Design: A Systematic Approach, *Springer-Verlag*. Berlin
- Rudov-Clark, Stecki C., & Stecki, J., (2009). The language of FMEA: on the effective use and reuse of FMEA data, *Sixth DSTO International Conference on Health & Usage Monitoring*.
- Rudov-Clark, S. J., Ryan, A.J., Stecki, C.M., & Stecki, J., (2009). Automated design and optimization of sensor sets for Condition-Based Monitoring, *Sixth DSTO International Conference on Health & Usage Monitoring*.
- QSI (2012), Testability engineering and maintenance system (TEAMS) Tool, <http://www.teamqsi.com/prods.html>
- SAE J905, (2009). "Fuel filter test methods", *SAE International recommended practice*
- Stone, R.B., & Wood, K.L., (2000). Development of a functional basis for design, *Journal of Mechanical Design*, pp. 359-370.
- Tumer, I.Y., & Smidts, C.S., (2011). Integrated Design-Stage Failure Analysis of Software-Driven Hardware Systems, *Computers, IEEE Transactions on* , vol.60, no.8, pp.1072-1084.
- Ullman, D., (1997). The Mechanical Design Process. 2nd ed., *McGraw-Hill*.
- Ulrich, K. & Eppinger, S., (1995), Product Design and Development, *McGraw-Hill*.
- Walker, M., & Kapadia, R., (2009). Integrated Design of On-line Health and Prognostics Management, *Proceeding of International Conference on Prognostics and Health Management*. San Diego, CA September 27 – October 1, USA.
- Wilmering, T.J., (2008). A practical examination of model reuse in support of health management application integration, *AUTOTESTCON*, pp. 189-194.

## BIOGRAPHIES



**Octavian Niculita** gained his BSc in Automation Control and Computer Science (Technical University of Iasi, Romania). He undertook his PhD research at TUIASI, Romania and at the University of Ferrara, Italy as part of the European Doctorate of Sound and Vibration Studies. Since 2009 he has been actively involved in the first project of the IVHM Centre regarding health management development process and construction of test facilities for assessment of IVHM design and deployment software tools for high-tech, high value systems.

number of technical roles, gaining experience in aerodynamics, heat transfer, fluid systems, mechanical design, combustion, services and IVHM. He moved to Cranfield in July 2008 as Professor and Director of the newly formed IVHM Centre. He has led the development and growth of the Centre, in research and education, over the last four years. Ian is on the editorial Board for the International Journal of Condition Monitoring, a Director of the PHM Society, contributing member of the SAE IVHM Steering Group and HM-1 IVHM committee, a Fellow of IMechE, RAeS and ASME. He is the editor of the recent SAE book: IVHM – Perspectives on an Emerging Field.



Professor **Phil Irving** gained his BSc and PhD at Birmingham University. In the early 1970s he joined the very active fracture mechanics research group at Birmingham, performing some of the earliest measurements of fatigue crack growth thresholds in metallic materials. In 1973 he moved to the National Physical Laboratory where he developed techniques for determining stress corrosion cracking behaviour of steels in high pressure hydrogen gas environments, and continued with exploration of factors controlling fatigue cracking in high strength steels. In 1978 he began work at GKN Technology, the research and development organisation of the GKN Automotive group, where for the first time he became involved in putting his knowledge of fatigue and fracture to practical use in design and service life prediction of a wide range of automotive components. In 1991 he moved to Cranfield, accepting a CAA sponsored Chair in Damage Tolerance. In the time since then he has worked extensively on fatigue fracture and damage tolerance in helicopters and fixed wing aircraft. A continuing theme has been research into techniques and benefits of structural health monitoring to structural integrity of aircraft. This interest leads to his close involvement in the setting up of the Cranfield Integrated Vehicle Health Management (IVHM) Centre and his pursuit of IVHM and SHM research within the centre. In addition to his research work he has lectured and published extensively in his research areas and has been external examiner at undergraduate, masters and PhD level on many occasions. He is a Chartered Engineer and a Fellow of IM3.



Professor **Ian K Jennions** career spans over 30 years, working mostly for a variety of gas turbine companies. He has a Mechanical Engineering degree and a PhD in CFD both from Imperial College, London. He has worked for Rolls-Royce (twice), General Electric and Alstom in a

# A Batch Detection Algorithm Installed on a Test Bench

Jérôme Lacaille<sup>1</sup>, Valerio Gerez<sup>1</sup>

<sup>1</sup>*Snecma, 77550 Moissy-Cramayel, France*

[jerome.lacaille@snecma.fr](mailto:jerome.lacaille@snecma.fr)

[valerio.gerez@snecma.fr](mailto:valerio.gerez@snecma.fr)

## ABSTRACT

Test benches are used to evaluate the performance of new turbofan engine parts during development phases. This can be especially risky for the bench itself because no one can predict in advance whether the component will behave properly. Moreover, a broken bench is often much more expensive than the deterioration of the component under test. Therefore, monitoring this environment is appropriate, but as the system is new, the algorithms must automatically adapt to the component and to the driver's behavior who wants to experience the system at the edge of its normal domain.

In this paper we present a novelty detection algorithm used in batch mode at the end of each cycle. During a test cycle, the pilot increases the shaft speed by successive steps then finally ends the cycle by an equivalent slow descent. The algorithm takes a summary of the cycle and works at a cycle frequency producing only one result at the end of each cycle. Its goal is to provide an indication to the pilot on the reliability of the bench's use for a next cycle.

## 1. INTRODUCTION

This document follows two previous articles published in 2010 and 2011 in the PHM Society. The first one (Lacaille & Gerez & Zouari, 2010b) presents the health-monitoring architecture we deploy on one of our test benches and gives clues about adaptation to context changes in the use of the machine. We proposed an algorithmic solution using simultaneously an autoadaptive clustering algorithm and local detection tools calibrated on each cluster. In the second paper (Lacaille & Gerez, 2011c) a lighter solution based on similitude computations and nearest neighbor algorithms was given. This implementation was essentially given to be embedded in the FADEC computer of the engine. In fact the algorithms used on test benches are also

good prototypes for online solutions. It's why a fast solution needed to be developed to check if it could also work on dedicated hardware when the engine is installed under an aircraft wing.

Those two previous propositions deal with online abnormality detection: during the execution of the test. They essentially detect fleeting events that suddenly appears without more warning. This paper presents a solution for an off line analysis. The algorithm was already implemented in a lighter form on operational data broadcasted via SatCom (as ACARS messages) to the ground. This limited version of the algorithms was partly described in (Lacaille, 2009c), the current proposition deals with automatic detection of stationary levels, building of temporal snapshots, analysis of the ground database of such snapshots with a clustering algorithm to detect recurrent configurations, and the novelty detection algorithm. Figure 1 shows the OSA-CBM decomposition of each layer of the algorithm.

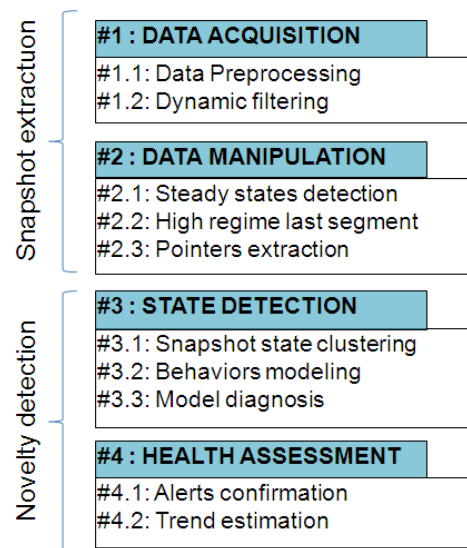


Figure 1: OSA-CBM architecture of the algorithm.

Jérôme Lacaille et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

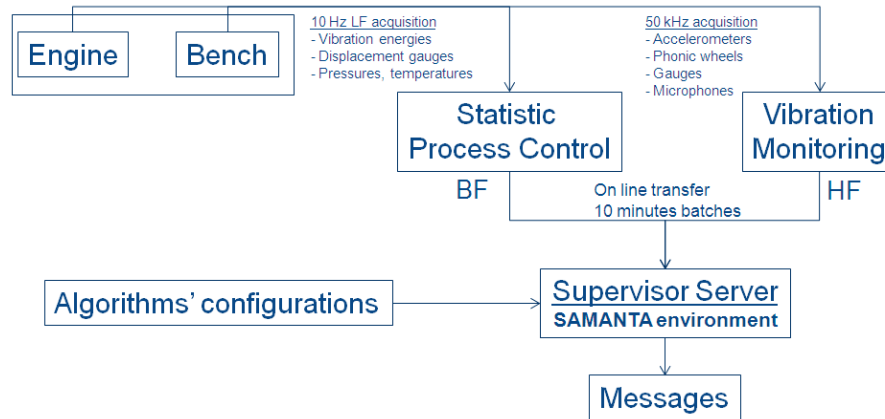


Figure 2: Deployment of the PHM system on a distinct server.

The health-monitoring algorithms are developed by Snecma under the SAMANTA platform which was previously described in (Lacaille, 2009b). This environment industrializes blocs of mathematic processing tools in graphical units. Aeronautic engineers are able to exploit each mathematic module to build their own specific solutions. In our case this method uses signal filters to prepare the data, a stationary process detector, a clustering tool, some regression and dimension reduction algorithms to help normalize observations and make them as much independent of the acquisition context (bench cell driving) as possible. For the novelty detection part, it uses a score computation, a threshold based configurable statistic test and a diagnosis confirmation tool. The two first OSA-CBM layers (mostly signal filtering) are assembled apart from the rest of the algorithm. They produce an off line database of snapshots which is processed by the learning phase of the analysis layer (#3). The results from the data-driven part and the statistic test in last layer (#4) produce the diagnosis.

SAMANTA platform embed also some automatic validation layers (see Lacaille, 2010a and 2012) that helps compute key performance indicators (with precision) using cross-validation schemes.

## 2. CONTEXT OF APPLICATION

A turbofan development test bench is subject to lots of changes in behavior. The interaction between pilot and engineers is really tense and the system may be stopped at any instant if some analyst finds an abnormality in the observations. The sensors are directly broadcasted to observations consoles and validated numeric solutions may launch alarms. The health-monitoring goal is not to stop the process but to provide information about the health of the test bench itself (or the tested engine part, but we give a lot more attention to the bench which is more expensive and less damageable than the tested prototypes).

### 2.1. Implementation in the test cell

To minimize interactions between the driving of the system and the PHM algorithms we implement an execution driver of our SAMANTA platform on a separate server with a local memory buffer able to deal with days of high frequency acquisition data (50kHz) and weeks of low frequency acquisition (10Hz) and enough storage space to manage a big database of snapshots (some data vectors per cycle – with one or two cycles each day).

### 2.2. Reliability computation

The PHM algorithms should present computation results with a minimum of reliability because we don't want to interrupt an expensive test experiment scheduled for weeks or months with bad reasons. Hence a very important attention is given to the false alarm performance indicator (PFA). The other indicator we follow is the probability of detection (POD). It is a lot easier to compute because we have some past logbooks on which all historical events where recorded. The main job in that case was to label those (handwritten) data and to compute the detection rate on past tests.

The PFA indicator is given by Eq. (1). If one writes  $P(Detected)$  the probability that an abnormality is detected by the algorithm,  $P(Healthy)$  the probability that the system is healthy. Then the false alarm rate is just the probability that the system is healthy but that an abnormality is detected. It is represented as the following conditional probability:

$$PFA = P(Healthy|Detected) \quad (1)$$

This is clearly different from the usual  $\alpha = P(Detected|Healthy)$  which is the first species statistic error that one needs to calibrate to define the test rejection domain. This PFA value really represents the inconvenience of stopping a test for no reason.

The probability of detection is simply given by Eq. (2):

$$POD = P(Detected|Faulty) \quad (2)$$

It is the standard  $1 - \beta$  value, usually called “test power” in statistical background.

PFA can also be rewritten according to Bayes’ rule and a computation of  $\alpha$  in Eq. (3) shows that the test threshold should be chosen very far from normal behavior when one intends to respect a small boundary constraint on the false alarm rate:

$$\alpha = POD \cdot \frac{PFA}{1 - PFA} \cdot \frac{\rho}{1 - \rho} \approx POD \times PFA \times \rho \quad (3)$$

where  $\rho = P(Faulty)$  is usually very small for aircraft engine parts (less than  $10^{-6}$  per hour). A very careful attention is needed for the choice of this detection threshold. It is why the decision part of the algorithm has two additional modules: one for confirmation by several successive detections and another for the optimization of the threshold, using a model of the score distribution queue with Parzen windows.

Figure 3 shows the meaning of the rejection threshold computed from a choice of  $\alpha$  and the power  $1 - \beta$  of a statistical test.

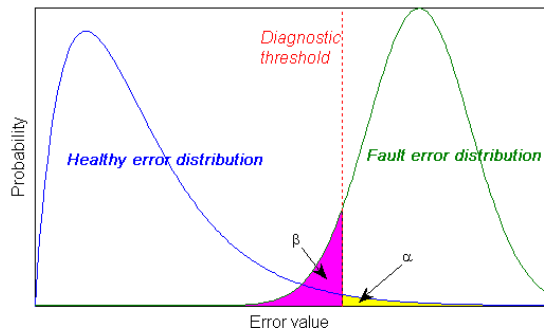


Figure 3: Threshold selection for decision test.

### 2.3. Description of the bench data

The main element we have to monitor is the rotating shaft and the principal bearing (called #4 here). One of the exogenous information we have to deal with is the external loading applied on the right of this shaft. This is a longitudinal force which have a lot of influence on the system behavior because it may change the dynamic mode positions.

Most measurements come from dynamic high frequency acquisitions. The corresponding low-frequency observations are filtered energy computations which may be either piloted by the shaft speed or be total vibration energy, eventually quantified according to given bandwidths.

The table below (Table 1) gives the complete list of used sensors.

Index			
	ESN	Engine serial number	
	CYCL	Engine cycle reference	
	DATE	Date of cycle (start to off)	
N°	Sensor	Label	Unit
01	XN	Shaft rotation speed	tr/min
02	XN_DERIV	Accel. of the rotation	-
03	TORQUE	UI/w	kg/h
04	P4	Pressure Piston #4	bar
05	PORSDE	Position rectifier	deg
06	VANPRIM	Position primary vane	%
07	P1	Pressure Piston #1	bar
08	K_0N	Disp. Up Pilot	mmDA
09	K_0T	Disp. Up RMS	mmDA
10	K_1N	Disp. Down Pilot	mmDA
11	K_1T	Disp. Down RMS	mmDA
12	ACC_4RN	Accel. #4 Rad Pilot	cm/s eff
13	ACC_4RT	Accel. #4 Rad RMS	cm/s eff
14	T4	Temp. #4	degC
15	ACC_1HN	Accel. #1 Horiz Pilot	cm/s eff
16	ACC_2VN	Accel. #2 Vert Pilot	cm/s eff
17	ACC_3VN	Accel. #3 Vert Pilot	cm/s eff
18	ACC_MN	Accel. Engine Pilot	cm/s eff
19	ACC_1HT	Accel. #1 Horiz RMS	cm/s eff
20	ACC_2VT	Accel. #2 Vert RMS	cm/s eff
21	ACC_3VT	Accel. #3 Vert RMS	cm/s eff
22	T1	Temp. #1	degC
23	ACC_MT	Accel. Engine RMS	cm/s eff
24	T2	Temp. #2	degC
25	T3	Temp. #3	degC

Table 1: List of sensors and corresponding units, blue and green backgrounds identify respectively a selection for the exogenous and endogenous variables.

The #1 to #4 numbers refer to the different bearings where accelerometers are measuring vibration data. Those vibration values are summarized as local energies for a frequency band that corresponds to the shaft speed (N) or a total amount of energy (T). The first sensors (blue background) are used as context information or exogenous variables. The corresponding data vector is used to identify the context of the measurement. They are used to select stationary snapshots and to classify the snapshots into clusters. The others variables (endogenous) are used to monitor the bench when a context is clearly identified.

Other variables such as microphone band energies are not displayed in Table 1. Such selection of endogenous and exogenous variables defines an instance of the algorithm. It is possible to build different kind of instances (with corresponding algorithmic parameters) for any part of the test bench one selects to monitor.

The abnormalities may be very tricky to detect. For example on Figure 4 one can see measurements taken during a test cycle that contains such anomaly. Just looking the data is rarely sufficient to find the abnormal behavior. A mathematic comparative analysis is definitely needed.

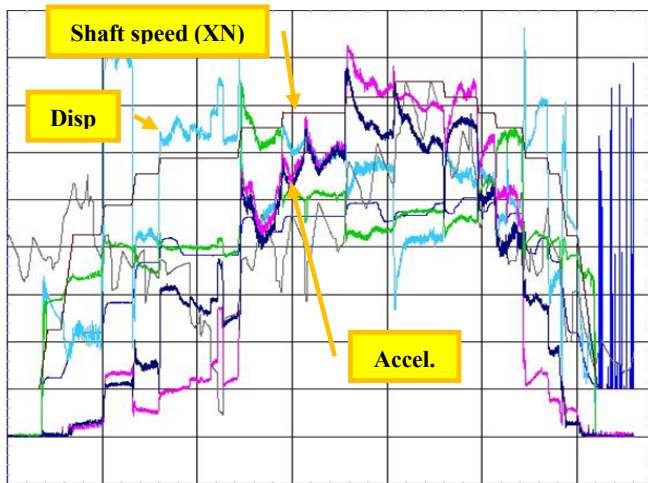


Figure 4: Example of measurements acquired during a whole cycle. The black stepwise line represents the shaft speed. Displacement (light blue and green) and accelerometers (dark blue and magenta) are highlighted. The other blue sensor is the load (the last data have no meaning since the shaft stopped). An abnormality is hidden in those data.

### 3. ALGORITHMS DESCRIPTION

The algorithm is made of two parts. The first one identifies stationary measurement intervals (in context data) and builds a snapshot of the endogenous measurements. The second part loads the database of snapshots, builds clusters, and for each cluster search for abnormalities.

#### 3.1. Snapshots extraction

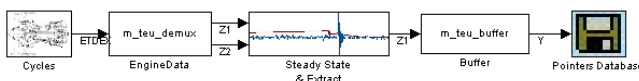


Figure 5: Graph of SAMANTA modules to extract snapshots and build a database.

The first step of snapshot extraction is the selection of measurements to identify stationary data. The stationary measurement detector waits for a main control value to be almost constant and tests a vector of endogenous measurements for second order statistic stationarity. In our case we use the shaft speed as main control and test other endogenous data for stationarity.

Once a stable point detected, a buffer of observations is recorded and defines the snapshot. Figure 6 shows a list of snapshots detected on a symbolic cycle that may represent a real flight.

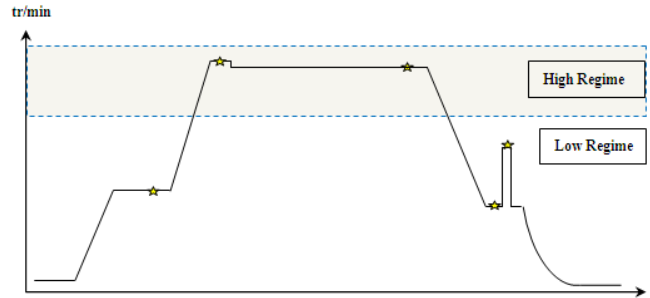


Figure 6: Example of snapshots identification, each star represents a point detected as a possible snapshot for the test cycle.

#### 3.2. Novelty detection

The detection part uses three mathematic models: the clustering algorithm, the score algorithm and the decision algorithm. Each one needs a specific learning phase to calibrate.

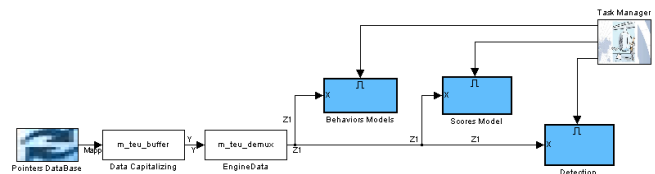


Figure 7: The novelty detection graph of SAMANTA modules.

The **clustering learning phase** uses the whole snapshot database (eventually obtained from a sub-sampling of the snapshot buffers) but only exogenous vectors of values to isolate homogeneous clusters with an EM algorithm. This algorithm, as described in (Lacaille et al. 2010b), is a generative statistic model from a mixture of Gaussian distributions. Each Gaussian identifies a different set of snapshots. The number of classes is estimated by a BIC criterion and the unclassified snapshots are not used.

During the learning phase a database of snapshot buffers is used to define the individual classes which can eventually further be labeled as flight regimes or operating modes. To make this possible, each buffer signal curve is compressed into a set of shape indicators hence replacing the multivariate temporal signal by a vector of indicators  $U$ . The compression scheme (Figure 8) uses specific algorithms to enlighten changes in the data: for examples an algorithm computes the trend of the signal, another looks for jumps and a generic compression uses automatic templates built from a principal component analysis (PCA).

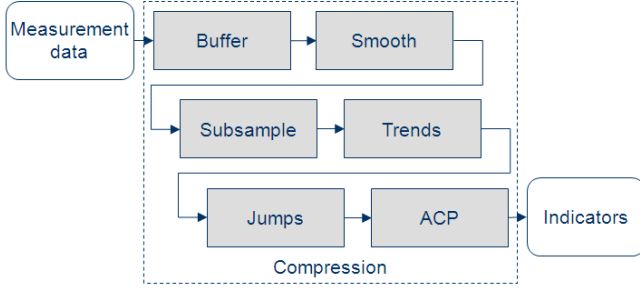


Figure 8: Compression process that builds indicators from a multivariate temporal buffer.

The set of exogenous indicator vector  $U$  is then used by the classification algorithm to build classes as a mixture of Gaussians. The number of classes to build is controlled by a BIC criterion but may be also given by expert bounds as the snapshots are essentially identified as standard layers.

Each set of classed snapshots (the ones that belong to a cluster) are used to **calibrate a score model**. Once identified in a specific class, each multivariate temporal signal of endogenous data is compressed locally in another indicator vector  $Y$  (Figure 9). The score process has two steps; the first one normalizes the endogenous data suppressing disparities due to little variations in the context. This is done by a regression algorithm controlled by a  $L_1$  criterion (LASSO algorithm) as described in (Lacaille & Côme, 2011b). The second step is a model of the residual of this regression by a Gaussian score (a Mahalanobis distance) see (Lacaille, 2009c).

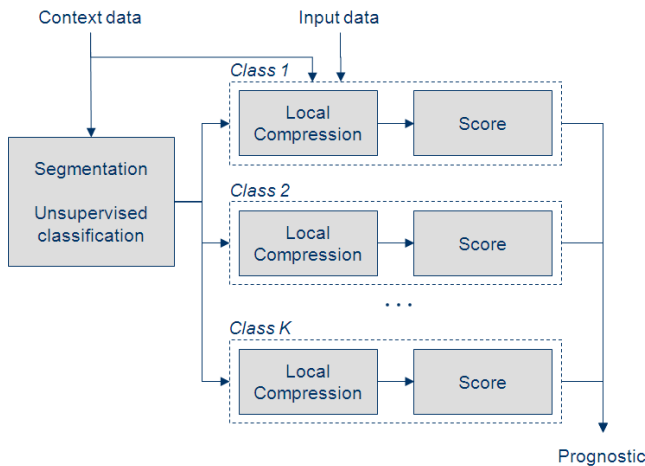


Figure 9: Score analysis after snapshot identification.

Equation 4 explains the mathematics of the regression model. Each parameter  $X_{i,j}$  correspond to a combination  $j$ , non necessary linear, of exogenous variables in  $U$  used to predict endogenous observation  $Y_{i,k}$  of parameter  $k$  (or a

function of endogenous parameters) for all snapshots  $i$  of a given cluster.

$$\text{Arg min } \sum_i (Y_{i,k} - \sum_j \lambda_j X_{i,j})^2 \quad \text{subject to } \sum_j |\lambda_j| < C \quad (4)$$

The final and optimal constant  $C$  is chosen such that the generalization error of the regression is the smallest. The generalization error is computed by a cross validation scheme. The next graph (Figure 10) explains in 2 dimensions why the constraint is to be chosen in absolute value instead of Euclidian norm. The figure schematizes the mean square regression coefficient as the point  $\beta_0$  and iso-square errors as red ellipses (first part of equation 4). The blue shapes (disk for  $L_2$  constrain and square for  $L_1$  constraint) represents the value of the second part of equation 4. The value  $\beta$  to select is on a tangent intersection of an ellipse (ellipsoid in higher dimension) and the surface of the blue shape. The radius of the ellipse is the square error and the radius of the blue shape is the constraint. It appears clearly that with a square (cube, hypercube) the chance to find an edge point of the surface is important. As soon as those points are on the main axis, most of the coefficients of  $\beta$  should be zero.

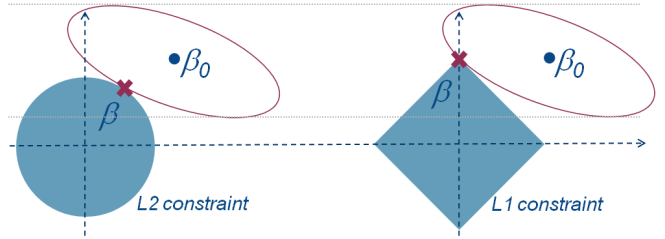


Figure 10: Minimization subject to  $L_1$  constraint (absolute value) instead of a  $L_2$  constraint (Euclidian norm) ensures that most coefficients are set to zero, hence improving the regression robustness with a small loss in mean square error.

As this model automatically selects variables to ensure a good robustness, we may use a big set of computations as inputs. In fact the aeronautic experts give clues about the physic process and help the conception of a big vector  $X$  of indicator functions that makes a great uses of formal physical equations.

The final score  $s$  of a single observation is given by (Eq. 5) where  $r_k = Y_k - \hat{Y}_k$  is the residual of the preceding regression for a variable  $k$  and a current snapshot and  $\Sigma$  is the matrix of covariance of the vector  $r = (r_k)$ .

$$s = r' \Sigma^{-1} r \quad (5)$$

The score should follow a standard  $\chi^2$  distribution under very restrictive constraints on the model and residual distribution. In general those constraints are not completely respected, and in any cases as the real dimensionality of the problem stays approximate, an indetermination of the freedom level is possible. We use a last module of algorithm to establish a more precise **decision rule**. This decision rule

is not only based on one score computation but on a small list of successive scores. Each individual score is compared to a threshold  $\alpha$  and the Boolean results are combined together by a vote process. The law of the scores is modeled with Parzen windows on healthy observations. Then artificial abnormal behaviors are produced with the help of aeronautic experts and a mechanical physic simulator. Using the empirical distribution of scores and the pseudo abnormal observations one is able to determine a good choice for the threshold for a given  $\alpha$  computed from equation 3.

#### 4. RESULTS AND CONCLUSION

Two campaigns of measurements were done on the same bench test cell. The first one was to challenge a civilian compressor and lasted almost 3 months. It was a bench calibration test. The second campaign uses a military compressor as an extractor for another development bench, we get also around 3 months of data. In each case, when the bench was working we may have one to three runs per day. Results cannot be presented in this article; the main goal for the PHM team was to validate the algorithm and the monitoring process.

##### 4.1. Examples of detected abnormalities

**Example 1.** Normally, during a stabilized step, if the axial pressure on the shaft increases, the vibration level should decrease. It was not the case on Figure 11 and this was detected as an abnormal feature.

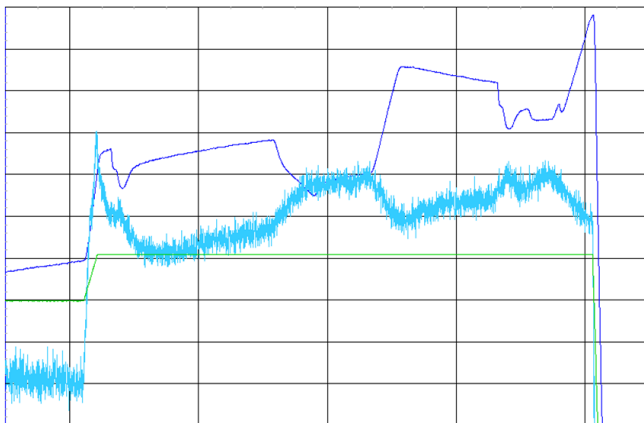


Figure 11: On that test run the vibration level, measured by an accelerometer (light blue), decreases when the pressure augments (dark blue) and during a stabilized level (XN in green).

**Example 2.** Figure 12 shows a sudden increase of the total vibration levels during a deceleration phase. This is also another kind of abnormal behavior which was not detected at first because the system was not able to differentiate between main accelerations and deceleration (begin or end of the run), but as soon as we enter this information as a context observation the detection is fixed.

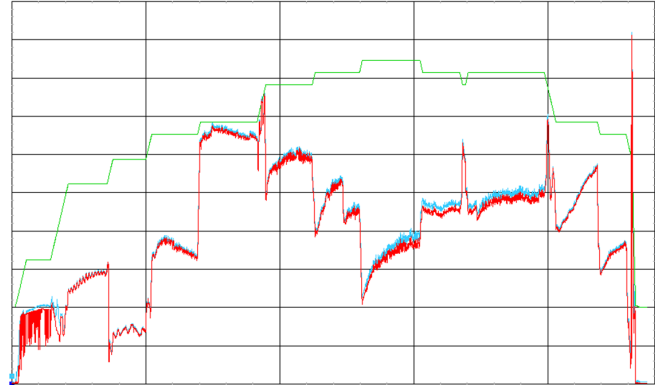


Figure 12: A sudden augmentation of vibration was detected by vibration sensors (red and blue) during a deceleration (shaft speed in green).

##### 4.2. Conclusion

The work was tested on data obtained during the 6 months of experimentation. The PFA indicators were computed on observed data with the prerequisite identification of all the real abnormalities referenced in the logbook. Then new simulated defects compatible with the real observations were artificially added to the data at known random positions on the signals. A false alarm rate of PFA=1.3% and a detection rate of POD=78% were obtained on high speed rotation clusters. On low speed clusters, too much variability of the endogenous variables was observed to give conclusive results. More work will be done on the identification of specific recurrent clusters, but in any cases the algorithm may still be used if some standard states are defined, and if one asks the pilot to reach those states at the beginning of each run.

Another model should also be defined for non-stationary measurements because some known difficulties may arise when the bench crosses a vibration mode during a transient phase such as acceleration or deceleration.

##### NOMENCLATURE

<i>BIC</i>	Bayesian Information Criterion
LASSO	Least Absolute Shrinkage and Selection Operator
FADEC	Full Authority Digital Engine Control
OSA-CBM	Open Systems Architecture for Condition-based Maintenance
<i>PFA</i>	Probability of False Alarm
<i>POD</i>	Probability Of Detection
<i>SAMANTA</i>	Snecma Algorithm Maturation And Test Application

##### REFERENCES

Bellas, A. & Bouveyron, Ch. & Cottrell, M. & Lacaille, J. (2012), *Robust Clustering of High Dimensional Data*. In proceedings of ESANN, Bruges, Belgium.



- Blanchard, S. & Lacaille, J. & Cottrell, M. (2009), *Health monitoring des moteurs d'avions*. In « Les entretiens de Toulouse », France.
- Cômes, E. & Cottrell, M. & Verleysen, M. & Lacaille, J. (2010a), *Aircraft engine health monitoring using self-organizing maps*. In proceedings of ICDM, Berlin, Germany.
- Cômes, E. & Cottrell, M. & Verleysen, M. & Lacaille, J. (2010b), *Self organizing star (sos) for health monitoring*. In proceedings of ESANN, Bruges, Belgium.
- Cômes, E. & Cottrell, M. & Verleysen, M. & Lacaille, J. (2011), *Aircraft engine fleet monitoring using Self-Organizing Maps and Edit Distance*. In WSOM proceedings, Espoo, Finland.
- Cottrell, M. & Gaubert, P. & Eloy, C. & François, D. & Hallaux, G. & Lacaille, J. & Verleysen, M. (2009), *Fault prediction in aircraft engines using self-organizing maps*. In proceedings of WSOM, Miami, FL.
- Flandrois, X. & Lacaille, J. (2009), *Expertise transfer and automatic failure classification for the engine start capability system*. In proceedings of AIAA Infotech, Seattle, WA.
- Hazan, A. & Verleysen, M. & Cottrell, M. & Lacaille, J. (2010a), *Trajectory clustering for vibration detection in aircraft engines*. In proceedings of ICDM, Berlin, Germany.
- Hazan, A. & Verleysen, M. & Cottrell, M. & Lacaille, J. (2010b), *Linear smoothing of FRF for aircraft engine vibration monitoring*. In proceedings of ISMA, Louvain.
- Hazan, A. & Lacaille, J. & Madani, K. (2012), *Extreme value statistics for vibration spectra outlier detection*. In proceedings of CM-MFPT, London, UK.
- Klein, R. (2009), *Model based approach for identification of gears and bearings failure modes*. In proceedings of PHM Society Conference, San Diego, CA.
- Lacaille, J. & R. Nya-Djiki, R. (2009), *Model based actuator control loop fault detection*. In proceedings of Euroturbo Conference, Graz, Austria.
- Lacaille, J. (2009a), *An automatic sensor fault detection and correction algorithm*. In proceedings of AIAA ATIO, Hilton Head, SC.
- Lacaille, J. (2009b), *A maturation environment to develop and manage health monitoring algorithms*. In proceedings of PHM Society Conference, San Diego, CA.
- Lacaille, J. (2009c), *Standardized failure signature for a turbofan engine*, In proceedings of IEEE Aerospace Conference, Big Sky, MT.
- Lacaille, J. (2010a), *Validation of health-monitoring algorithms for civil aircraft engines*, In proceedings of IEEE Aerospace Conference, Big Sky, MT, 2010.
- Lacaille, J. & Gerez, V. & Zouari, R. (2010b). *An Adaptive Anomaly Detector used in Turbofan Test Cells*. In proceedings of PHM Society Conference, Portland, OR.
- Lacaille, J. & Côme, E. (2011a), *Visual Mining and Statistics for a Turbofan Engine Fleet*. In Proceedings of IEEE Aerospace Conference, Big Sky, MT.
- Lacaille, J. & Côme, E. (2011b), *Sudden change detection in turbofan engine behavior*. In proceedings of the 8th International Conference on Condition Monitoring and Machinery Failure Prevention Technologies, Cardiff, UK.
- Lacaille, J. & Gerez, V. (2011c), *Online Abnormality Diagnosis for real-time Implementation on Turbofan Engines and Test Cells*. In proceedings of PHM Society Conference, Montreal, Canada.
- Lacaille, J. (2012), *Validation Environment of Engine Health Monitoring Algorithms*. In proceedings of IEEE Aerospace Conference, Big Sky, MT.
- Nya-Djiki, R. & Hezard, E. & Lacaille, J. (2012), *Enhanced Massive Visualization of Engines Performance*. In proceedings of COMADEM, Huddersfield, UK.
- Seichepine, N. & Ricordeau, J. & Lacaille, J. (2011), *Datamining of flight measurements*. In proceedings of AIAA@Infotech, Saint Louis, MO.

#### BIOGRAPHY



**Jérôme Lacaille** is a Safran emeritus expert which mission for Snecma is to help in the development of mathematic algorithms used for the engine health monitoring. Jérôme has a PhD in Mathematics on “Neural Computation” and a HDR (habilitation à diriger des recherches) for “Algorithms Industrialization” from the Ecole Normale Supérieure (France). Jérôme has held several positions including scientific consultant and professor. He has also co-founded the Miriad Technologies Company, entered the semiconductor business taking in charge the direction of the Innovation Department for Si Automation (Montpellier - France) and PDF Solutions (San Jose - CA). He developed specific mathematic algorithms that were integrated in industrial process. Over the course of his work, Jérôme has published several papers on integrating data analysis into industry infrastructure, including neural methodologies and stochastic modeling.



**Valerio Gerez** is a mechanical engineer who works for Snecma since 1982. He has almost 30 years of experience in aircraft engines in the areas of quality and especially in Engine dynamics, both in test cells and in Aircrafts. In 2006, he joined the Diagnostic and Prognostic Department and now manages R&D HM projects for Snecma future applications and the deployment of algorithms in test cells.

# Spare Parts Inventory Control for Non-Repairable Items Based on Prognostics and Health Monitoring Information

Leonardo Ramos Rodrigues<sup>1</sup> and Takashi Yoneyama<sup>2</sup>

<sup>1</sup>EMBRAER S.A., São José dos Campos, São Paulo, 12227-901, Brazil  
leonardo.ramos@embraer.com.br

<sup>2</sup>ITA – Instituto Tecnológico de Aeronáutica, São José dos Campos, São Paulo, 12228-900, Brazil  
takashi@ita.br

## ABSTRACT

The application of PHM (Prognostics and Health Monitoring) techniques can provide a wide range of benefits to aircraft operators. Since the primary goal of PHM systems is to estimate the health state of components and equipments, as well as forecasting their RUL (Remaining Useful Life), they are often closely associated with the reduction in the number of unscheduled maintenance tasks. Indeed, the avoidance of unscheduled maintenance is a very important factor, but this technology may potentially lead to considerable further savings in other fields. The usage of PHM information by the logistics team for spare parts inventory control is a good example to illustrate that a PHM system can potentially provide benefits for other teams besides the maintenance team. The purpose of this work is to present a comparison between two different inventory control policies for non-repairable parts in terms of average total cost required and service level achieved. The well known  $[R, Q]$  (re-order point, economic order quantity) inventory model will be used as a reference. This model will be compared with a model based on information obtained from a PHM system. Discrete event simulation will be used in order to simulate and assess the performance of both models.

## 1. INTRODUCTION

PHM technology is recognized by the members of the aeronautical sector such as aircraft operators, MRO (Maintenance, Repair and Overhaul) service providers, aircraft manufacturers and OEMs (Original Equipment Manufacturers) as a relevant tool that may lead to important competitive advantages such as reduction in operational cost and increase in fleet reliability. However, quantifying PHM benefits is not a simple task. Hess, Frith and Suarez (2006) stated that cost-benefit models are the key to estimate the value of PHM technology.

In order to demonstrate the benefit of PHM technologies, many cost-benefit models have been proposed (Hess et al., 2006; Luna, 2009; Sandborn & Wilkinson, 2007; Feldman, Jazouli & Sandborn, 2009). Some of these works (Luna, 2009; Sandborn & Wilkinson, 2007) comprise discrete-event simulation models.

The objective of these models is to simulate the behavior of the maintenance or logistics departments when a PHM system is available for a set of components. Such models can be divided basically into three blocks: Fleet simulation, decision making and cost evaluation (Rodrigues, Gomes, Bizarria, Galvão & Yoneyama, 2010). Figure 1 shows how each simulation block interacts with others.

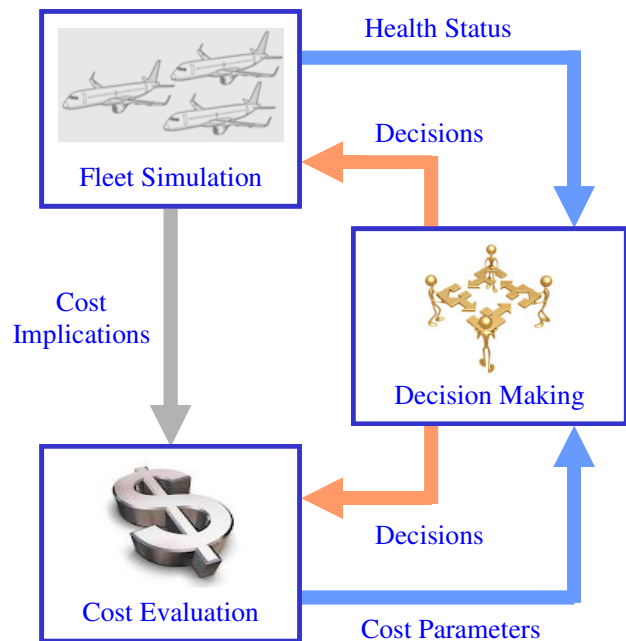


Figure 1. Simulation block diagram

Rodrigues, L. R. et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

The fleet simulation block comprises a simulation of a set of components that are monitored by a PHM system. Failures are inserted in the simulation based on historical rates and the PHM system is assumed to anticipate a given number of failures by providing PDFs (Probability Density Functions) of failure instant.

Based on that information, the decision making block is responsible for defining which actions should be taken in a certain moment in time. The logic of the decision making block depends on the activities covered by the simulation. When the maintenance planning is simulated, the decision making block can define the best time and the best location to perform maintenance. If logistics department is simulated, the decision making block can define the best moment to place a new purchase order for spare parts and the ideal number of spare parts to be acquired.

The cost evaluation block computes the total cost resulting from the actions taken during the simulation. Usually, this cost is compared to the cost obtained by using a conventional model in order to quantify the benefits due to the PHM system.

This work presents a spare parts inventory control policy for non-repairable items. The proposed policy is based on the health condition information obtained from a PHM system. A discrete event simulation is performed in order to compute the costs associated with the implementation of the proposed method. A comparison between the proposed method and the classical  $[R, Q]$  inventory control model is made in terms of average total cost required and service level achieved.

## 2. PHM BASIC CONCEPTS

PHM can be defined as the ability of assessing the health state, predicting impending failures and forecasting the expected RUL of a component or system based on a set of measurements collected from the aircraft systems (Vachtsevanos, Lewis, Roemer, Hess & Wu, 2006). It comprises a set of techniques which use analysis of measurements to assess the health condition and predict impending failures of monitored equipment or system.

The main goal of a PHM system is to estimate the health state of the monitored equipment and forecast when a failure is expected to occur (Roemer, Byington, Kacprzynski & Vachtsevanos, 2005). In order to accomplish this task, it is necessary to collect a set of data from the aircraft. The choice of the parameters that will be recorded is based on the type of equipment/system to be monitored (hydraulic, electronic, mechanic, structural, etc.) and the failure modes that shall be covered by the PHM system. These factors also guide the data collection specification (sample rate, flight phase, etc.).

A health monitoring algorithm must be developed for each monitored system. Each algorithm processes the relevant data and generates a degradation index that indicates how degraded the monitored system is. A degradation index can be generated for each flight leg or for a defined period of time (a day, a week, etc.).

In many cases it is possible to establish a threshold that defines the system failure. When the failure threshold is known, it is possible to extrapolate the curve generated by the evolution of the degradation index over time and estimate a time interval in which the failure is likely to occur (Leão, Yoneyama, Rocha & Fitzgibbon, 2008; Kacprzynski, Roemer & Hess, 2002). This estimation is usually represented as a probability density function, as illustrated in Figure 2. Due to the operational characteristics of some equipment – such as tires and the braking system – it can be useful to express the remaining useful life in terms of flight cycles. There is always a confidence level associated with the predicted time interval.

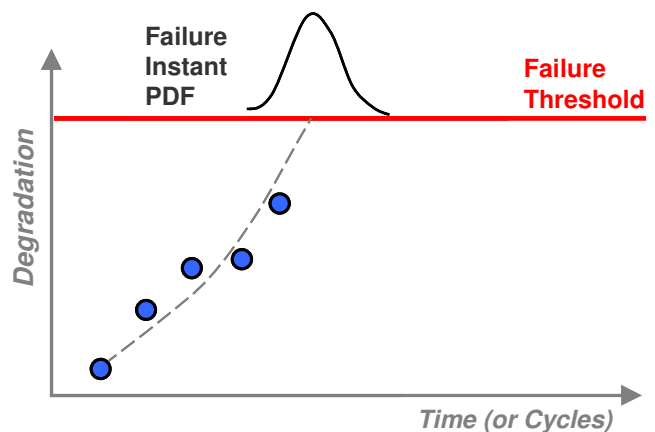


Figure 2. Degradation evolution and instant of failure prediction

The term IVHM (Integrated Vehicle Health Management) is commonly used when the information generated by a PHM system is used as decision support tool. IVHM can be defined as the ability of making appropriate decisions in both strategic and tactic levels based on diagnostics/prognostics information, available resources, logistics information and operational demand in order to optimize the efficiency of operation (Puttini, 2009).

### 3. CLASSICAL INVENTORY MODELS

Failure events occur during fleet operation and spare parts must be available to keep aircraft flying. To deal with these failure events and avoid AOG (Aircraft on Ground) situations, aircraft operators have to maintain a spare parts inventory. An inventory control program must be implemented in order to fulfill the highest possible number of spare parts demand at the least possible cost.

There is a set of classical inventory control models described in literature that can be used to establish an inventory policy. Most of these models define an inventory policy based on total cost minimization. Inventory cost can be broken down into the following factors (Ballou, 2006; Hillier & Lieberman, 2005):

#### Ordering Cost

The cost of ordering an amount of  $Q$  spare parts is presented in Eq. (1). It is composed by two main components: The acquisition cost  $C$  (directly proportional to the amount ordered) and a constant term  $K$  representing the administrative cost of placing a new purchase order.

$$OC(Q) = K + C \cdot Q \quad (1)$$

Where  $OC$  is the ordering cost,  $Q$  is the number of spare parts to be purchased,  $K$  is the administrative cost and  $C$  is the unit cost.

#### Holding Cost

Holding cost is also known as storage cost and represents the aggregated cost related to the storage of the inventory until it is used. It comprises the cost of capital tied up, warehouse space leasing, insurance, obsolescence, protection, inventory management labor, etc.

The holding cost can be computed either continuously or on a period-by-period basis. In the latter case, the cost may be a function of the maximum quantity held during the period, the average amount held, or the quantity in inventory at the end of the period. If holding cost is computed continuously, it can be calculate as indicated in Eq. (2).

$$HC = \int H \cdot X(t)dt \quad (2)$$

Where  $HC$  is the holding cost,  $H$  is the holding cost per unit per unit of time held in inventory and  $X(t)$  is the number of spare parts held in inventory at instant  $t$ . In some cases,  $H$  is defined as a fraction of the unit cost.

#### Stockout Cost

Stockout cost is the cost of not having a spare part on hand when it is needed. In the event of a failure, if the failed component cannot be replaced due to the lack of a spare part, it may result in an AOG situation. In this case, the stockout cost represents the losses in the aircraft operator's revenue related to the aircraft unavailability.

Some aircraft operators outsource the spare part inventory management. In this case, the stockout cost is also known as penalty cost and represents possible contractual penalties for the inventory owner. Indirect costs such as company reputation and damage to customer relationship may also be included as part of the stockout cost.

Inventory simulation can adopt two different scenarios for stockout costs. In the first scenario, when a spare part is required and there are no spares on hand, the aircraft with the failed component remains unavailable until the inventory is replenished and demand for the failed component can be satisfied. This scenario is called backlogging.

In the second scenario, when a spare part is required and there are no spares on hand, the inventory is no longer responsible for satisfying that specific demand. In this scenario, this demand is considered to be met by a priority shipment. This scenario is called no backlogging.

In this work, the backlogging scenario is considered. The stockout cost is given by:

$$SC = \int P \cdot Y(t)dt \quad (3)$$

Where  $SC$  is the stockout cost,  $P$  is the stockout cost (or penalty cost) per unit per unit of time and  $Y(t)$  is the number of spare parts requests not satisfied by the inventory at instant  $t$ . Sometimes,  $P$  is defined as a fraction of the unit cost.

#### 3.1. Deterministic Models and Stochastic Models

Inventory models can be divided in two categories: Deterministic models and stochastic models, according to whether the demand for a specific period is known or is a random variable having a known probability distribution (Hillier & Lieberman, 2005).

Deterministic inventory models are used when the demand for future periods can be forecast with good precision. An inventory policy can be developed in order to satisfy all spare parts requests.

On the other hand, when it is not possible to forecast future demand with acceptable precision, stochastic inventory models are used. These models assume that future demand is a random variable having a known probability distribution. The inventory policy is designed based on the service level desired. Service level is the percentage of spare parts requests that are satisfied immediately.

In this work, demand is considered to be stochastic with a normal distribution.

### 3.2. Continuous Review and Periodic Review

Another common classification for inventory control models refers to whether the current inventory level is monitored continuously or periodically (Ballou, 2006).

In continuous review models, a reorder point is defined as the quantity that triggers the need for a new order. Then a new order is placed as soon as the stock level falls down to the reorder point.

In periodic review models, a maximum inventory level is defined and the current inventory level is checked at discrete intervals, e.g., at the end of each week or month. A new order is placed every time the inventory level is checked in order to replenish it to its maximum value.

In this work, the inventory level will be continuously monitored.

### 3.3. The [R, Q] Model

A continuous review inventory policy for a specific component normally will be based on two critical numbers: The reorder point ( $R$ ) and the order quantity ( $Q$ ). That is the reason for calling this the  $[R, Q]$  model. In this model, whenever the effective stock level of the component drops to  $R$  units, an order for  $Q$  more units is placed to replenish the inventory. The effective stock is the total of spare parts in the warehouse and replenishments ordered but not yet received.

In this work, a  $[R, Q]$  model will be simulated and the results will be compared with the results obtained when the proposed model used. The assumptions of the  $[R, Q]$  model used in this work are described as follows:

- Each  $[R, Q]$  model establishes the policy for a single component.
- The inventory level is under continuous review.
- There is a lead time between when the order is placed and when the order quantity is received. This lead time is considered to be fixed.

- The demand for withdrawing units from inventory during the lead time is uncertain. However, the probability distribution of demand is known.
- If a stockout occurs before the order is received, the excess demand is backlogged, so that the backorders are filled once the inventory is replenished.
- A fixed administrative cost  $K$  is incurred each time an order is placed (as described in Eq. (1)).
- There is no discount for large quantity order.
- A certain holding cost  $H$  is incurred for each unit in inventory per unit time.
- When a stockout occurs, a stockout cost  $P$  is incurred for each unit backordered per unit time until the backorder is filled.

To simulate an inventory model based on this policy, the only decisions to be made are to choose  $R$  and  $Q$ . The expression used to calculate  $Q$  is the EOQ (Economic Order Quantity) formula (Hillier & Lieberman, 2005):

$$Q = \sqrt{\frac{2 \cdot D \cdot K \cdot (H + P)}{H \cdot P}} \quad (4)$$

Where  $Q$  is the quantity of spare parts to be purchased when a new order is placed,  $D$  is the average demand per unit of time,  $K$  is the administrative cost of placing an order,  $H$  is the holding cost per unit per unit of time held in inventory and  $P$  is the stockout cost per unit per unit of time.

The reorder point  $R$  is determined based on the desired service level ( $SL$ ). In this model, service level is related to the probability that a stockout will not occur between the time an order is placed and it is received (called lead time).

A managerial decision needs to be made on the desired service level. Since the demand probability distribution is known,  $R$  is chosen so that the area under the demand curve is at least equal to the defined service level. The procedure to determine  $R$  is illustrated in Figure 3.

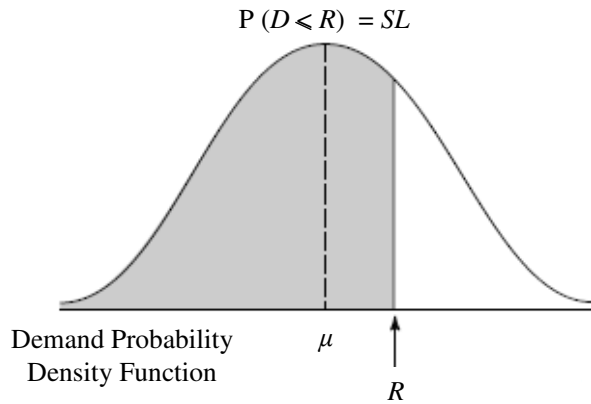


Figure 3. Reorder point definition for the  $[R, Q]$  model

### 3.4. Extensions of the EOQ Model

Several works exploring the EOQ formula and some extensions have been published by the Operational Research community over the last decades (Syntetos, Boyland & Disney, 2009). In most of these works, continuous demand is considered (Yong, Ying & Bing, 2011). Other techniques such as Lot for Lot Ordering (Omar & Supadi, 2003), Wagner-Within Algorithm (Wagner & Whitin, 1958), Least Period Cost Model (Ho, Chang & Solis, 2006) and Silver-Meal Algorithm (Omar & Deris, 2001) are also applied to deal with discrete demand problems.

Although demand for spare parts presents characteristics similar to a discrete pattern, many studies consider the assumption that spare parts demand is continuous and apply the EOQ formula (Sakaguchi & Kodama, 2009). It happens because the EOQ model is very easy to understand and simple to implement, while most of techniques developed to deal with discrete demand are complex and hard to implement.

Wongmongkolrit and Rassameethes (2011) proposed a modification to the EOQ model in order to adapt it to be used in discrete demand problems.

### 4. PROPOSED MODEL

This section describes the proposed model to control the spare parts inventory for a non-repairable item. All assumptions listed on section 3.3 for the  $[R, Q]$  model are valid for the proposed model, which also considers the following assumption:

- The proposed model receives information from a PHM system that systematically monitors the health status of the items installed on the fleet.

It can be noticed that the set of assumptions considered by the proposed model is very similar to the set of assumptions of the classical  $[R, Q]$  model presented in the previous section. In fact, the proposed model is essentially a  $[R, Q]$  model, but it differs from the classical  $[R, Q]$  model in how the reorder point  $R$  is calculated.

In the classical model, reorder point  $R$  is fixed. It is only necessary to calculate  $R$  at the beginning of simulation because it does not change unless desired service level or demand probability distribution change during the simulation.

However, in the proposed model, reorder point is obtained based on the information received from the PHM system. Since in this work PHM information is considered to be updated on a daily basis, the reorder point  $R$  will be also updated at the same rate.

Figure 4 illustrates the procedure to calculate the reorder point  $R$  for the proposed model. Since the reorder point is systematically updated, it will be called  $R(t)$ .

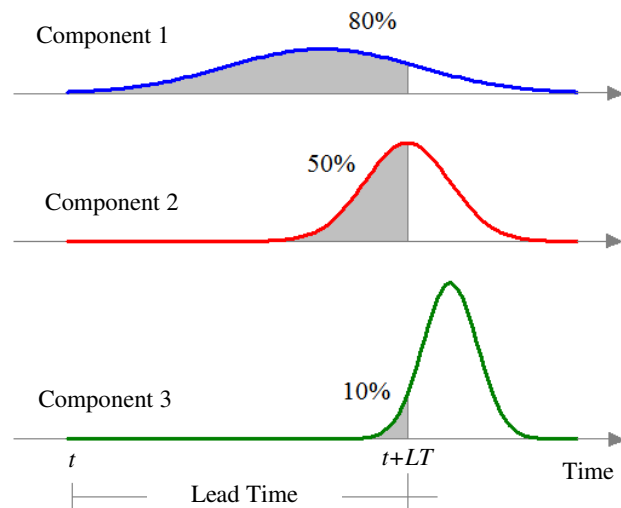


Figure 4. Reorder point definition for the proposed model

Let's assume that  $t$  is the current day and the curves showed in Figure 4 are the instant of failure probability density functions given by a PHM system for three similar components. The reorder point  $R(t)$  must be calculated in order to define whether a new order must be placed or not. An order placed on day  $t$  will be delivered on day  $t+LT$ , where  $LT$  is the lead time. The proposed model will calculate  $R(t)$  based on the probability that each component will fail before instant  $t+LT$ . These probabilities correspond to the gray area under each probability density function in Figure 4.

It can be seen that component 1 has a probability 80% of failing before instant  $t+LT$ . The probabilities for components 2 and 3 fail before instant  $t+LT$  are, respectively, 50% and 10%.

Based on this information, the model calculates the probability of  $N$  components fail before instant  $t+LT$ . For instance, in the example above the probability of all three components to fail before instant  $t+LT$  is obtained by multiplying the probability of each component to fail before instant  $t+LT$  (i.e.  $80\% \times 50\% \times 10\% = 4\%$ ). The complete fail probability table for this example is shown in Table 1.

Number of failures (N)	Probability that exactly N failures will occur before instant $t+LT$	Probability that at most N failures will occur before instant $t+LT$
0	9%	9%
1	46%	55%
2	41%	96%
3	4%	100%

Table 1. Fail probability table

The fail probability table and the desired service level are used to define the reorder point  $R(t)$ . Let's suppose that the desired service level for this example is 95%. The last column on the right in Table 1 shows that if there are 2 spare parts in inventory, there will be a probability of 96% that stockout will not occur. In other words, having 2 spare parts on the inventory corresponds to a service level of 96% (higher than the desired 95%). Since 2 is the lowest number of spare parts that satisfies the service level requirements, the reorder point  $R(t)$  is 2.

The EOQ formula will be used to calculate the number of parts to be purchased in the proposed model. The only difference between the proposed model and the classical  $[R, Q]$  model will be reorder point calculation.

### 5. SIMULATION

The spare part inventory control simulation is described in this section. As mentioned earlier, a comparison will be made between the classical  $[R, Q]$  model described in section 3 and the proposed model based on information obtained from a PHM system described in section 4.

In order to compare the performance of both inventory models, two identical fleets will be simulated. The classical  $[R, Q]$  model will be used to control the spare part inventory of the first fleet, while the other fleet will have its spare part inventory controlled by the proposed model.

### 5.1. Scenario Description

The spare parts logistic network considered in the simulation is illustrated in Figure 5. There is only one supplier and the spare parts are held in only one warehouse.

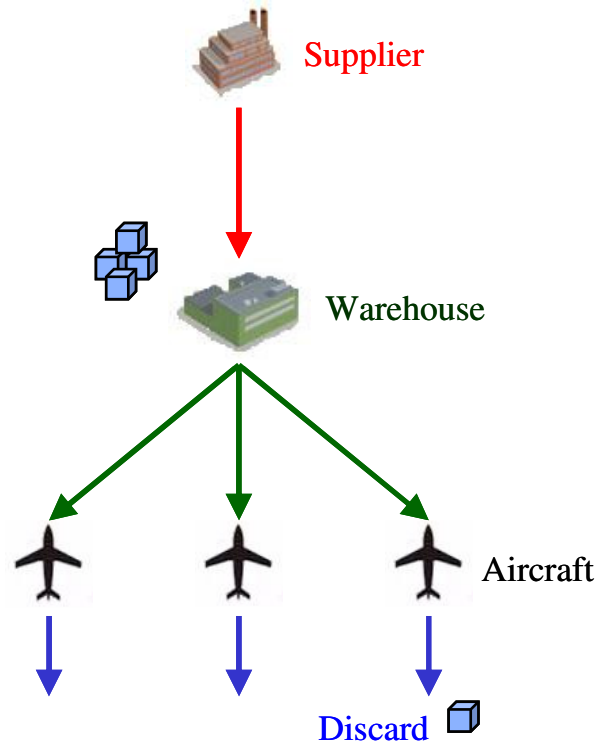


Figure 5. Spare parts logistic network

Spare parts enter the system when a new order is placed. Supplier always delivers the spare parts to the warehouse. Even if there is an aircraft waiting for the part, it is sent to the warehouse and then to the aircraft. There is a lead time between when the order is placed and when the order quantity is received.

When a failure occurs and a component has to be replaced, a spare part is supplied by the warehouse. Since components are considered to be non-repairable, faulty components are discarded.

If a failure occurs and there is no spare parts at the warehouse, the aircraft with the faulty component waits the next spare part delivery.

### 5.2. Simulation Parameters

In order to run the simulation, there are some parameters that must be set. A list of the parameters used during the simulation is shown in Table 2.

Parameter	Value	Unit
Administrative Cost (K)	50	Monetary Units (M.U.)
Unit Cost (C)	500	M.U.
Holding Cost (H)	1	M.U. per day per unit
Stockout Cost (P)	5	M.U. per day per unit
Fleet Size	10	Aircraft
Lead Time	15	Days
MTBF	180	Days
MTBF Standard Deviation	30	Days
Simulation Step	1	Day
Simulated Period of Time	15	Years

Table 2. Simulation parameters

### 5.3. Simulation Results

Five different service levels were defined (80%, 85%, 90%, 95% and 99%) and for each service level 15 simulations were run for each model (classical and proposed).

For the classical model, the economic order quantity  $Q$  is 3 units, calculated using Equation 4. The average demand used to calculate  $Q$  is the fleet size divided by the MTBF

(Mean Time Between Failures). As mentioned earlier, the economic order quantity  $Q$  does not depend on the desired service level.

On the other hand, the reorder point  $R$  changes according to the service level. Figure 6 illustrates an example of how inventory level changed over time for the classical  $[R, Q]$  model during a period of 300 days. In Figure 6, the desired service level is 80% and the calculated reorder point is 0.84 units. In real systems, the reorder point is commonly rounded up. In this work, decimal values were kept.

When the effective stock (dashed green) is lower than the reorder point  $R$  (dotted red), a new order of 3 units is placed. The ordered units are immediately added to the effective stock (dashed green). The stock on hand (solid blue), however, only receives the ordered units after the lead time.

As mentioned earlier, in the proposed model the EOQ formula is used. So, the economic order quantity  $Q$  for the proposed model is also 3 units for all service levels considered in the simulation. The reorder point  $R(t)$  is updated on a daily basis according to the information received from the PHM system. Figure 7 shows an example of how inventory level changed over time for the proposed model during a period of 300 days.

The desired service level in Figure 7 is 80%. When the effective stock (dashed green) is lower than the reorder point  $R(t)$  (dotted red), 3 spare parts are ordered. These spare parts are immediately added to the effective stock and, after the lead time, they are added to the stock on hand (solid blue).

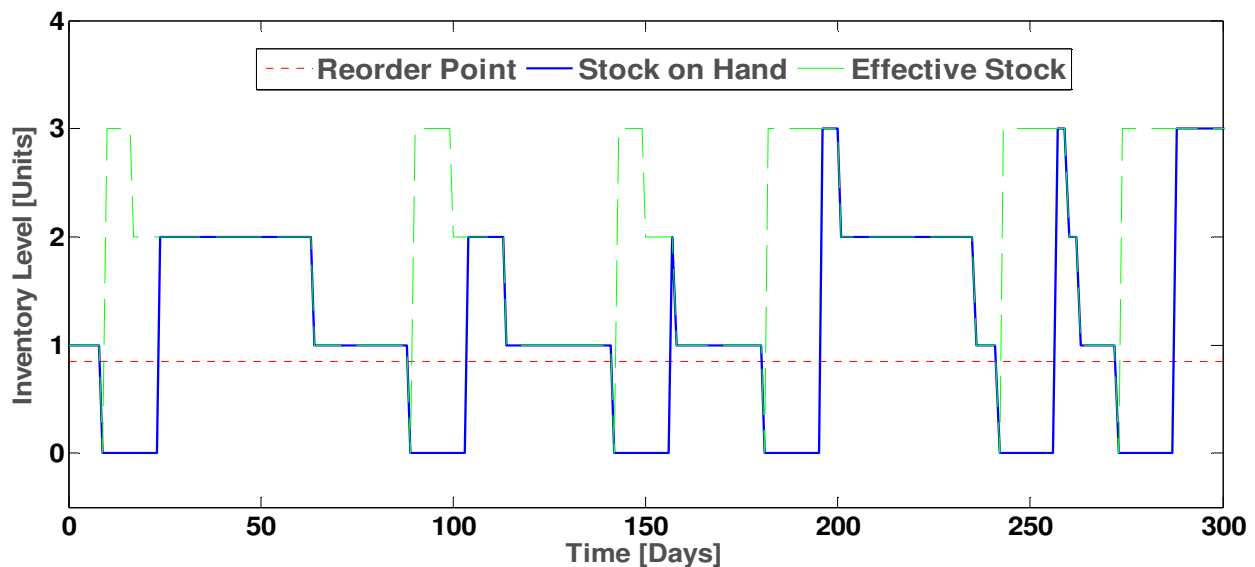


Figure 6. Inventory level evolution for the classical  $[R, Q]$  model



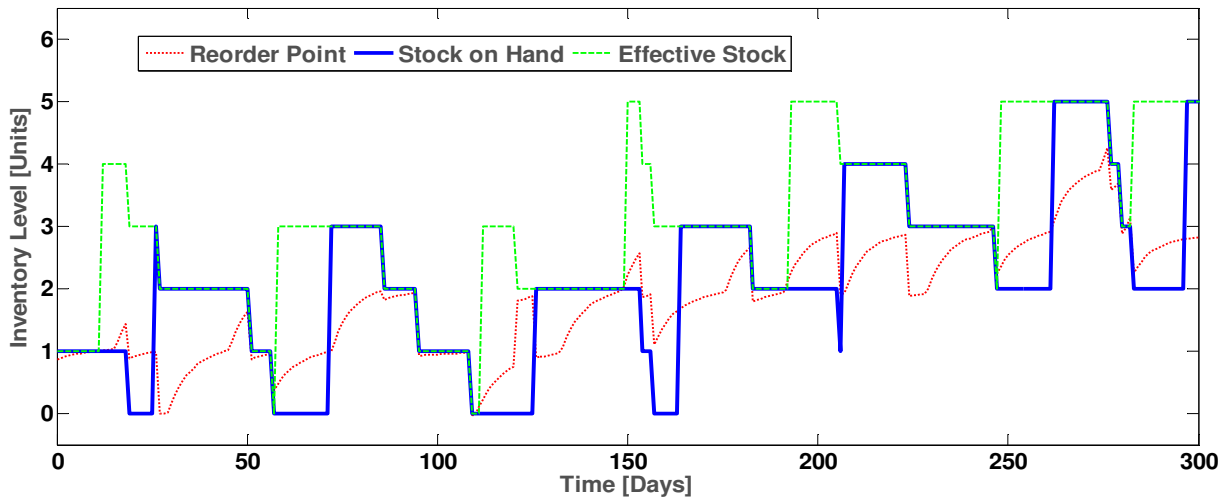


Figure 7. Inventory level evolution for the proposed model

For each simulation, the total cost required and the achieved service level were recorded for both classical and proposed models. Total costs are composed by ordering costs, holding costs and stockout costs. The results are shown in Figure 8.

For all desired service levels, the proposed model presented a lower average total cost. For high service levels, the better performance of the proposed model is more evident.

Figure 9 shows another comparison between the average total cost obtained during simulation of both classical  $[R, Q]$  model and the proposed model, where each cost component (ordering cost, holding cost and stockout cost) can be observed separately. For each service level in Figure 9, the bar on the left shows the average total cost obtained with the classical  $[R, Q]$  model, while the bar on the right shows the average total cost obtained with the proposed model.

For all service levels considered in this work, the average ordering cost obtained with both classical  $[R, Q]$  model and the proposed model were very similar. This result was expected, since the EOQ formula was used by the two models to determine the number of spare parts to be ordered. The average stockout cost values obtained with the two models were also very similar.

On the other hand, when the average holding costs obtained by simulating the two models are compared, it can be noticed that the proposed model allowed reducing this cost component in all service levels considered during the simulation.

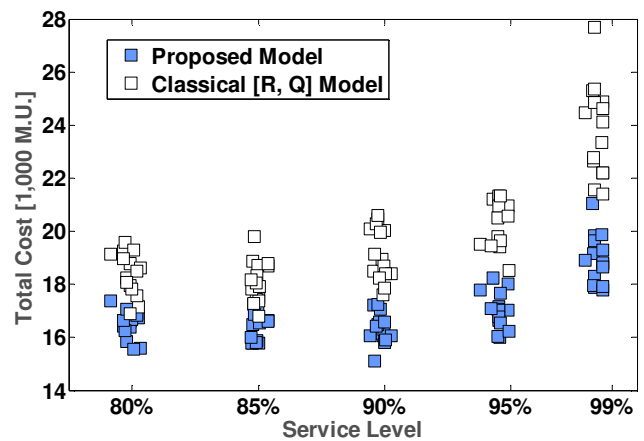


Figure 8. Total cost comparison

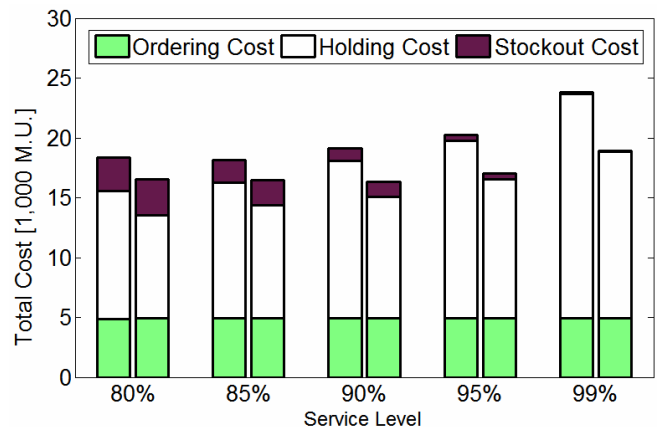


Figure 9. Average total cost components comparison

The proposed model differs from the classical  $[R, Q]$  model only in how the reorder point is calculated. This parameter determines when new spare parts shall be ordered and affects directly the average inventory level and the average holding cost. The information obtained by the PHM system allowed predicting future demands with better accuracy. Since parts are purchased closer to the instant when they will be used, the average period of time they stay in stock is reduced. Consequently, the average holding cost incurred is also reduced when compared to the classical  $[R, Q]$  model.

## 6. CONCLUSIONS

This paper presented a new inventory control model for non-repairable items, based on health condition data obtained from a PHM system.

The results obtained by simulating the proposed model and a classical inventory control model showed that the proposed model allows satisfying a defined service level with a lower average total cost. For high service levels, the proposed model showed itself even more efficient.

Future investigation could extend the idea presented in this paper by adapting the model to be used for repairable parts. Another opportunity to extend this work is to explore the performance of the proposed model when spare part inventories for multiple items are simultaneously controlled.

## ACKNOWLEDGMENT

The authors acknowledge the support of FINEP (Financiadora de Estudos e Projetos - Brazil), CNPq (research fellowship) and FAPESP (grant 2011/17610-0).

## REFERENCES

Ballou, R. H. (2006). Gerenciamento da Cadeia de Suprimentos / Logística Empresarial. In *5th ed. Porto Alegre*.

Feldman, K., Jazouli, T. & Sandborn, P. (2009). A Methodology for Determining the Return on Investment Associated with Prognostics and Health Management. *IEEE Transactions on Reliability, Vol. 58, No. 2*.

Hess, A., Frith, P. & Suarez, E. (2006). Challenges, Issues and Lessons Learned Implementing Prognostics for Propulsion Systems. In *Proceedings of ASME Turbo Expo Power for Land, Sea and Air*.

Hillier, F. S. & Lieberman, G. J. (2005). Introduction to Operations Research. In *8th ed. New York*.

Ho, J. C., Chang, Y. L. & Solis, A. O. (2006). Two Modifications of the Least Cost per Period Heuristic for Dynamic Lot Sizing. *Journal of Operation Research Society, Volume 57*.

Kacprzyński, G. J., Roemer, M. J. & Hess, A. J. (2002). Health Management System Design: Development, Simulation and Cost/Benefit Optimization. In *Proceedings of IEEE Aerospace Conference, Big Sky*.

Leão, B. P., Yoneyama, T., Rocha, G. C. & Fitzgibbon, K. T. (2008). Prognostics Performance Metrics and their Relation to Requirements, Design, Verification and Cost-Benefit. In *Proceedings of International Conference on Prognostics and Health Management, Denver*.

Luna, J. J. (2009). Metrics, Models, and Scenarios for Evaluating PHM Effects on Logistics Support. In *Proceedings of Annual Conference of the Prognostics and Health Management Society*.

Omar, M. & Deris, M. M. (2001). The Silver-Meal Heuristic Method for Deterministic Time-Varying Demand. *Journal of Matematika, Volume 17*.

Omar, M & Supadi, S. S. (2003). A Lot-for-Lot Model with Multiple Installments for a Production System under Time-Varying Demand Process. *Journal Matematika*.

Puttini, L. C. (2009). Gerenciamento da Saúde de Sistemas Aeronáuticos: Conceitos e Visão de Futuro. In *Proceedings of VIII Sitraer, São Paulo*.

Rodrigues, L. R., Gomes, J. P. P., Bizarria, C. O. Galvão, R. K. H. & Yoneyama, T. (2010). Using Prognostic System Forecasts and Decision Analysis Techniques in Aircraft Maintenance Cost-Benefit Models. In *Proceedings of IEEE Aerospace Conference, Big Sky*.

Roemer, M. J., Byington, C. S., Kacprzyński, G. J. & Vachtsevanos, G. (2005). An Overview of Selected Prognostic Technologies with Reference to an Integrated PHM Architecture. In *Proceedings of the First International Forum on Integrated System Health Engineering and Management in Aerospace*.

Sakaguchi, M. & Kodama, M. (2009). Sensitivity Analysis of an Economic Order Quantity for Dynamic Inventory Models with Discrete Demand. *International Journal of Manufacturing Technology and Management, Volume 18*.

Sandborn, P. A. & Wilkinson, C. (2007). A Maintenance Planning and Business Case Development Model for the Application of Prognostics and Health Management (PHM) to Electronic Systems. *Microelectronics Reliability, Volume 47, Issue 12, Electronic system prognostics and health management*.

Syntetos, A. A., Boyland, J. E. & Disney, S. M. (2009). Forecasting for Inventory Planning: A 50 Year Review. *Journal of the Operational Research Society, Volume 60*.

Vachtsevanos, G., Lewis, F. L., Roemer, M., Hess, A., & Wu, B. (2006). Intelligent Fault Diagnosis and Prognosis for Engineering Systems. In *1st ed. Hoboken*.

Wagner, H. M., Whitin, T. M. (1958). Dynamic Version of the Economic Lot Size Model. *Management Science, Volume 5*.

Wongmongkolrit, S. & Rassameethes, B. (2011). The Modification of EOQ Model under the Spare Parts Discrete Demand: A Case Study of Slow Moving Items. In *Proceedings of the World Congress on Engineering and Computer Science, Volume 2, San Francisco*.

Yong, Z. W., Ying, X. & Bing, S. (2011). Study on Spare Parts Inventory Control by Quantitative Analysis in the Environment of ERP System. In *Proceedings of the International Conference of Business Management and Electronic Information (BMEI)*.

## BIOGRAPHIES



**Leonardo Ramos Rodrigues** holds a bachelor's degree in Electrical Engineering from Universidade Federal do Espírito Santo (UFES, 2003), Brazil, and a Master Degree in Aeronautical Engineering from Instituto Tecnológico de Aeronáutica (ITA, 2008), São José dos Campos, São Paulo, Brazil. He is currently

pursuing his doctorate in Aeronautical Engineering at ITA. He is with EMBRAER S.A., São José dos Campos, São Paulo, Brazil, since 2006. He works as a Development Engineer in an R&T group at EMBRAER performing research on PHM technology for application to aeronautical systems. His current research interests are the application of health monitoring techniques for electronic components and the usage of PHM information for inventory optimization.



**Takashi Yoneyama** is a Professor of Control Theory with the Electronic Engineering Department of ITA. He received the bachelor's degree in electronic engineering from Instituto Tecnológico de Aeronáutica (ITA), Brazil, the M.D. degree in medicine from Universidade de Taubaté, Brazil, and the Ph.D. degree in electrical

engineering from the University of London, U.K. (1983). He has more than 250 published papers, has written four books, and has supervised more than 50 theses. His research is concerned mainly with stochastic optimal control theory. Prof. Yoneyama served as the President of the Brazilian Automatics Society in the period 2004-2006.

# Initial Condition Monitoring Experience on a Wind Turbine

Eric Bechhoefer<sup>1</sup>, Mathew Wadham-Gagnon<sup>2</sup>, and Bruno Boucher<sup>3</sup>

<sup>1</sup>*NRG Systems, Hinesburg, VT, 05461, USA*

*erb@NRGSystems.com*

<sup>2,3</sup>*TechnoCentre éolien, Gaspé, Qc, G4X 1G2, Canada*

*mgagnon@eolien.qc.ca*

*bboucher@eolien.qc.ca*

## ABSTRACT

The initial installation of a condition monitoring system (CMS) on a utility scale wind turbine produced a number of unexpected results. The CMS was installed on the TechnoCentre éolien Repower MM92. The installation allowed testing of a MEMS (microelectromechanical system) based sensor technology and allowed in-depth analysis of vibration data and revolutions per minute (RPM) data. A large 3/revolution effect, due to tower shadow and wind shear, required the development of an enhanced time synchronous average algorithm. The ability to easily measure changes in main rotor RPM, as a result of tower shadow and wind shear phenomenology, may also facilitate the detection of icing or blade pitch error.

## 1. INTRODUCTION

NRG System in collaborative partnership with TechnoCentre éolien, installed a CMS on a 2.05 MW Repower MM92 at the TechnoCentre's northern wind energy research location in Rivière-au-Renard. CMS allows operations and maintenance professionals to dramatically lower their costs by accurately predicting when components in the turbine's drive train are likely to fail months in advance. Maintenance and crane calls can then be scheduled at the most optimal time, such as during the low-wind season.

The goal of this collaboration is to validate the performance of a new CMS architecture and gain experience of CMS on a wind turbine. During the validation processes, both raw and processed data is available to NRG System and the TechnoCenter. The CMS installation consisted of:

- A two axis, low speed (1000 mv/g) MEMS

- accelerometer, used for measuring nacelle motion,
- Seven, High Speed (50 mv/g, 0-32 KHz) MEMS accelerometers for drivetrain monitoring,
- One tachometer (smart sensor), and
- A local data concentrator, which provides sensor control, temporary data storage, and Ethernet access.

The CMS was installed on a Repower MM92. This wind turbine uses a three stage, planetary gearbox with a total shaft rate increase of approximately 1:96. Power is generated from a double feed induction machine, allows the main shaft input rates to vary from 7 to 15 revolutions per minute (RPM, or 0.11 to 0.25 Hz). The CMS was configured to perform an acquisition every 10 minutes, and download raw vibration data every 6 hours.

Most industrial accelerometers have an operational range from 2 Hz to 10 KHz. The rates and frequencies associated with many of the gearbox components are below this rate. In order to do analysis on these components, the CMS used MEMS based accelerometers instead of traditional lead zirconate titanate (PZT) based accelerometers. The CMS's MEMS devices have a response to DC, which makes them appropriate for this application.

It was important to validate this CMS on an operational wind turbine in that many of the technologies have never before been implemented in a CMS. For example, the CMS, in addition to validating MEMS accelerometers performance:

- Tested new packaging design using conductive plastics to lower packaging cost of the sensor (approximately 1/7 cost of stainless steel packages),
- Implemented all of the condition indicator (CI) processing and analysis (such as the time synchronous average, gear analysis (residual, energy operator, narrowband/amplitude

E Bechhoefer et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

modulation/frequency modulation) and bearing envelope analysis) on the sensors themselves.

- Implemented a smart tachometer sensor which passed zero crossing data to all of the vibration sensors so that they can perform the shaft, gear and bearing analysis
- A bused sensor system to reduce the weight and cost of harnessing, and
- A cloud based user display, again to lower the setup and maintenance cost to the user.
- A true prognostic capability by providing an estimate of the remaining useful life of a component based on a physics of failure model.

Ultimately, it is anticipated that such architectures will deliver world class diagnostics/prognostic performance at a fraction of the cost of traditional, “star” architecture CMS using PZT accelerometers.

### 1.1. System Considerations with Using MEMS

The newest generation of MEMS accelerometers offers performance that in many cases is superior to traditional PZT devices *if it is packaged correctly*. MEMS accelerometers sense changes in capacitance, based on distance from a reference, instead of charge (piezoelectric effect) due to shear. Because of this physically different way to measure acceleration, these devices can measure from DC to 32 KHz. However, since MEMS accelerometers are capacitive sensor (vibration changes sensor capacitance, which is proportional to voltage), they are sensitive to electromagnetic interference (PZT have better electromagnetic noise immunity). As such, to ensure performance near a large generator, they must be packaged with an analog to digital converter at the sensor.

One disadvantage of MEMS accelerometers is that they are noisier than PZT accelerometers. The power spectral density of a typical accelerometer at 1 KHz is 10 to 190 $\mu\text{g}/\sqrt{\text{Hz}}$  (see Analog). Compare this to a wideband MEMS device, such as Analog Devices ADXL001 with 4 $\text{mg}/\sqrt{\text{Hz}}$ , or approximately 2 to 40 times noisier. That said, from a system perspective the data acquisition, processing and condition indicator (CI) generation gives significant process gain and a large reduction in noise.

As an example, a typical shaft analysis would result in a time synchronous average (TSA, Randal 2011, McFadden 1994) of length 8196 for 20 revolutions. The reduction in non-synchronous noise (part of which is accelerometer self noise) is the product of the process gain due to the TSA ( $1/\sqrt{\text{rev}}$  or 0.2236) and the noise reduction of the spectrum of the TSA ( $1/\sqrt{(\text{length}/2)}$  or 0.011), which in total is 0.0025 \* the spectral density. This reduction is more than adequate for most CMS analysis. It was observed that use of a MEMS

accelerometer does not significantly degrade the ability to detect component fault given the process gain of the CMS analysis.

### 1.2. Opportunities for Using MEMS

The MEMS accelerometer was packaged with the analog-to-digital converter (ADC) because it is sensitive to electromagnetic interference. This necessitates RAM to store the vibration data during the acquisition, a microcontroller to drive the ADC, and a receiver/transmitter to move data. By selecting a microcontroller with a floating point unit, it was possible to perform all of the processing of the vibration data on the sensor.

The sensor, when finished processing (about 20 seconds) sends condition indicator data to the local data concentrator. This greatly reduced to the overall system cost of the system by allowing the use of low cost microcontrollers vs. higher cost Intel or ARM based processors.

### 1.3. Tachometer Function

Because vibration processing is done locally at the sensor, the sensor will need zero crossing data to perform analysis on the components that it monitors. The zero crossing data is used for calculation of the time synchronous average.

A tachometer sensor was developed that, instead of sending the digitized output of the Hall Effect speed sensor, measured the zero crossing times (ZCT). The ZCT was then broadcast by the tachometer sensor to all vibration sensors on the network. The output of the Hall Effect speed sensor was tied to the microcontroller general purpose input/output (GPIO) pin. When the GPIO pin went high, the microcontroller measured the time from the last interrupt. The clock on the microcontroller was 100 MHz, with a jitter 20 parts per million.

## 2. INITIAL VETTING AND VERIFICATION

Following the installation in early December 2011, the CMS was evaluated to ensure:

- That the system MEMS based sensors were measuring vibration data,
- That the configuration was correct, and processing appropriate for Shaft, Gear and Bearing condition indicators.

The CMS was configured to acquire data for 40 seconds. As noted, the generator is a variable speed system: a lower limit of 11,000 rpm was set on the CMS to ensure that the CI collected would be taken at similar torque/rpm values. This gave a range of 7 to 12 revolutions on main rotor. At the high speed side (total gearbox ratio of 1:96) the output shaft and generator has 105 to 150 revolutions.

## 2.1. MEMS Accelerometer Accuracy

There was an initial concern that the MEMS accelerometer would be too noisy to accurately measure the vibration on the gearbox. Shaker table testing found that the MEMS accelerometers were typically within 2% error. This testing was conducted at higher frequencies and G values (needed because of the feedback system on the shaker itself) than would be seen on the planetary section of the wind turbine.

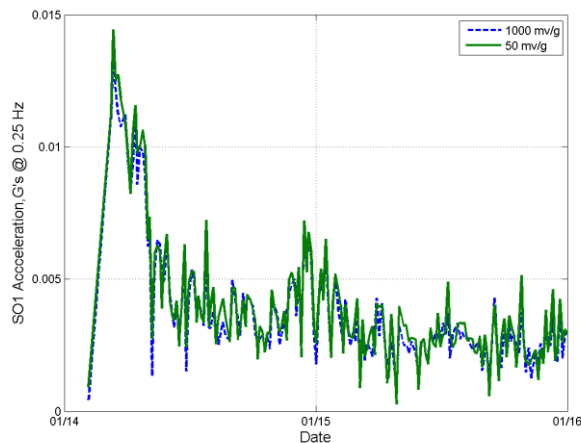


Figure 1 Comparison of Co-Located Sensor (1000 mv/g vs. 50 mv/g)

Two sensors were co-located on the main bearing of the turbine. The first sensor was a low frequency (0-20 Hz) sensor, 1000 mv/g, sampled at 1KHz. The second sensor was a high frequency (0-32 KHz) sensor, 50.4 mv/g, sampled at 3.3 Hz. This gives a time synchronous average (TSA) length of 32768 points. Over the course of the winter, a large SO1 value was measured on both sensors (and on the planetary carrier sensor as well). It was suspected that this was an icing event, which was subsequently confirmed (Figure 2). Because this was a real signal, the SO1 between the two sensors should be seen and were compared to see how the high frequency (low gain) sensor performed relative to the more accurate, low frequency sensor (Figure 1).

The SO1 values were calculated from TSA with only 7 revolutions, at a shaft rate of .25 Hz. The correlation between the two sensors was 0.98, with a 4% bias error in the high frequency sensor. The RMS error is  $5e-4$  Gs. This is remarkable performance in detecting low G signals at low frequency. More so in that the high speed sensor is operating at only 0.007% of full range. This suggests that the MEMS accelerometers are capable of detecting gear faults in the planetary section of the gearbox.



Figure 2 Icing on Wind Turbine Blade

## 2.2. Tachometer and Main Rotor Speed Variation

It was immediately noted that the main rotor had large variations in speed over the 40 second acquisition. Because of the extremely low shaft rate (0.18 to .25 Hz), the acquisition must be extended long enough to capture an imbalance on the main rotor. Additionally, an acquisition on the Ring/Carrier/Planets must be long enough to generate valid TSA (admittedly, 7 revolutions is not a large number of revolutions). Because of the large variance in wind speed, it was found that variation in main rotor RPM average 0.5%. Some acquisitions had variations in RPM of greater than 2.5% (Figure 3).

These large variations in shaft speeds will be propagated throughout the gearbox, with the effect being greatest on the low speed shafts/gears. This is because the higher speed shafts require smaller acquisition times. As noted, the high speed shaft, which is turning 95.9 times faster than the low speed shaft, requires significantly less time to get one revolution. In fact, a six second acquisition results in 150 revolutions of the shaft.

Without some method to normalize the variation in shaft rate, there will be smearing in the spectrum (Figure 4). This example is taken from the planetary carrier sensor, where the 123 tooth ring gear frequency is clearly present at 27.2 Hz. There are three planets, and which will result in sidebands at 26.53 and 27.86 Hz ( $\pm 3 \times$  shaft rate, which for this acquisition was 0.22 Hz). In the Figure 4 subplots, the spectral representation of the raw spectrum is smeared. At higher harmonics (2<sup>nd</sup> harmonic at 54.4 Hz, and 3<sup>rd</sup> harmonic at 81.6 Hz), the raw spectrum is hardly greater than the base noise.

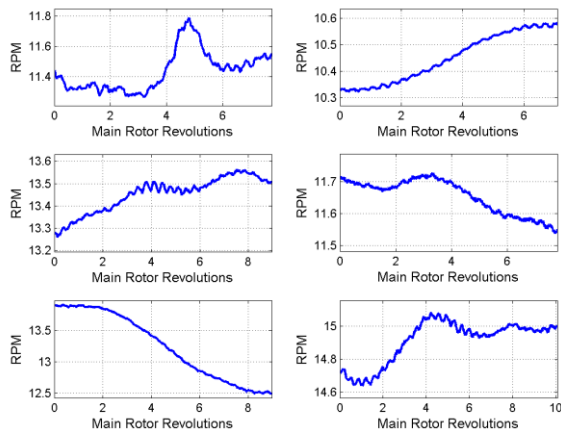


Figure 3 Examples of Main Rotor Speed Variations

This is in comparison to the TSA spectrum, which clearly shows the expected sidebands. Since many gear fault algorithms are based on the ratio of the gear mesh frequency to its sideband, without normalization, the ability to detect gear fault on the lower speed shaft is greatly reduced.

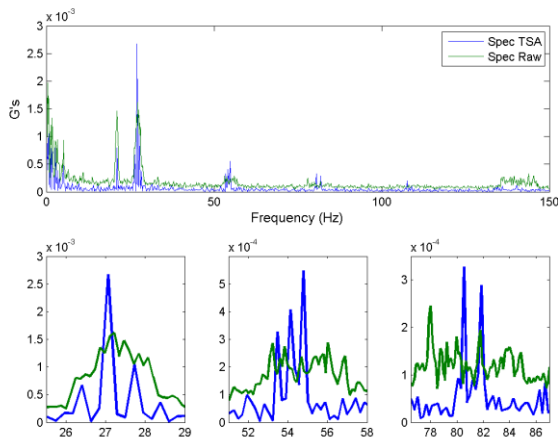


Figure 4 Comparison of Spectrum of TSA vs. Spectrum of Raw Vibration Data

### 2.3. The Need for Improved an TSA

The observant reader will note that in Figure 3, there is a ripple in the rotor speed. On closer observation (Figure 5), it is seen that there is a 3/revolution change in RPM overlaid on the RPM change as a result of changes in wind speed.

This phenomenology has been observed (Dolan, 2006) by wind turbine controls and power conversion engineers. These 3/rev oscillations are important from their perspective since they could have wide ranging effects on control systems and power quality. In systems connected directly to the grid, these torque oscillations could affect of grid power quality. For systems interfaced to the grid through

converters, these torque oscillations would be more important in terms of converter control.

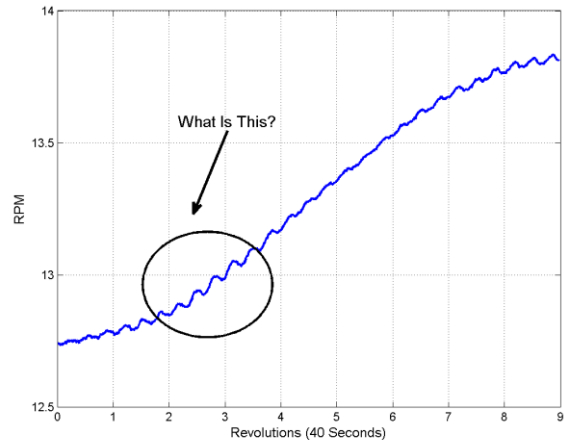


Figure 5 Example 3/Rev Torque Ripple

This torque ripple is the result of tower shadow and wind shear. Tower shadow occurs because the wind flow directly in front of the tower is stalled. As the blade passes in front of the tower at the bottom of the arc, it generates less lift which reduces the torque on the hub. Wind shear occurs because air is a viscous fluid: wind speed increases with height. As the blade reaches the top of the rotor arch, the blade generates more lift which increases the torque on the hub.

From a condition monitoring perspective, there has been no reporting of this shaft behavior. This could have deleterious effects on the performance of the TSA. Typically (McFadden (1997), Bechhoefer (2009)) the model for the TSA assumes linear increase/decrease in rotor speed. TSA using Spline interpolation could control a 1/rev torque ripple. The 3/rev torque ripple violates both methodologies. This required the development of an “enhanced” TSA algorithm.

Current TSA algorithms uses a tachometer input to calculate the time (and number of sampled data points) in one revolution of the shaft under analysis. The sampled data points are then re-sampled using linear/spline interpolation. In the enhanced TSA algorithm, each revolution was partitioned into 16 inter revolution sections, on which the data points were re-sampled (Figure 6).

16 inter revolution sections where used because:

- From Nyquist, to reconstruct the 3/rev, at least 6 sub-sections would be needed.
- The Fourier Transform used in this implementation was a Radix-2, thus the TSA always is a power of 2. To divide evenly, the sub-sections should also be a power of 2

- Both 8 and 16 inter revolution sections methods were tested, the 16 inter revolution section version had marginally improved performance (RMS error between re-sampled and original data).

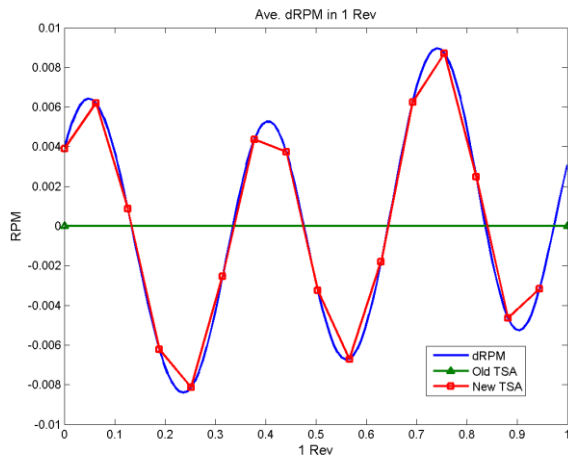


Figure 6 Comparison of the 3/Rev change in RPM, vs the TSA and "Enhanced" TSA

For example, the TSA for the carrier sensor, based on a .22 Hz shaft rate and a sample rate of 3296 sps, had 32768 data points. For each of the 16 inter revolution sections, the sampled vibration data was linearly interpolated into 2048 data values.

In Figure 6, the TSA of the shaft RPM data was taken, and the result was de-trended. This represents the change in main rotor RPM over one revolution. The old/current TSA algorithm would resample the data linearly between one revolution. The enhanced TSA represents the 3/Rev change in RPM by “chopping” one revolution into 16 pieces, and linearly interpolating.

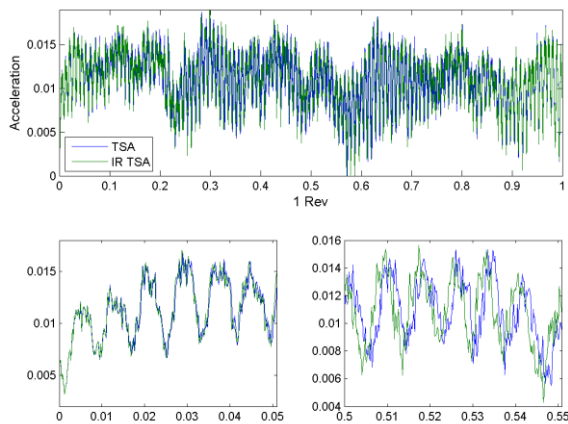


Figure 7. Difference in Phase Between Old and Enhanced TSA

The enhanced TSA better represents the phase changes of the vibration signal better than the original TSA (Figure 7).

In Figure 7, the two TSA algorithms start and end in phase (see subplot 1). However, the difference in phase soon becomes apparent. This phase error is similar to jitter, which reduces the ability of the FFT to produce an accurate spectrum. Similar to the comparison of a raw spectrum with a TSA spectrum, the enhanced TSA will show a better representation of the gearbox spectrum (Figure 8).

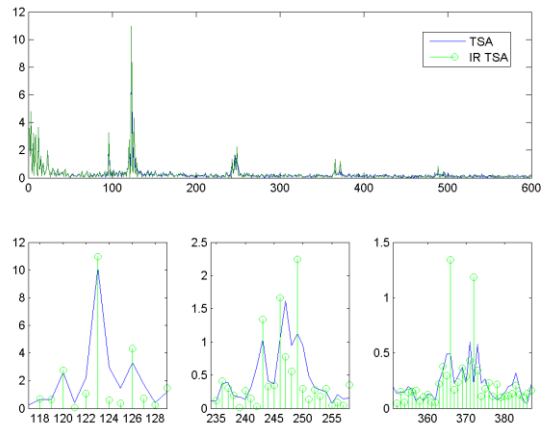


Figure 8 Difference in Spectrum Between Old and Enhanced TSA

Note that the enhanced TSA has more distinct side bands, and that the 2<sup>nd</sup> and 3<sup>rd</sup> harmonics of the ring gear are clearly more prominent than in the old/current TSA algorithm. The enhanced TSA algorithm itself did not greatly add to the processing time.

#### 2.4. Inline Decimation in the TSA

There are two contending system issues when selecting the accelerometer sample rate. For bearing analysis, one needs to sample at a high enough sample rate to capture the structural resonance of the bearing. This is needed for bearing envelope analysis. For the TSA, one needs to sample at a low enough rate such that the length of the TSA is less than the maximum allowable FFT length (which is 32768 data points).

This becomes a problem for larger wind turbines (2MW and greater) where the main shaft rate is a fraction of a Hertz. For example, consider main shaft with turning at 11 RPM (0.18 Hertz). For the main and carrier bearing, one would like capture the 2 KHz to 2.2 KHz window for bearing analysis. This requires sampling at greater than 4.4 KHz. The closest sample rate for the CMS is 6104 sps. For this shaft rate, the length of the TSA is:

$$(1)$$



This is longer the maximum allowable FFT. The next lower sample rate is 3052, but this is too low for the bearing analysis.

Because there is limited processing resources on the sensor, an inline low pass filter and decimate capability was added to the TSA:

- If the length of the TSA,  $n > 32768$ , then
  - Decimate =  $n/32768$ ,
  - Filter coefficients are derived for a 4 point, FIR filter design, where the normalized frequency is  $1/\text{Decimate}$ .
  - For Decimation of 2,  $b = [0.204 \ 0.593 \ 0.204]$

The flow of the enhanced TSA algorithm is:

```

for 1 to # of TSA Revolution
  for 1 to 16 (the number of sub sections to capture 3/rev
    interpolate the vector of zero cross times
    get the change in time between the re-sampled data, dt.
    for the length of each sub section
      if decimate = 1 (no decimation)
        interpolate the data point at
          index, index + 1, for time dt.
      Else
        Interpolate the data point by filtering
          the data point at index -1,0,1 using b
          the data point at index 0,1,2 using b
  
```

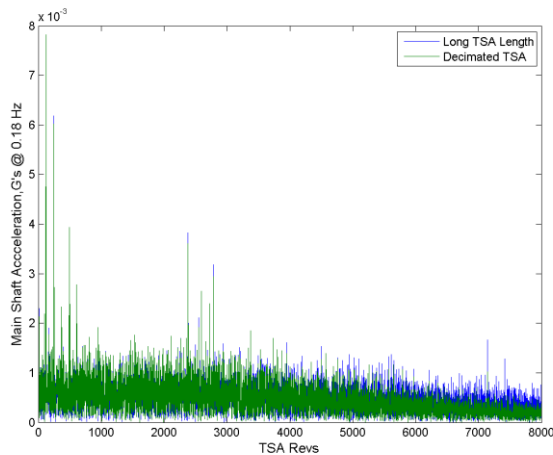


Figure 9 Effect on In-line Decimation on TSA Spectrum

This inline process allows the higher sample rate required for bearing analysis, and does not greatly increase the order of operations for the TSA algorithm. It does not affect the gear or shaft analysis at all. Shaft analyses measures the 1st, 2nd and 3rd harmonics (bin 2, 3 and 4 of the TSA FFT), while the gear diagnostics, which analysis higher harmonics, is improved. This is because of the reduction in higher frequency noise (Figure 9).

The ring gear is 123 teeth, giving the first 5 harmonics (up to bin 615). There are additional tones at 2382 through 2782 (which represent 428 to 500 hz values) are undetermined. Above this frequency, the spectrum is measuring broadband noise.

### 3. VALIDATION OF BEARING FAULT DETECTION WITH MEMS ACCELEROMETERS

There is a concern that the high spectral noise floor of MEMS based accelerometers will make them an inappropriate sensor for bearing analysis. Early stage (stage 3) bearing faults have spectral content typically 3 orders or magnitude smaller than spectral content of gears or shafts. This makes fault detection difficult with even the lowest noise PZT accelerometers. To verify the ability to detect bearing faults, a test rig was developed on which nominal and faulted bearing could be run.

Both inner and outer races faults were developed. Testing was conducted with a shaft rate of 25 Hz, which is approximately the rate of wind turbine high speed shaft. The load on the bearing was varied from 0, 25, 50, 100, 150, 200, 250 and 300 lbs of load (the design load of the bearing was 1025 lbs.). Figure 10 is an example of the outer race fault.



Figure 10. Example Outer Race Fault

The outer race bearing fault rate was 80.4 Hz. Envelope analysis performed on the sensor was with windows of: 0.5-1.5 KHz, 2.5-3.5 KHz, 4-5 KHz, 9-10 KHz, 10-11 KHz, 13-14 KHz, and 22-24 KHz. Surprisingly, the envelope energy did not vary greatly with window, and was relatively independent of load. For the level of damage (Figure 10), it was found that the damage outer race energy was approximately 10x the nominal bearing energy and easily detected (Figure 11).

Similar results were obtained for inner race fault. This data set will be made available at [www.mftp.org](http://www.mftp.org).

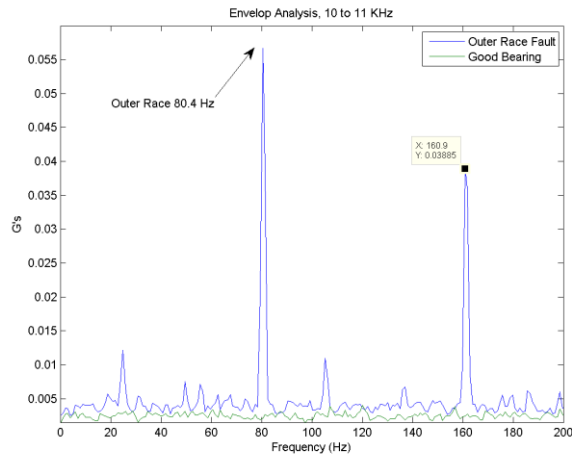


Figure 11 Outer Race Fault Using MEMS Accelerometer

#### 4. ALGORITHM FOR THE DETECTION OF ICE/BLADE PITCH ERROR

It has been shown that the main rotor is sensitive to both tower shadow and wind shear. These phenomena, due to changes in blade lift as a result of changes in wind speed, could be used to detect difference in lift between each blade. For example, if each blade is identical, then the lift generated on each blade would be identical at a given angle on the hub. This in turn would generate a sinusoidal 3/revolution change in RPM. The amplitude of the Hilbert transform of this 3/rev sinusoid would be nearly constant (McFadden, 1986).

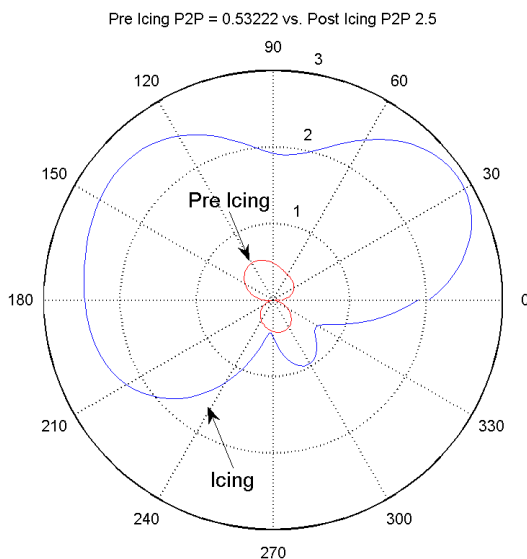


Figure 12 Pre/Post Icing Change in RPM

Consider what would happen if either the blades had icing or if the pitch angle of one blade was in error. The lift

generated by that blade would be less. This blade would generate less torque, and as a result, there would a smaller increase in RPM due to when coming out of the tower shadow or as a result of wind shear.

Most operators will not allow one to deliberately fault a turbine. But, as noted (see Figure 2) icing can occur. Since the CMS is down loading raw, time domain data four times a day, if one can capture a raw data collection during an icing event, one can test the proposed hypothesis. This occurred during the January 14 icing even (Figure 12).

During an icing event, the balance of the rotor can be greatly affected. Prior to the icing event, the SO1 imbalance was .001Gs (about .25 ips). The peak-2-peak change in amplitude of the Hilbert transform of the main rotor RPM (AHT) was 0.5 RPM. Just after the icing event, the SO1 acceleration peaked at 0.14Gs (3.5 ips!) with an AHT of 2.5 RPM.

Since blade pitch error is a common cause of underperformance in a wind turbine, this potentially could be a good indicator of that type of fault.

#### CONCLUSIONS

Condition Monitoring of wind turbines poses some unusual requirement on the CMS. The slow shaft rate of the main shaft results in low amplitude, low frequency vibrations, while the high speed side requires high bandwidth to detect gear and bearing faults. This in turn requires the development of highly sensitive accelerometer with a bandwidth from close to DC (0 Hz) to above 10 KHz. While MEMS sensors are typically noisier than PZT accelerometers, it was found that the MEMS sensors were both accurate, and have low enough spectral noise to capture the vibration features on the turbine. This was observed at very low signal intensities and frequencies. This would be difficult to replicate this performance with a PZT accelerometer.

The extremely low frequencies on the main shaft required the development of an in-line decimation enhancement to the TSA. This allowed the sample rate of the sensor to be high enough for bearing envelope analysis, while limiting the length of the TSA to a maximum length of 32768.

The ability of MEMS accelerometers to diagnose and detect stage 3 bearing faults was also validated.

Other peculiarities of a wind turbine are:

- The large change in main rotor RPM due to changing wind conditions over an acquisition, and
- A smaller, 3/revolution change in RPM due to tower shadow and wind shear.

This required the development of an enhanced TSA algorithm to accurately control the 3/rev change in speed.

The ability to detect small changes in main rotor RPM can facilitate icing or blade pitch errors. This was demonstrated during an icing event. This new algorithm will be deployed and verified in the near future.

#### REFERENCES

- ADXL001 product specification, 'http://www.analog.com/en/mems-sensors/mems-accelerometers/adxl001/products/product.html', Analog Devices
- Randall, R. (2011) *Vibration-based Condition Monitoring: Industrial, Aerospace & Automotive Application*, John Wiley, New York.
- McFadden, P., (1987). "A revised model for the extraction of periodic waveforms by time domain averaging", *Mechanical Systems and Signal Processing* 1 (1), pages 83-95.
- Bechhoefer, E., Morton, B., (2012). "Condition Monitoring Architecture to Reduce Total Cost of Ownership". *IEEE PHM Conference*, Denver.
- Dolan, D., Lehn, R., (2006). "Simulation Model of Wind Turbine 3p Torque Oscillation due to Wind Shear and Tower Shadow", *IEEE Transaction on Energy Conversion*, VOL. 21, NO. 3.
- Bechhoefer, E., Kingsely, M., (2009) "A Review of Time Synchronous Average Algorithms", Annual Conference of the Prognostics and Health Management Society.

McFadden, P.D. (1986). Detecting Fatigue Cracks in Gear by Amplitude and Phase Demodulation of the Meshing Vibration. *ASME J. of Vibration, Acoustics, Stress, and Reliability in Design* 108, 165-170.

#### BIOGRAPHIES

**Eric Bechhoefer** is the chief engineer at NRG Systems. A former naval aviator, Dr. Bechhoefer recently joined NRG from the aerospace industry. Dr. Bechhoefer has 20 patents and over 70 juried papers related to condition monitoring of rotating equipment.

**Matthew Wadham-Gagnon** is a project manager in wind turbine blade icing related topics at the TechnoCentre Eolien (TCE) since 2011. Prior to joining the TCE, he was involved in the wind energy industry for over 4 years in fields related to structural composite design and composite processing. Matthew has a Master's degree in Mechanical Engineering.

**Bruno Boucher** is the Operations and Maintenance at the TechnoCentre Eolien (TCE) since 2010. Prior to joining the TCE, he was involved in the wind industry for over 7 years in O&M of wind farm and design of permanent magnet generator. Bruno has a Bachelor's Degree in Electromechanical Systems Engineering.

# A Distributed Approach to System-Level Prognostics

Matthew Daigle<sup>1</sup>, Anibal Bregon<sup>2</sup>, and Indranil Roychoudhury<sup>3</sup>

<sup>1</sup> NASA Ames Research Center, Moffett Field, CA 94035, USA  
matthew.j.daigle@nasa.gov

<sup>2</sup> University of Valladolid, Valladolid, Spain  
anibal@infor.uva.es

<sup>3</sup> SGT Inc., NASA Ames Research Center, Moffett Field, CA 94035, USA  
indranil.roychoudhury@nasa.gov

## ABSTRACT

Prognostics, which deals with predicting remaining useful life of components, subsystems, and systems, is a key technology for systems health management that leads to improved safety and reliability with reduced costs. The prognostics problem is often approached from a component-centric view. However, in most cases, it is not specifically component lifetimes that are important, but, rather, the lifetimes of the systems in which these components reside. The system-level prognostics problem can be quite difficult due to the increased scale and scope of the prognostics problem and the relative lack of scalability and efficiency of typical prognostics approaches. In order to address these issues, we develop a distributed solution to the system-level prognostics problem, based on the concept of structural model decomposition. The system model is decomposed into independent submodels. Independent local prognostics subproblems are then formed based on these local submodels, resulting in a scalable, efficient, and flexible distributed approach to the system-level prognostics problem. We provide a formulation of the system-level prognostics problem and demonstrate the approach on a four-wheeled rover simulation testbed. The results show that the system-level prognostics problem can be accurately and efficiently solved in a distributed fashion.

## 1. INTRODUCTION

Prognostics is the process of predicting the end of (useful) life (EOL) and/or the remaining useful life (RUL) of components, subsystems, or systems. The prognostics problem itself can be divided into two distinct problems: (i) the *estimation* problem, which determines the current state of the system, and (ii)

the *prediction* problem, which, using the current system state estimate, computes EOL and/or RUL. In this paper, we focus on a model-based prognostics approach (Orchard & Vachtsevanos, 2009; Daigle & Goebel, 2011b; Saha & Goebel, 2009; Luo et al., 2008). In model-based prognostics, an underlying model of the system, its components, and how they fail is leveraged, where health state estimation is formulated as a joint state-parameter estimation problem, typically using a filtering approach, and prediction is formulated as a simulation problem (Daigle, Saha, & Goebel, 2012).

To the best of our knowledge, all prognostics research to date has been focused on *individual components*, and determining their EOL and RUL, e.g., (Orchard & Vachtsevanos, 2009; Saha & Goebel, 2009; Daigle & Goebel, 2011a; Celaya et al., 2011; Bolander et al., 2010; Luo et al., 2008; Byington et al., 2004). However, in many cases, the desired information is the EOL of the *system*, which is obtained through *system-level prognostics*. Generally, the EOL of a system depends on its constituent components and how they interact. Approaching this problem from the centralized perspective becomes very difficult, as common (centralized) prognostics algorithms may not scale to the system level.

In order to address the problems with centralized approaches, in recent work, we have developed a distributed model-based prognostics architecture that allows the decomposition of a large prognostics problem into several independent local subproblems from which local results can be merged into a global result (Daigle et al., 2011; Daigle, Bregon, & Roychoudhury, 2012). Since each local subproblem can be solved independently, each can be assigned to a different processing unit and be solved in parallel. Such a distributed approach is in contrast to other proposed distributed prognostics architectures in which the global problem is not decomposed and the computation is distributed onto multiple processing units, e.g., (Saha, Saha, & Goebel, 2009). Our distributed approach

Matthew Daigle et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

scales well and the resulting subproblems are typically small and easy to solve, resulting in an efficient and flexible distributed solution to the prognostics problem. Such an approach has obvious advantages when applied to the system-level prognostics problem. In this paper, we formulate the system-level prognostics problem and propose a solution using this distributed prognostics framework. We apply our system-level prognostics approach to a rover testbed and provide results in simulation to empirically demonstrate and validate the approach.

The paper is organized as follows. Section 2 formulates the system-level prognostics problem and overviews the proposed distributed solution. Section 3 describes the estimation problem, and Section 4 describes the prediction problem. Section 5 presents the rover case study, and shows prognostics results in simulation. Section 6 concludes the paper.

## 2. SYSTEM-LEVEL PROGNOSTICS

While most prognostics approaches focus on individual components, in most practical cases it is actually the EOL of the system that must be determined. With this prediction, the future usage of the system may be optimally planned to maximize system life and to schedule system-wide maintenance activities. It is often important to take a system-level perspective of prognostics, because the degradation of individual components is often coupled, i.e., the way one component degrades is dependent on how a connected component degrades. This may occur, for example, if one component provides the inputs to another component, in which case, prognostics of the latter component cannot be performed in isolation.

In this section, we first define the system-level prognostics problem. We then introduce the system-level prognostics approach and architecture using a distributed prognostics framework that is based on structural model decomposition.

### 2.1. Problem Formulation

The goal of system-level prognostics is the prediction of the EOL and/or RUL of a system. We assume the system model may be generally defined as

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{f}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{v}(t)), \\ \mathbf{y}(t) &= \mathbf{h}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{n}(t)),\end{aligned}$$

where  $\mathbf{x}(t) \in \mathbb{R}^{n_x}$  is the state vector,  $\boldsymbol{\theta}(t) \in \mathbb{R}^{n_\theta}$  is the unknown parameter vector,  $\mathbf{u}(t) \in \mathbb{R}^{n_u}$  is the input vector,  $\mathbf{v}(t) \in \mathbb{R}^{n_v}$  is the process noise vector,  $\mathbf{f}$  is the state equation,  $\mathbf{y}(t) \in \mathbb{R}^{n_y}$  is the output vector,  $\mathbf{n}(t) \in \mathbb{R}^{n_n}$  is the measurement noise vector, and  $\mathbf{h}$  is the output equation.<sup>1</sup> This model describes both the nominal behavior and faulty behavior, including the fault progression functions.

<sup>1</sup>Here, we use bold typeface to denote vectors, and use  $n_a$  to denote the length of a vector  $\mathbf{a}$ .

In system-level prognostics, we are interested in when the performance of a system lies outside some desired region of acceptable behavior. The desired performance is expressed through a set of  $n_c$  constraints,  $C_{EOL} = \{c_i\}_{i=1}^{n_c}$ , where  $c_i : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_u} \rightarrow \mathbb{B}$  maps a given point in the joint state-parameter space given the current inputs,  $(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t))$ , to the Boolean domain  $\mathbb{B} \triangleq [0, 1]$ , where  $c_i(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t)) = 1$  if the constraint is satisfied. If  $c_i(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t)) = 0$ , then the constraint is not satisfied, and the behavior of the system is deemed to be unacceptable. These deterministic constraints may refer to component-level, subsystem-level, or system-level specifications or requirements and define a fixed partition of the state-parameter-input space into acceptable and unacceptable regions of behavior. When the constraints are violated, it does not necessarily refer to a hard failure, but any point at which the operational risk is too large to continue system operation, or future behaviors of the system will be in some way unacceptable. At this point we say the system has no *useful* life remaining.

These individual constraints may be combined into a single system-level *threshold function*  $T_{EOL} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_u} \rightarrow \mathbb{B}$ , defined as

$$T_{EOL}(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t)) = \begin{cases} 1, & 0 \in \{c_i(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t))\}_{i=1}^{n_c} \\ 0, & \text{otherwise.} \end{cases}$$

$T_{EOL}$  evaluates to 1, i.e., the system has reached an unacceptable region of behavior, when any of the constraints are violated. EOL is then defined as

$$EOL(t_P) \triangleq \inf\{t \in \mathbb{R} : t \geq t_P \wedge T_{EOL}(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t)) = 1\},$$

i.e., EOL is the earliest time point at which  $T_{EOL}$  is met (evaluates to 1). RUL is expressed using EOL as

$$RUL(t_P) \triangleq EOL(t_P) - t_P.$$

Note that because  $\mathbf{x}(t)$  is a random variable,  $EOL$  and  $RUL$  must necessarily be random variables also.

### 2.2. Prognostics Approach

In order to make an EOL or RUL prediction for the system, the initial state from which to make a prediction is required. In general, this initial state is not directly observed, and must be estimated. Therefore, there are two sequential problems for prognostics: the *estimation* problem and the *prediction* problem. The estimation problem is to find a joint state-parameter estimate  $p(\mathbf{x}(t), \boldsymbol{\theta}(t) | \mathbf{y}_{0:t})$  based on the history of observations up to time  $t$ ,  $\mathbf{y}_{0:t}$ . This estimate is represented as a probability distribution because, generally, the system

state is not directly observed, and there is sensor noise,  $\mathbf{n}(t)$ , and process noise,  $\mathbf{v}(t)$ . At a given prediction time,  $t_P$ , the prediction algorithm uses the joint state-parameter estimate  $p(\mathbf{x}(t_P), \boldsymbol{\theta}(t_P) | \mathbf{y}_{0:t_P})$  and computes  $p(EOL(t_P) | \mathbf{y}_{0:t_P})$  and  $p(RUL(t_P) | \mathbf{y}_{0:t_P})$ . Along with the uncertainty in the state-parameter estimate, process noise and uncertainty in the future inputs to the system all contribute to the uncertainty in the EOL/RUL prediction.

This system-level prognostics problem, consisting of estimating the system state and then predicting its evolution to EOL, can be solved using component-level approaches by treating the entire system as a single component and applying these approaches directly. However, for a large system, both the estimation and prediction problems are correspondingly large. Due to the large state-parameter dimension, a centralized approach does not scale well, and can be very inefficient.

Therefore, we propose to decompose the *global* system-level prognostics problem into independent *local* subproblems, such that the solutions to the local subproblems may be easily merged to form the solution to the global prognostics problem. This forms a naturally distributed approach in which the local subproblems, since they are independent, may be solved in parallel, thus providing scalability and efficiency. Further, the approach allows different algorithms to be employed on each subproblem. The subproblems often correspond directly to component-level prognostics problems, and the approach provides a mechanism to combine component-level prognostics results into system-level results.

In (Daigle et al., 2011), we developed such a distributed solution to the estimation part of the prognostics problem, based on the concept of structural model decomposition (Pulido & Alonso-González, 2004). In recent work, the same concept was used to decompose the prediction problem (Daigle, Bregon, & Roychoudhury, 2012). Structural model decomposition allows one to decompose a system model into a set of submodels for which local prognostics problems can be directly defined. The global model of the system, denoted as  $\mathcal{M}$ , is defined as follows.

**Definition 1 (Model).** The model of a system,  $\mathcal{M}$ , is a tuple  $\mathcal{M} = (X, \Theta, U, Y, C)$ , where  $X$  is the set of state variables of  $\mathbf{x}$ ,  $\Theta$  is the set of unknown parameters of  $\boldsymbol{\theta}$ ,  $U$  is the set of input variables of  $\mathbf{u}$ ,  $Y$  is the set of output variables of  $\mathbf{y}$ , and  $C$  is the set of model constraints of  $\mathbf{f}$ ,  $\mathbf{h}$ , and  $C_{EOL}$ .

Informally, a model consists of a set of variables and a set of constraints among the variables. While technically  $\mathbf{f}$  and  $\mathbf{h}$  themselves are (complex) constraints, we represent them instead as sets of simple constraints. This view is also more consistent with the way modelers describe  $\mathbf{f}$  and  $\mathbf{h}$ , i.e., as sets of equations, each describing a single state or output variable.

Model decomposition is accomplished by assigning some variables as local inputs for which the values are known (e.g.,

they are directly measured). In this way, the submodels are made computationally independent of each other. Within this scheme, a submodel is then defined as follows.

**Definition 2 (Submodel).** A submodel  $\mathcal{M}_i$  of a system model  $\mathcal{M} = (X, \Theta, U, Y, C)$  is a tuple  $\mathcal{M}_i = (X_i, \Theta_i, U_i, Y_i, C_i)$ , where  $X_i \subseteq X$ ,  $\Theta_i \subseteq \Theta$ ,  $U_i \subseteq X \cup U \cup Y$ , and  $Y_i \subseteq Y$  are the state, parameter, input, and output variables, respectively, and  $C_i \subseteq C$  are the submodel constraints.

For distributed prognostics, we find a set of submodels that satisfy a certain set of properties. For distributed estimation, the submodels use  $U_i \subseteq U \cup (Y - Y_i)$ , and we find a set of minimal submodels such that each  $Y_i$  is a singleton, and over all  $Y_i, Y_j$  where  $i \neq j$ ,  $Y_i \cap Y_j = \emptyset$ . So, each submodel uses some global model inputs and some measured values as local inputs, and, in this way, the submodels become decoupled and may be computed independently from each other. By creating submodels with one output variable each, we maximize the number of estimation submodels and the opportunity for parallelization of the estimation task. By making the submodels minimal, they require no constraints or variables that are not strictly necessary to compute  $Y_i$ . An algorithm for computing the set of submodels with these properties is given in (Daigle et al., 2011), which is based on the model decomposition algorithms presented in (Pulido & Alonso-González, 2004; Bregon, Biswas, & Pulido, 2012).

For distributed prediction, the submodels use  $U_i \subseteq U_P$ , where  $U_P \subseteq X \cup U$ . Here,  $U_P$  is a set of variables whose future values can be hypothesized. In the centralized case,  $U_P = U$ . We find a set of minimal submodels such that each submodel has at least one  $c \in C_{EOL}$  belonging to  $C_i$ , and over all submodels,  $C_{EOL}$  is covered. This ensures that  $T_{EOL}$  may be computed for the system; since  $T_{EOL}$  is 1 whenever any of the constraints in  $C_{EOL}$  are violated, we can independently evaluate when those individual constraints will be violated and then take the minimum to obtain the system EOL. An algorithm for computing the set of submodels with these properties is given in (Daigle, Bregon, & Roychoudhury, 2012). Both decomposition algorithms work in a similar way; essentially, they start with a variable or constraint that must be computed in the local submodel, and then trace the dependencies backwards until local inputs are reached, including all variables and constraints found throughout the search within the submodel.

Note that the problem of defining  $U_P$  is critical to obtaining accurate results for system-level EOL in a distributed manner. On average, the most accurate result will be achieved when the system model is directly used for prediction, because it captures all the interdependencies between the components. In the general case, damage could be progressing in multiple components, and damage progression in one component may have an effect on damage progression in another component due to their coupling. In such cases, for system-level prog-

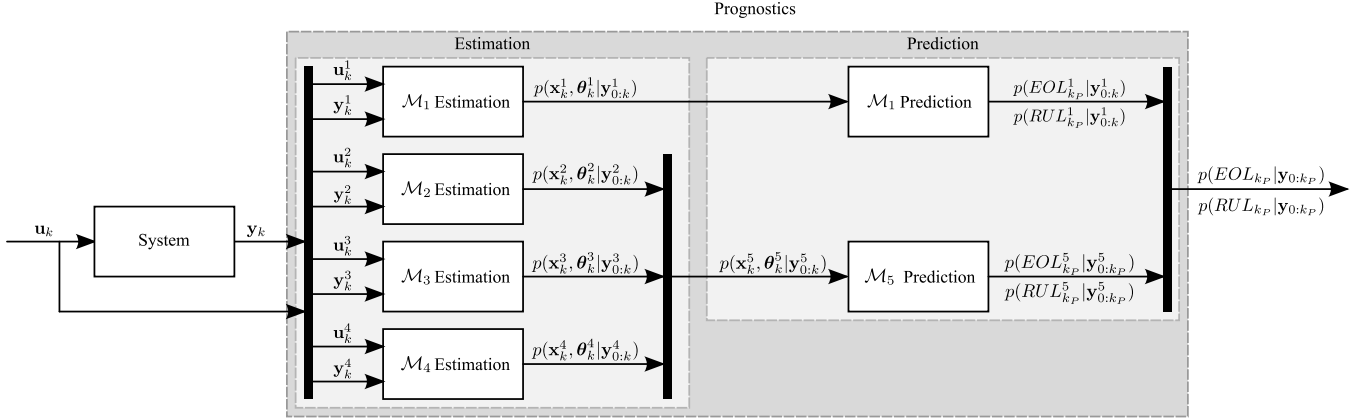


Figure 1. Sample system-level prognostics architecture.

nostics the components cannot be decoupled due to these interactions, and the prediction problem cannot be decomposed into two independent problems, one for each component. It is only appropriate to neglect these interactions when they are either negligible or predictable a priori. It will be shown in Section 5 how this is an important consideration.

### 2.3. Prognostics Architecture

A sample system-level prognostics architecture based on the distributed framework is shown in Fig. 1. In discrete time  $k$ , and using a discrete-time version of the model, the damage estimation module takes as input both  $\mathbf{u}_k$  and  $\mathbf{y}_k$  and splits them into local inputs and outputs for the submodels. Estimation is performed for each submodel using an appropriate algorithm, computing local state-parameter estimates  $p(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i | \mathbf{y}_{0:k}^i)$ . Some of these local estimates are merged corresponding to the prediction submodels. For example, submodel  $\mathcal{M}_5$  builds its local state using the estimates from the estimators of  $\mathcal{M}_2$ ,  $\mathcal{M}_3$ , and  $\mathcal{M}_4$ . The local predictors compute local EOL/RUL predictions  $p(EOL_{k_P}^i | \mathbf{y}_{0:k_P}^i)$  and  $p(RUL_{k_P}^i | \mathbf{y}_{0:k_P}^i)$  at given prediction time  $k_P$  based on the local EOL constraints. Local predictions are then merged into global predictions  $p(EOL_{k_P} | \mathbf{y}_{0:k_P})$  and  $p(RUL_{k_P} | \mathbf{y}_{0:k_P})$  by taking the minimum of the local predictions.

### 3. DISTRIBUTED ESTIMATION

As described in Section 2, in our distributed estimation scheme, the local estimator for each submodel  $\mathcal{M}_i$  produces a local estimate  $p(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i | \mathbf{y}_{0:k}^i)$ , where  $\mathbf{x}_k^i \subseteq \mathbf{x}_k$ ,  $\boldsymbol{\theta}_k^i \subseteq \boldsymbol{\theta}_k$ , and  $\mathbf{y}_k^i \subseteq \mathbf{y}_k$ . Here, the local inputs used,  $\mathbf{u}^i$ , consist of elements from both  $\mathbf{u}$  and  $\mathbf{y}$ , where measured values are directly used as local inputs. The estimation problem is decomposed by finding a set of minimal submodels that together cover the subset of  $\mathbf{x}$  and  $\boldsymbol{\theta}$  required for prediction, by using these local inputs. This approach to distributed estimation is different from approaches like the distributed decentralized extended Kalman filter (Mutambara, 1998) or other estimation fusion

techniques (Sinha et al., 2008) where local estimates are communicated between local estimators. Here, local estimators do not communicate and operate completely independently.

In order to effectively perform joint state-parameter estimation, the system should be observable, among other requirements. If the global model is structurally observable, then we are guaranteed that the local submodels for estimation are as well (Moya et al., 2010).

Any suitable algorithm may be used for joint state-parameter estimation. In this paper, we use an unscented Kalman filter (UKF) (Julier & Uhlmann, 1997, 2004) with a variance control algorithm (Daigle, Saha, & Goebel, 2012). The UKF assumes the general nonlinear form of the state and output equations described in Section 2, but restricted to additive Gaussian noise.

We summarize the main details of the UKF below, and refer the reader to (Julier & Uhlmann, 1997, 2004) for details. In the UKF, distributions are approximated using the unscented transform (UT). The UT takes a random variable  $\mathbf{x} \in \mathbb{R}^{n_x}$ , with mean  $\bar{\mathbf{x}}$  and covariance  $\mathbf{P}_{xx}$ , that is related to a second random variable  $\mathbf{y} \in \mathbb{R}^{n_y}$  by some function  $\mathbf{y} = \mathbf{g}(\mathbf{x})$ , and computes the mean  $\bar{\mathbf{y}}$  and covariance  $\mathbf{P}_{yy}$  using a minimal set of *deterministically* selected weighted samples, called *sigma points* (Julier & Uhlmann, 1997).  $\mathcal{X}^i$  denotes the  $i$ th sigma point from  $\mathbf{x}$  and  $w^i$  denotes its weight.<sup>2</sup> The sigma points are always chosen such that the mean and covariance match those of the original distribution,  $\bar{\mathbf{x}}$  and  $\mathbf{P}_{xx}$ . Each sigma point is passed through  $\mathbf{g}$  to obtain new sigma points  $\mathcal{Y}$ , i.e.,

$$\mathcal{Y}^i = \mathbf{g}(\mathcal{X}^i)$$

<sup>2</sup>Sigma point weights do not directly represent discrete probabilities, so are not restricted to  $[0, 1]$ .

with mean and covariance calculated as

$$\bar{\mathbf{y}} = \sum_i w^i \mathbf{y}^i$$

$$\mathbf{P}_{yy} = \sum_i w^i (\mathbf{y}^i - \bar{\mathbf{y}})(\mathbf{y}^i - \bar{\mathbf{y}})^T.$$

In this paper, we use the symmetric unscented transform, in which  $2n_x + 1$  sigma points are symmetrically selected about the mean according to (Julier & Uhlmann, 2004):

$$w^i = \begin{cases} \frac{\kappa}{(n_x + \kappa)}, & i = 0 \\ \frac{1}{2(n_x + \kappa)}, & i = 1, \dots, 2n_x \end{cases}$$

$$\mathbf{x}^i = \begin{cases} \bar{\mathbf{x}}, & i = 0 \\ \bar{\mathbf{x}} + \left( \sqrt{(n_x + \kappa) \mathbf{P}_{xx}} \right)^i, & i = 1, \dots, n_x \\ \bar{\mathbf{x}} - \left( \sqrt{(n_x + \kappa) \mathbf{P}_{xx}} \right)^i, & i = n_x + 1, \dots, 2n_x, \end{cases}$$

where  $\left( \sqrt{(n_x + \kappa) \mathbf{P}_{xx}} \right)^i$  refers to the  $i$ th column of the matrix square root of  $(n_x + \kappa) \mathbf{P}_{xx}$ . Here,  $\kappa$  is a free parameter that can be used to tune higher order moments of the distribution. If  $\mathbf{x}$  is assumed Gaussian, then selecting  $\kappa = 3 - n_x$  is recommended (Julier & Uhlmann, 1997).

In the filter, first,  $n_s$  sigma points  $\hat{\mathbf{x}}_{k-1|k-1}^i$  are derived from the current mean  $\hat{\mathbf{x}}_{k-1|k-1}$  and covariance estimates  $\mathbf{P}_{k-1|k-1}$  using a sigma point selection algorithm. The prediction step is:

$$\hat{\mathbf{x}}_{k|k-1}^i = \mathbf{f}(\hat{\mathbf{x}}_{k-1|k-1}^i, \mathbf{u}_{k-1}), i = 1, \dots, n_s$$

$$\hat{\mathbf{y}}_{k|k-1}^i = \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}^i), i = 1, \dots, n_s$$

$$\hat{\mathbf{x}}_{k|k-1} = \sum_i^{n_s} w^i \hat{\mathbf{x}}_{k|k-1}^i$$

$$\hat{\mathbf{y}}_{k|k-1} = \sum_i^{n_s} w^i \hat{\mathbf{y}}_{k|k-1}^i$$

with

$$\mathbf{P}_{k|k-1} = \mathbf{Q} + \sum_i^{n_s} w^i (\mathbf{x}_{k|k-1}^i - \hat{\mathbf{x}}_{k|k-1})(\mathbf{x}_{k|k-1}^i - \hat{\mathbf{x}}_{k|k-1})^T,$$

where  $\mathbf{Q}$  is the process noise covariance matrix. The update

step is:

$$\mathbf{P}_{yy} = \mathbf{R} + \sum_i^{n_s} w^i (\mathbf{y}_{k|k-1}^i - \hat{\mathbf{y}}_{k|k-1})(\mathbf{y}_{k|k-1}^i - \hat{\mathbf{y}}_{k|k-1})^T$$

$$\mathbf{P}_{xy} = \sum_i^{n_s} w^i (\mathbf{x}_{k|k-1}^i - \hat{\mathbf{x}}_{k|k-1})(\mathbf{y}_{k|k-1}^i - \hat{\mathbf{y}}_{k|k-1})^T$$

$$\mathbf{K}_k = \mathbf{P}_{xy} \mathbf{P}_{yy}^{-1}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1})$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{P}_{yy} \mathbf{K}_k^T,$$

where  $\mathbf{R}$  is the sensor noise covariance matrix.

Joint state-parameter estimation is accomplished in the UKF by augmenting the state vector with the unknown parameters, and the corresponding diagonal elements of the process noise matrix,  $\mathbf{Q}$ , are set to nonzero values. In this way, the parameter estimates become time-varying and are modified by the filter using the measured outputs.

The variance values in  $\mathbf{Q}$  associated with the unknown parameters determine both the rate of parameter estimation convergence and the estimation performance once convergence is achieved, therefore, techniques have been developed to tune this value online to maximize performance, e.g., (Liu & West, 2001; Orchard, Tobar, & Vachtsevanos, 2009; Daigle, Saha, & Goebel, 2012). We adopt the approach presented in (Daigle, Saha, & Goebel, 2012), in which the algorithm tries to control the variance of the hidden wear parameter estimate to a user-specified range by modifying the process noise variance. Effectively, the algorithm increases the variance when the relative parameter spread is below the desired level, and decreases it otherwise. With the proper settings, the parameter estimates converge quickly and track with high accuracy and precision.

#### 4. DISTRIBUTED PREDICTION

Each local prediction module takes as input local state-parameter estimates formed from the local estimators, as discussed in Section 2. The required estimates must be constructed from the local estimates of the submodels used for estimation. A prediction submodel has a set of states  $X_i$  and parameters  $\Theta_i$ , and it must construct a local distribution  $p(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i | \mathbf{y}_{0:k}^i)$ . To do this, we assume that the local state-parameter estimates may be sufficiently represented by a mean  $\boldsymbol{\mu}^i$  and covariance matrix  $\boldsymbol{\Sigma}^i$ . For each prediction submodel  $\mathcal{M}_i$ , we combine the estimates from estimation submodels that estimate states and parameters in  $X_i \cup \Theta_i$  into  $\boldsymbol{\mu}^i$  and covariance  $\boldsymbol{\Sigma}^i$ . If there is overlap in the state-parameter estimates, i.e., if two submodels both estimate the same state variable  $x$  or parameter  $\theta$ , then this may be resolved by a number of techniques, e.g., taking the estimate with the smallest variance, or taking an average. Note that, due to the decomposition into independent local submodels,



---

**Algorithm 1** EOL Prediction
 

---

**Inputs:**  $\{(\mathbf{x}_{k_P}^{i(j)}, \boldsymbol{\theta}_{k_P}^{i(j)}), w_{k_P}^{i(j)}\}_{j=1}^N$   
**Outputs:**  $\{EOL_{k_P}^{i(j)}, w_{k_P}^{i(j)}\}_{j=1}^N$   
**for**  $j = 1$  **to**  $N$  **do**  
      $k \leftarrow k_P$   
      $\mathbf{x}_k^{i(j)} \leftarrow \mathbf{x}_{k_P}^{i(j)}$   
      $\boldsymbol{\theta}_k^{i(j)} \leftarrow \boldsymbol{\theta}_{k_P}^{i(j)}$   
     Predict  $\hat{\mathbf{u}}_k^i$   
     **while**  $T_{EOL}^i(\mathbf{x}_k^{i(j)}, \boldsymbol{\theta}_k^{i(j)}, \hat{\mathbf{u}}_k^i) = 0$  **do**  
         Predict  $\hat{\mathbf{u}}_k^i$   
          $\boldsymbol{\theta}_{k+1}^{i(j)} \sim p(\boldsymbol{\theta}_{k+1}^i | \boldsymbol{\theta}_k^{i(j)})$   
          $\mathbf{x}_{k+1}^{i(j)} \sim p(\mathbf{x}_{k+1}^i | \mathbf{x}_k^{i(j)}, \boldsymbol{\theta}_k^{i(j)}, \hat{\mathbf{u}}_k^i)$   
          $k \leftarrow k + 1$   
          $\mathbf{x}_k^{i(j)} \leftarrow \mathbf{x}_{k+1}^{i(j)}$   
          $\boldsymbol{\theta}_k^{i(j)} \leftarrow \boldsymbol{\theta}_{k+1}^{i(j)}$   
     **end while**  
      $EOL_{k_P}^{i(j)} \leftarrow k$   
**end for**

---

we recover only an approximation to the joint posterior distribution as would have been found by a global estimator. In particular, covariance information is lost due to the decoupling and will appear as zeros in the merged covariance matrix. As shown in (Daigle et al., 2011) and as will be seen in Section 5, the approximation still results in accurate predictions.

Given the mean and covariance information, we represent the distribution with a set of sigma points derived using the unscented transform. Then, as in (Daigle & Goebel, 2010), each sigma point is simulated forward to EOL, and we recover the statistics of the EOL distribution given by the sigma points.

The prediction algorithm is executed for each submodel  $i$ , deriving local EOL predictions using its local threshold function based on the local EOL constraints. The pseudocode for the prediction procedure is given as Algorithm 1 (Daigle & Goebel, 2011b). For a given submodel  $\mathcal{M}_i$ , each sigma point  $j$  is propagated forward until  $T_{EOL}^i(\mathbf{x}_k^{i(j)}, \boldsymbol{\theta}_k^{i(j)})$  evaluates to 1. The algorithm hypothesizes future inputs  $\hat{\mathbf{u}}_k^i$ .

Each prediction submodel  $\mathcal{M}_i$  computes a local EOL/RUL distribution, i.e.,  $p(EOL_{k_P}^i | \mathbf{y}_{0:k_P}^i)$  and  $p(RUL_{k_P}^i | \mathbf{y}_{0:k_P}^i)$ . The system EOL is determined by the minimum of all the local distributions, since  $T_{EOL}$  of the system is 1 whenever any of the local constraints are violated, and each local distribution is associated with a subset of these constraints. Specifically, for  $m$  prediction submodels,

$$p(EOL_{k_P} | \mathbf{y}_{0:k_P}) = \min(\{p(EOL_{k_P}^i | \mathbf{y}_{0:k_P}^i)\}_{i=1}^m).$$

To compute this, we sample from each local EOL distribution and take the minimum of the local samples. This is repeated many times and the statistics of the global EOL distribution are computed.

## 5. CASE STUDY

In this section, we apply our system-level prognostics approach to a four-wheeled rover testbed developed at NASA Ames Research Center. We develop a model of the rover, and demonstrate the approach using simulated scenarios.

### 5.1. Rover Modeling

The rover model was originally presented in (Balaban et al., 2011). In this section we summarize the main features and include some extensions to the model.

The rover consists of a symmetric rigid frame with four independently-driven wheels. The wheel speeds are governed by

$$\dot{\omega}_{FL} = \frac{1}{J_{FL}} (\tau_{mFL} - \tau_{fFL} - \tau_{glFL} + \tau_{grFL}) \quad (c_1)$$

$$\dot{\omega}_{FR} = \frac{1}{J_{FR}} (\tau_{mFR} - \tau_{fFR} - \tau_{glFR} - \tau_{grFR}) \quad (c_2)$$

$$\dot{\omega}_{BL} = \frac{1}{J_{BL}} (\tau_{mBL} - \tau_{fBL} - \tau_{glBL} + \tau_{grBL}) \quad (c_3)$$

$$\dot{\omega}_{BR} = \frac{1}{J_{BR}} (\tau_{mBR} - \tau_{fBR} - \tau_{glBR} - \tau_{grBR}) \quad (c_4)$$

The  $F$ ,  $B$ ,  $L$ , and  $R$  subscripts stand for *front*, *left*, *back*, and *right*, respectively. Here, for wheel  $w \in \{FL, FR, BL, BR\}$ ,  $J_w$  denotes the wheel inertia;  $\tau_{mw} = k_\tau i_w$  is the motor torque, where  $i_w$  is the motor current and  $k_\tau$  is an energy transformation gain;  $\tau_{fw} = \mu_{fw} \omega_w$  is the wheel friction torque, where  $\mu_{fw}$  is a friction coefficient;  $\tau_{glw} = r_w \mu_{gls} (v_w - v)$  is the torque due to slippage, where  $r_w$  is the wheel radius,  $\mu_{gls}$  is a friction coefficient,  $v_w$  is the translational wheel velocity, and  $v$  is the translation velocity of the rover body; and  $\tau_{grw} = r_w \mu_{grw} \omega \cos \gamma$  is the torque due to the rotational movement of the rover body, where  $\mu_{grw}$  is a friction coefficient,  $\omega$  is the rotational velocity of the rover body, and  $\gamma = \arctan l/b$  with  $l$  being the rover length and  $b$  being its width.

We consider here friction-based damage progression in the motors, resulting in an increase in motor friction over time, which will lead to an increase in power consumption. For wheel  $w$ ,  $\mu_{fw}$  is governed by (Daigle & Goebel, 2011b)

$$\dot{\mu}_{fFL} = \nu_{fFL} \mu_{fFL} \omega_{FL}^2 \quad (c_5)$$

$$\dot{\mu}_{fFR} = \nu_{fFR} \mu_{fFR} \omega_{FR}^2 \quad (c_6)$$

$$\dot{\mu}_{fBL} = \nu_{fBL} \mu_{fBL} \omega_{BL}^2 \quad (c_7)$$

$$\dot{\mu}_{fBR} = \nu_{fBR} \mu_{fBR} \omega_{BR}^2, \quad (c_8)$$

where for wheel  $w$ ,  $\nu_{fw}$  is an unknown wear coefficient.

The translational velocity  $v$  of the rover is described by

$$\dot{v} = \frac{1}{m} (F_{glFL} + F_{glFR} + F_{glBL} + F_{glBR}), \quad (c_9)$$

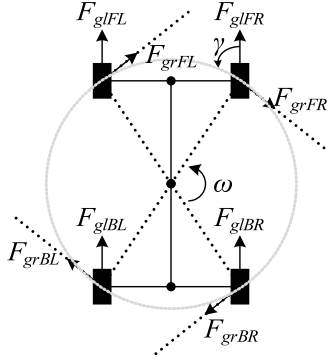


Figure 2. Rover forces.

where  $m$  is the rover mass, and for wheel  $w$ ,  $F_{glw} = \mu_{glw}(v_w - v)$  is the force due to slippage. The rotational velocity  $\omega$  is described by

$$\begin{aligned} \dot{\omega} = & \frac{1}{J} (d \cos \gamma F_{glFR} + d \cos \gamma F_{glBR} - d \cos \gamma F_{glFL} \\ & - d \cos \gamma F_{glBL} - d F_{grFL} - d F_{grFR} - d F_{grBL} \\ & - d F_{grBR}). \end{aligned} \quad (c10)$$

Here,  $J$  is the rotational inertia of the rover,  $d$  is the distance from the center of the rover to each wheel, and for wheel  $w$ ,  $F_{grw} = \mu_{grw}\omega$  is the force due to the rotational movement of the rover body. The rover forces are illustrated in Fig. 2.

The wheels are driven by DC motors with PI control that sets the voltages  $V$  applied to the motors. The motor currents are governed by

$$\dot{i}_{FL} = \frac{1}{L} (V_{FL} - i_{FL}R_{FL} - k_{\omega}\omega_{FL}) \quad (c11)$$

$$\dot{i}_{FR} = \frac{1}{L} (V_{FR} - i_{FR}R_{FR} - k_{\omega}\omega_{FR}) \quad (c12)$$

$$\dot{i}_{BL} = \frac{1}{L} (V_{BL} - i_{BL}R_{BL} - k_{\omega}\omega_{BL}) \quad (c13)$$

$$\dot{i}_{BR} = \frac{1}{L} (V_{BR} - i_{BR}R_{BR} - k_{\omega}\omega_{BR}). \quad (c14)$$

Here,  $L$  is the motor inductance,  $k_{\omega}$  is an energy transformation term, and for wheel  $w$ ,  $R$  is the motor resistance. The voltages applied to the motors are determined by the controllers, where for wheel  $w$ ,  $V_w = P * (u_w - \omega_w) + I * e_{iw}$ , where  $P$  is a proportional gain,  $u_w$  is the commanded wheel speed,  $I$  is an integral gain, and  $e_{iw}$  is the integral error term. The integral error terms are governed by

$$\dot{e}_{iFL} = u_{FL} - \omega_{FL} \quad (c15)$$

$$\dot{e}_{iFR} = u_{FR} - \omega_{FR} \quad (c16)$$

$$\dot{e}_{iBL} = u_{BL} - \omega_{BL} \quad (c17)$$

$$\dot{e}_{iBR} = u_{BR} - \omega_{BR}. \quad (c18)$$

The motor windings heat up as current passes through them.

The temperature of the windings for the motors are governed by

$$\dot{T}_{dFL} = \frac{1}{J_d} (i_{FL}^2 R - h_{dFL}(T_{dFL} - T_{mFL})) \quad (c19)$$

$$\dot{T}_{dFR} = \frac{1}{J_d} (i_{FR}^2 R - h_{dFR}(T_{dFR} - T_{mFR})) \quad (c20)$$

$$\dot{T}_{dBL} = \frac{1}{J_d} (i_{BL}^2 R - h_{dBL}(T_{dBL} - T_{mBL})) \quad (c21)$$

$$\dot{T}_{dBR} = \frac{1}{J_d} (i_{BR}^2 R - h_{dBR}(T_{dBR} - T_{mBR})), \quad (c22)$$

where  $J_d$  is the thermal inertia of the windings, and for wheel  $w$ ,  $h_{dw}$  is a heat transfer coefficient, and  $T_{mw}$  is the motor surface temperature. It is assumed that heat is lost only to the motor surface, and that winding resistance  $R$  is approximately constant for the temperature range considered. The surface temperature of the motor for wheel  $w$  is given by

$$\dot{T}_{mFL} = \frac{1}{J_s} (h_{dFL}(T_{dFL} - T_{mFL}) - h_{aFL}(T_{mFL} - T_a)) \quad (c23)$$

$$\dot{T}_{mFR} = \frac{1}{J_s} (h_{dFR}(T_{dFR} - T_{mFR}) - h_{aFR}(T_{mFR} - T_a)) \quad (c24)$$

$$\dot{T}_{mBL} = \frac{1}{J_s} (h_{dBL}(T_{dBL} - T_{mBL}) - h_{aBL}(T_{mBL} - T_a)) \quad (c25)$$

$$\dot{T}_{mBR} = \frac{1}{J_s} (h_{dBR}(T_{dBR} - T_{mBR}) - h_{aBR}(T_{mBR} - T_a)), \quad (c26)$$

where  $J_s$  is the thermal inertia of the motor surface, and for wheel  $w$ ,  $h_{aw}$  is a heat transfer coefficient, and  $T_a$  is the ambient temperature. Heat is transferred from the windings to the surface and lost to the environment.

The batteries, which are connected in series, are described by a simple electrical circuit equivalent model that includes a large capacitance  $C_b$  in parallel with a resistance  $R_p$ , together in series with another resistance  $R_s$ .<sup>3</sup> The battery charge variables  $q_i$  are governed by

$$\dot{q}_1 = -V_1/R_{p1} - (i_{FL} + i_{FR} + i_{BR} + i_{BL}) \quad (c27)$$

$$\dot{q}_2 = -V_2/R_{p2} - (i_{FL} + i_{FR} + i_{BR} + i_{BL}) \quad (c28)$$

$$\dot{q}_3 = -V_3/R_{p3} - (i_{FL} + i_{FR} + i_{BR} + i_{BL}) \quad (c29)$$

$$\dot{q}_4 = -V_4/R_{p4} - (i_{FL} + i_{FR} + i_{BR} + i_{BL}). \quad (c30)$$

<sup>3</sup>We use a simple model here only for demonstration purposes. More detailed battery models for prognostics can be found in the literature, e.g., (Saha & Goebel, 2009).

Submodel	$X_i$	$\Theta_i$	$U_i$	$Y_i$	$C_i$
$\mathcal{M}_1$	$q_1$	$C_{b1}, R_{s1}$	$i_b^*$	$V_1^*$	$C_{27}, C_{31}, C_{35}$
$\mathcal{M}_2$	$q_2$	$C_{b2}, R_{s2}$	$i_b^*$	$V_2^*$	$C_{28}, C_{32}, C_{35}$
$\mathcal{M}_3$	$q_3$	$C_{b3}, R_{s3}$	$i_b^*$	$V_3^*$	$C_{29}, C_{33}, C_{35}$
$\mathcal{M}_4$	$q_4$	$C_{b4}, R_{s4}$	$i_b^*$	$V_4^*$	$C_{30}, C_{34}, C_{35}$
$\mathcal{M}_5$	$T_{dFL}, T_{mFL}$	$h_{dFL}, h_{aFL}$	$i_{FL}^*$	$T_{mFL}^*$	$C_{19}, C_{23}, C_{36}, C_{40}$
$\mathcal{M}_6$	$T_{dFR}, T_{mFR}$	$h_{dFR}, h_{aFR}$	$i_{FR}^*$	$T_{mFR}^*$	$C_{20}, C_{24}, C_{37}, C_{41}$
$\mathcal{M}_7$	$T_{dBL}, T_{mBL}$	$h_{dBL}, h_{aBL}$	$i_{BL}^*$	$T_{mBL}^*$	$C_{21}, C_{25}, C_{38}, C_{42}$
$\mathcal{M}_8$	$T_{dBR}, T_{mBR}$	$h_{dBR}, h_{aBR}$	$i_{BR}^*$	$T_{mBR}^*$	$C_{22}, C_{26}, C_{39}, C_{43}$
$\mathcal{M}_9$	$i_{FL}, e_{iFL}$	$\emptyset$	$u_{FL}, \omega_{FL}^*$	$i_{FL}^*$	$C_{11}, C_{15}, C_{36}, C_{44}$
$\mathcal{M}_{10}$	$i_{FR}, e_{iFR}$	$\emptyset$	$u_{FR}, \omega_{FR}^*$	$i_{FR}^*$	$C_{12}, C_{16}, C_{37}, C_{45}$
$\mathcal{M}_{11}$	$i_{BL}, e_{iBL}$	$\emptyset$	$u_{BL}, \omega_{BL}^*$	$i_{BL}^*$	$C_{13}, C_{17}, C_{38}, C_{46}$
$\mathcal{M}_{12}$	$i_{BR}, e_{iBR}$	$\emptyset$	$u_{BR}, \omega_{BR}^*$	$i_{BR}^*$	$C_{14}, C_{18}, C_{39}, C_{47}$
$\mathcal{M}_{13}$	$\omega_{FL}, v, \omega, \mu_{fFL}$	$\nu_{fFL}$	$i_{FL}^*, \omega_{FR}^*, \omega_{BL}^*, \omega_{BR}^*$	$\omega_{FL}^*$	$C_1, C_5, C_9, C_{10}, C_{36}, C_{45}, C_{46}, C_{47}$
$\mathcal{M}_{14}$	$\omega_{FR}, v, \omega, \mu_{fFR}$	$\nu_{fFR}$	$i_{FR}^*, \omega_{FL}^*, \omega_{BL}^*, \omega_{BR}^*$	$\omega_{FR}^*$	$C_2, C_6, C_9, C_{10}, C_{37}, C_{44}, C_{46}, C_{47}$
$\mathcal{M}_{15}$	$\omega_{BL}, v, \omega, \mu_{fBL}$	$\nu_{fBL}$	$i_{BL}^*, \omega_{FL}^*, \omega_{FR}^*, \omega_{BR}^*$	$\omega_{BL}^*$	$C_3, C_7, C_9, C_{10}, C_{38}, C_{44}, C_{45}, C_{47}$
$\mathcal{M}_{16}$	$\omega_{BR}, v, \omega, \mu_{fBR}$	$\nu_{fBR}$	$i_{BR}^*, \omega_{FL}^*, \omega_{FR}^*, \omega_{BL}^*$	$\omega_{BR}^*$	$C_4, C_8, C_9, C_{10}, C_{39}, C_{44}, C_{45}, C_{46}$

Table 1. Estimation Submodels

The available sensors measure the voltages of the batteries,

$$V_1^* = q_1/C_{b1} - R_{s1} * (i_{FL} + i_{FR} + i_{BR} + i_{BL}) \quad (C31)$$

$$V_2^* = q_2/C_{b2} - R_{s2} * (i_{FL} + i_{FR} + i_{BR} + i_{BL}) \quad (C32)$$

$$V_3^* = q_3/C_{b3} - R_{s3} * (i_{FL} + i_{FR} + i_{BR} + i_{BL}) \quad (C33)$$

$$V_4^* = q_4/C_{b4} - R_{s4} * (i_{FL} + i_{FR} + i_{BR} + i_{BL}), \quad (C34)$$

the battery current,

$$i_b^* = i_{FL} + i_{FR} + i_{BR} + i_{BL}, \quad (C35)$$

the motor currents,

$$i_{FL}^* = i_{FL} \quad (C36)$$

$$i_{FR}^* = i_{FR} \quad (C37)$$

$$i_{BL}^* = i_{BL} \quad (C38)$$

$$i_{BR}^* = i_{BR}, \quad (C39)$$

the motor surface temperatures,

$$T_{mFL}^* = T_{mFL} \quad (C40)$$

$$T_{mFR}^* = T_{mFR} \quad (C41)$$

$$T_{mBL}^* = T_{mBL} \quad (C42)$$

$$T_{mBR}^* = T_{mBR}, \quad (C43)$$

and the wheel speeds,

$$\omega_{FL}^* = \omega_{FL} \quad (C44)$$

$$\omega_{FR}^* = \omega_{FR} \quad (C45)$$

$$\omega_{BL}^* = \omega_{BL} \quad (C46)$$

$$\omega_{BR}^* = \omega_{BR}. \quad (C47)$$

Here, the \* superscript indicates a measured value.

We are interested in predicting when any of the rover batteries are at their voltage threshold, beyond which the batteries will be damaged (Saha & Goebel, 2009). The constraints are

given as

$$V_1 > V^- \quad (C48)$$

$$V_2 > V^- \quad (C49)$$

$$V_3 > V^- \quad (C50)$$

$$V_4 > V^-, \quad (C51)$$

where the voltage threshold is given by  $V^- = 9.6$  V, and for battery  $i$ ,  $V_i = q_i/C_{bi} - R_{si} * (i_{FL} + i_{FR} + i_{BR} + i_{BL})$ . We are also interested in when the motor temperature gets too high. The motor windings are designed to withstand temperatures up to a certain point, after which, the insulation breaks down, the windings short, and the motor fails (Balaban et al., 2010). The constraints are given as

$$T_{mFL} < T_m^+ \quad (C52)$$

$$T_{mFR} < T_m^+ \quad (C53)$$

$$T_{mBL} < T_m^+ \quad (C54)$$

$$T_{mBR} < T_m^+, \quad (C55)$$

where the temperature limit is given by  $T_m^+ = 70^\circ$  C. The rover cannot be operated when any of these constraints,  $c_{48} - c_{55}$ , are violated.

In the general case, we consider uncertainty in the friction wear parameters  $\nu_{fFL}$ ,  $\nu_{fFR}$ ,  $\nu_{fBL}$ , and  $\nu_{fBR}$ ; the heat transfer coefficients  $h_{dFL}$ ,  $h_{dFR}$ ,  $h_{dBL}$ ,  $h_{dBR}$ ,  $h_{aFL}$ ,  $h_{aFR}$ ,  $h_{aBL}$ , and  $h_{aBR}$ ; the battery capacitances  $C_{b1}$ ,  $C_{b2}$ ,  $C_{b3}$ , and  $C_{b4}$ ; and the battery resistances  $R_{s1}$ ,  $R_{s2}$ ,  $R_{s3}$ , and  $R_{s4}$ . Sensor and process noise were estimated based on data from the actual rover testbed.

## 5.2. Results

To demonstrate the validity of the approach, we describe two scenarios for system-level prognostics of the rover. In the first, the rover is operating nominally without any faults present, and in the second, friction damage is progressing

Submodel	$X_i$	$\Theta_i$	$U_i$	$Y_i$	$C_i$
$\mathcal{M}_{17}$	$q_1$	$C_{b1}, R_{s1}$	$i_{FL}, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_{27}, C_{48}$
$\mathcal{M}_{18}$	$q_2$	$C_{b2}, R_{s2}$	$i_{FL}, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_{28}, C_{49}$
$\mathcal{M}_{19}$	$q_3$	$C_{b3}, R_{s3}$	$i_{FL}, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_{29}, C_{50}$
$\mathcal{M}_{20}$	$q_4$	$C_{b4}, R_{s4}$	$i_{FL}, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_{30}, C_{51}$
$\mathcal{M}_{21}$	$T_{dFL}, T_{mFL}$	$h_{dFL}, h_{aFL}$	$i_{FL}$	$\emptyset$	$C_{19}, C_{23}, C_{52}$
$\mathcal{M}_{22}$	$T_{dFR}, T_{mFR}$	$h_{dFR}, h_{aFR}$	$i_{FR}$	$\emptyset$	$C_{20}, C_{24}, C_{53}$
$\mathcal{M}_{23}$	$T_{dBL}, T_{mBL}$	$h_{dBL}, h_{aBL}$	$i_{BL}$	$\emptyset$	$C_{21}, C_{25}, C_{54}$
$\mathcal{M}_{24}$	$T_{dBR}, T_{mBR}$	$h_{dBR}, h_{aBR}$	$i_{BR}$	$\emptyset$	$C_{22}, C_{26}, C_{55}$

Table 2. Prediction Submodels Using Motor Currents as Local Inputs

on one motor. In both cases, the rover travels between various waypoints, moving at an average speed of 0.5 m/s. The unknown parameters are initialized incorrectly (with around 10% error) so that the local estimators must converge to the true values. In both cases, the estimation step is performed in a distributed manner using the set of submodels derived by using measured values as local inputs, shown in Table 1. For example, submodel  $\mathcal{M}_1$  computes an estimate of  $V_1^*$  using the measured value of  $i_b^*$  as a local input, and using the minimal set of constraints to do this. For the prediction submodels, as will be shown, the correct submodels to use depends on the scenario, and illustrates when and when not the prediction step can be decomposed.

### 5.2.1. Nominal Operation

We first consider a scenario involving nominal, fault-free operations. In this case, EOL occurs around 3 h. An RUL prediction is made every 500 s. With the rover traveling at an average speed of 0.5 m/s, the motor currents average to about 0.15 A each and so the total current draining from the four batteries is 0.6 A. Since these values do not vary much during nominal operation, we can use the motor currents as local inputs for the model decomposition. These submodels are shown in Table 2. Note that the estimates from the estimation submodels  $\mathcal{M}_1$ – $\mathcal{M}_8$  are used directly in the prediction submodels  $\mathcal{M}_{17}$ – $\mathcal{M}_{24}$ , respectively, and that estimation submodels  $\mathcal{M}_9$ – $\mathcal{M}_{16}$  are not necessary. Note also that the prediction submodels do not compute any outputs, rather, their goal is to compute EOL constraints (e.g.,  $\mathcal{M}_{17}$  computes  $c_{48}$ ).

The system-level prediction results are shown in Fig. 3. Predictions from the battery submodels are shown in Fig. 4. In this case, the motor temperatures reach a steady-state that is below the temperature threshold, so only the batteries have an impact on system EOL, which is the minimum of the EOLs predicted for the battery submodels. In particular, it is the first and fourth batteries (corresponding to  $\mathcal{M}_{17}$  and  $\mathcal{M}_{20}$ , respectively) that discharge the fastest, as shown explicitly in Fig. 4. The figures show the means of the predicted RUL distributions, the true RUL,  $RUL^*$ , and an accuracy cone of  $\alpha = 10\%$  around it. In Fig. 3, we show both the system-level predictions using the distributed approach with  $\mathcal{M}_{17}$ – $\mathcal{M}_{24}$  and the centralized approach using the global prediction model  $\mathcal{M}_0$ . The global prediction model contains all

the states, parameters, and constraints given in the previous subsection, minus the output constraints, and uses the commanded wheel speeds (known a priori) as hypothesized inputs. Since the currents are also known a priori, the system-level prediction can be decomposed, and the predictions made using the local submodels closely match those made using the global model, as shown in the figure.

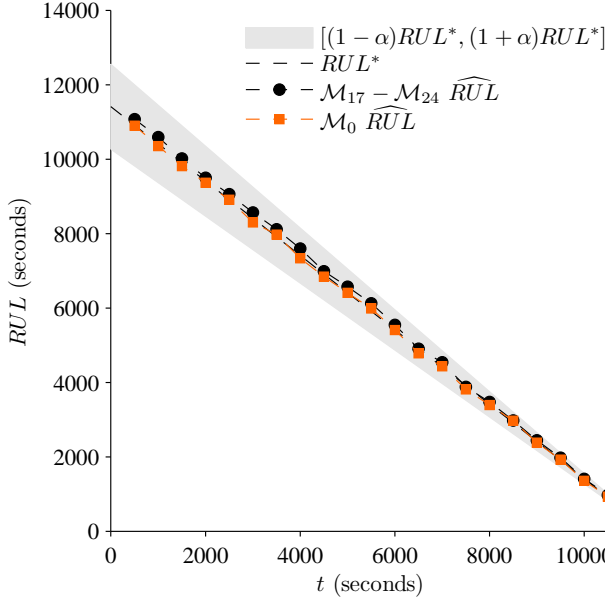
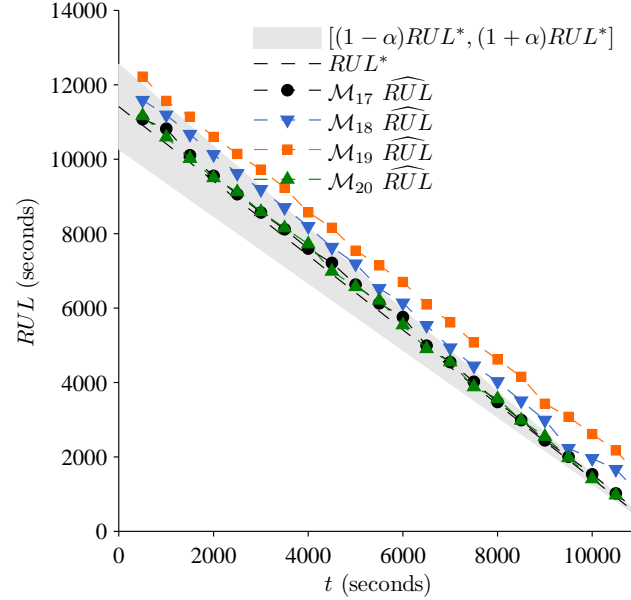
We use the relative accuracy (RA) metric (Saxena et al., 2010) for prediction accuracy. Averaged over all predictions, RA is 97.48% for the distributed approach and 98.74% for the centralized approach. Using relative standard deviation (RSD) as a measure of spread, and averaged over all prediction points, RSD is 0.40% for the distributed approach and 0.43% for the centralized approach. The distributed approach is only slightly less accurate but has better precision. Here, both approaches are very accurate since the system state-parameter estimates are very accurate, and there is only a small amount of error associated with assuming a constant average motor current or wheel speed. Correspondingly, the prediction spread is relatively small because the uncertainty in the state-parameter estimate is very small.

### 5.2.2. Friction Damage Progression

We now consider a scenario in which for the front-left motor, there is nonlinear friction damage progression with  $\nu_{fFL} = 1 \times 10^{-4}$  s. As a result of the continuously increasing friction, the current drawn by the motor increases as well in order for the motor controller to maintain the same desired wheel speed. Hence, the total current drawn from the batteries is increased, and EOL occurs around 2 h. Because  $i_{FL}$  is constantly changing, and in a way that is dependent on the motor state, it cannot be predicted *a priori*, and so cannot be used as a local input because the resulting predictions will not be accurate. Therefore, we require a submodel that estimates  $i_{FL}$ , and we so employ submodels using as local inputs average values for the remaining motor currents, average commanded wheel speeds, and average rover translational velocity  $v$  and rotational velocity  $\omega$ . The prediction submodels for this case are shown in Table 3. For comparison, we demonstrate also prediction using  $\mathcal{M}_{17}$ – $\mathcal{M}_{24}$ , and, for this strategy, at each prediction point the average value of current measured over the last minute is used as the future hypothesized value. Of course, this will not yield accurate results since future values

Submodel	$X_i$	$\Theta_i$	$U_i$	$Y_i$	$C_i$
$\mathcal{M}_{25}$	$q_1, i_{fFL}, e_{iFL}, \omega_{FL}, \mu_{fFL}$	$C_{b1}, R_{s1}, \nu_{fFL}$	$u_{FL}, v, \omega, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_1, C_5, C_{11}, C_{15}, C_{27}, C_{48}$
$\mathcal{M}_{26}$	$q_2, i_{fFL}, e_{iFL}, \omega_{FL}, \mu_{fFL}$	$C_{b2}, R_{s2}, \nu_{fFL}$	$u_{FL}, v, \omega, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_1, C_5, C_{11}, C_{15}, C_{28}, C_{49}$
$\mathcal{M}_{27}$	$q_3, i_{fFL}, e_{iFL}, \omega_{FL}, \mu_{fFL}$	$C_{b3}, R_{s3}, \nu_{fFL}$	$u_{FL}, v, \omega, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_1, C_5, C_{11}, C_{15}, C_{29}, C_{50}$
$\mathcal{M}_{28}$	$q_4, i_{fFL}, e_{iFL}, \omega_{FL}, \mu_{fFL}$	$C_{b4}, R_{s4}, \nu_{fFL}$	$u_{FL}, v, \omega, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_1, C_5, C_{11}, C_{15}, C_{30}, C_{51}$
$\mathcal{M}_{29}$	$T_{dFL}, T_{mFL}, i_{FL}, e_{iFL}, \omega_{FL}, \mu_{fFL}$	$h_{dFL}, h_{aFL}, \nu_{fFL}$	$u_{FL}, v, \omega$	$\emptyset$	$C_{19}, C_{23}, C_{52}, C_{11}, C_{15}, C_1, C_5$
$\mathcal{M}_{30}$	$T_{dFR}, T_{mFR}$	$h_{dFR}, h_{aFR}$	$i_{FR}$	$\emptyset$	$C_{20}, C_{24}, C_{53}$
$\mathcal{M}_{31}$	$T_{dBL}, T_{mBL}$	$h_{dBL}, h_{aBL}$	$i_{BL}$	$\emptyset$	$C_{21}, C_{25}, C_{54}$
$\mathcal{M}_{32}$	$T_{dBR}, T_{mBR}$	$h_{dBR}, h_{aBR}$	$i_{BR}$	$\emptyset$	$C_{22}, C_{26}, C_{55}$

Table 3. Prediction Submodels Using Commanded Wheel Speeds and Rover Velocities as Local Inputs


 Figure 3. System RUL prediction performance under nominal conditions with  $\alpha = 0.1$ .

 Figure 4. Individual battery submodel RUL prediction performance under nominal conditions with  $\alpha = 0.1$ .

of the current will actually be larger. Note that the prediction submodels used in this case do not correspond directly to those used for estimation. So, when constructing the estimate for  $\mathcal{M}_{25}$ , for example, it takes the estimates from  $\mathcal{M}_1$ ,  $\mathcal{M}_9$ , and  $\mathcal{M}_{13}$ .

The system-level prediction results are shown in Fig. 5. Although the increased friction causes the temperature of the front-left motor to increase, it is still the batteries discharging that dominates the system-level EOL in this case. We show the predictions using  $\mathcal{M}_{17}$ – $\mathcal{M}_{24}$ ,  $\mathcal{M}_{25}$ – $\mathcal{M}_{32}$ , and the global model  $\mathcal{M}_0$ . For  $\mathcal{M}_{25}$ – $\mathcal{M}_{32}$ , average values of  $v = 0.5$  m/s and  $\omega = 0$  rad/s are used. Here, the predictions using the latter two approaches are virtually identical (the predictions using  $\mathcal{M}_{25}$ – $\mathcal{M}_{32}$  are hidden under those for  $\mathcal{M}_0$ ), and fairly accurate. In contrast, as expected, the predictions using  $\mathcal{M}_{17}$ – $\mathcal{M}_{24}$  are very inaccurate, and only converge towards the true RUL at the very end. This quite effectively demonstrates that, in this scenario, it is incorrect to use the front-left motor current as a local input for predictions, since it cannot be predicted independently from the front-left motor submodel,

and therefore a submodel that itself predicts this current is required to obtain accurate predictions.

Here, RA averages to 58.95% using  $\mathcal{M}_{17}$ – $\mathcal{M}_{24}$ , 94.24% using  $\mathcal{M}_{25}$ – $\mathcal{M}_{32}$ , and 94.32% using  $\mathcal{M}_0$ . RSD averages to 0.76% using  $\mathcal{M}_{17}$ – $\mathcal{M}_{24}$ , 1.62% using  $\mathcal{M}_{25}$ – $\mathcal{M}_{32}$ , 1.73% using  $\mathcal{M}_0$ . Here, we also observe an increase in prediction spread using the centralized approach with only a slight increase in accuracy over the distributed approach.<sup>4</sup> Overall, accuracy and precision are both decreased compared to the nominal scenario because there is more uncertainty in the state-parameter estimate, specifically, that dealing with the estimate of  $\nu_{fFL}$ . This uncertainty in the state-parameter estimate contributes to the additional uncertainty in the RUL predictions.

<sup>4</sup>The RSD for  $\mathcal{M}_{17}$ – $\mathcal{M}_{24}$  is the lowest because those submodels do not include the motor friction component, so do not have the additional uncertainty associated with the estimation of the wear parameter.

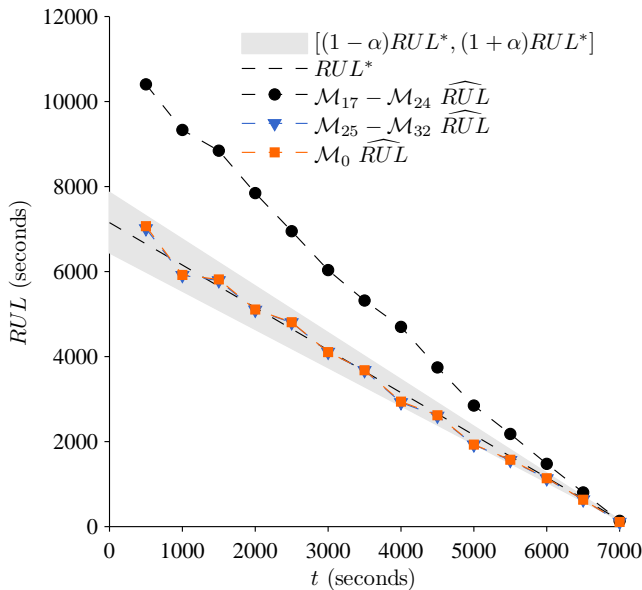


Figure 5. System RUL prediction performance with friction damage progression with  $\alpha = 0.1$ .

## 6. CONCLUSIONS

In this paper, we formulated the system-level prognostics problem and proposed a distributed solution based on structural model decomposition. Using a four-wheeled rover as a simulation-based case study, we demonstrated the effectiveness of the approach. Most importantly, the distributed approach allows for, in many practical circumstances, the decomposition of the system-level prognostics problem into component-level prognostics problems and provides a mechanism to merge local prognostics results into a system-level result. Further, since the local subproblems are independent, this allows component experts to focus on prognostics solutions for their components. However, we showed also that this approach is not always possible if accurate results are desired, since in some cases the prediction problem cannot be so easily decomposed, and it depends crucially on correct assumptions about what variables may serve as local inputs for the prediction problem.

Although in this paper we focused on the model-based prognostics paradigm, our approach is flexible in that data-driven algorithms may be used also, once the local subproblems are defined. For example, in previous works, structural model decomposition was used to automatically design gray box diagnosis models that were implemented using different data-driven techniques (for instance, state space neural networks in (Pulido, Zamarreno, Merino, & Bregon, 2012) or machine learning techniques in (Alonso-Gonzalez, Rodríguez, Prieto, & Pulido, 2008)). By decomposing the system-level problem into independent subproblems through structural model decomposition, we can apply similar ideas to solve each prog-

nostics subproblem by using the most appropriate technique, whether it is a model-based, data-driven, or hybrid approach.

An important direction of future work is in algorithms for optimal placement of sensors for model decomposition, because the level of model decomposition that can be achieved is dependent on the number of sensors and where they are placed. This results in the most efficient decomposition of the system-level prognostics problem. Current work also addresses combining the distributed prognostics framework with a distributed diagnostic approach for integrated diagnostics and prognostics (Bregon, Daigle, & Roychoudhury, 2012).

## ACKNOWLEDGMENTS

M. Daigle and I. Roychoudhury's funding for this work was provided by the NASA System-wide Safety and Assurance Technologies (SSAT) Project. A. Bregon's funding for this work was provided by the Spanish MCI TIN2009-11326 grant.

## REFERENCES

- Alonso-Gonzalez, C. A., Rodríguez, J. J., Prieto, O., & Pulido, B. (2008, September). Machine learning and model-based diagnosis using possible conflicts and system decomposition. In *Proc. of the 19th international workshop on principles of diagnosis* (p. 215-222). Blue Mountains, Australia.
- Balaban, E., Narasimhan, S., Daigle, M., Celaya, J., Roychoudhury, I., Saha, B., et al. (2011, September). A mobile robot testbed for prognostics-enabled autonomous decision making. In *Annual conference of the prognostics and health management society* (p. 15-30). Montreal, Canada.
- Balaban, E., Saxena, A., Narasimhan, S., Roychoudhury, I., Goebel, K., & Koopmans, M. (2010, September). Airborne electro-mechanical actuator test stand for development of prognostic health management systems. In *Annual conference of the prognostics and health management society*.
- Bolander, N., Qiu, H., Eklund, N., Hindle, E., & Rosenfeld, T. (2010, October). Physics-based remaining useful life prediction for aircraft engine bearing prognosis. In *Proceedings of the annual conference of the prognostics and health management society 2010*.
- Bregon, A., Biswas, G., & Pulido, B. (2012, May). A decomposition method for nonlinear parameter estimation in TRANSCEND. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, 42(3), 751-763.
- Bregon, A., Daigle, M., & Roychoudhury, I. (2012, July). An integrated model-based distributed diagnosis and prognosis framework. In *Proceedings of the 23rd interna-*

- tional workshop on principles of diagnosis.*
- Byington, C. S., Watson, M., Edwards, D., & Stoelting, P. (2004, March). A model-based approach to prognostics and health management for flight control actuators. In *Proceedings of the 2004 IEEE Aerospace Conference* (Vol. 6, pp. 3551–3562).
- Celaya, J. R., Kulkarni, C., Biswas, G., Saha, S., & Goebel, K. (2011, September). A model-based prognostics methodology for electrolytic capacitors based on electrical overstress accelerated aging. In *Proceedings of the annual conference of the prognostics and health management society 2011*.
- Daigle, M., Bregon, A., & Roychoudhury, I. (2011, September). Distributed damage estimation for prognostics based on structural model decomposition. In *Proceedings of the annual conference of the prognostics and health management society 2011* (p. 198-208).
- Daigle, M., Bregon, A., & Roychoudhury, I. (2012). *Distributed prognostics based on structural model decomposition*. (Manuscript submitted for publication.)
- Daigle, M., & Goebel, K. (2010, October). Improving computational efficiency of prediction in model-based prognostics using the unscented transform. In *Proc. of the annual conference of the prognostics and health management society 2010*.
- Daigle, M., & Goebel, K. (2011a, August). A model-based prognostics approach applied to pneumatic valves. *International Journal of Prognostics and Health Management*, 2(2).
- Daigle, M., & Goebel, K. (2011b, March). Multiple damage progression paths in model-based prognostics. In *Proceedings of the 2011 IEEE Aerospace Conference*.
- Daigle, M., Saha, B., & Goebel, K. (2012, March). A comparison of filter-based approaches for model-based prognostics. In *Proceedings of the 2012 IEEE Aerospace Conference*.
- Julier, S. J., & Uhlmann, J. K. (1997). A new extension of the Kalman filter to nonlinear systems. In *Proceedings of the 11th international symposium on aerospace/defense sensing, simulation and controls* (pp. 182–193).
- Julier, S. J., & Uhlmann, J. K. (2004, March). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3), 401–422.
- Liu, J., & West, M. (2001). Combined parameter and state estimation in simulation-based filtering. *Sequential Monte Carlo Methods in Practice*, 197–223.
- Luo, J., Pattipati, K. R., Qiao, L., & Chigusa, S. (2008, September). Model-based prognostic techniques applied to a suspension system. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 38(5), 1156–1168.
- Moya, N., Biswas, G., Alonso-Gonzalez, C. J., & Koutsoukos, X. (2010, October). Structural observability: Application to decompose a system with possible conflicts. In *Proceedings of the 21st international workshop on principles of diagnosis* (p. 241-248).
- Mutambara, A. G. (1998). *Decentralized estimation and control for multisensor systems*. Boca Raton: CRC Press.
- Orchard, M., Tobar, F., & Vachtsevanos, G. (2009, December). Outer feedback correction loops in particle filtering-based prognostic algorithms: Statistical performance comparison. *Studies in Informatics and Control*(4), 295-304.
- Orchard, M., & Vachtsevanos, G. (2009, June). A particle filtering approach for on-line fault diagnosis and failure prognosis. *Transactions of the Institute of Measurement and Control*(3-4), 221-246.
- Pulido, B., & Alonso-González, C. (2004). Possible conflicts: a compilation technique for consistency-based diagnosis. *IEEE Trans. on Systems, Man, and Cybernetics, Part B, Special Issue on Diagnosis of Complex Systems*, 34(5), 2192-2206.
- Pulido, B., Zamarreno, J., Merino, A., & Bregon, A. (2012, July). *Using structural decomposition methods to design gray-box models for fault diagnosis of complex industrial systems: a beet sugar factory case study*.
- Saha, B., & Goebel, K. (2009, September). Modeling Li-ion battery capacity depletion in a particle filtering framework. In *Proceedings of the annual conference of the prognostics and health management society 2009*.
- Saha, B., Saha, S., & Goebel, K. (2009). A distributed prognostic health management architecture. In *Proceedings of the 2009 conference of the society for machinery failure prevention technology*.
- Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2010). Metrics for offline evaluation of prognostic performance. *International Journal of Prognostics and Health Management*, 1(1).
- Sinha, A., Chen, H., Danu, D., Kirubarajan, T., & Farooq, M. (2008). Estimation and decision fusion: A survey. *Neurocomputing*, 71(13), 2650–2656.

# Peer-to-peer Collaborative Vehicle Health Management – the Concept and an Initial Study

Yilu Zhang, Xinyu Du, and Mutasim Salman

*Electrical & Controls Integration Lab, GM Global R&D, General Motors Company, Warren, MI 48090, USA*

*yilu.zhang@gm.com*

*xinyu.du@gm.com*

*mutasim.salman@gm.com*

## ABSTRACT

Advanced vehicle diagnostics and prognostics (D&P) technology enhances ownership experience, and reduces corporate warranty cost. D&P performance optimization requires significant algorithm tuning and a large amount of test data collection, which is resource consuming.

In this paper, we propose a novel D&P framework called Collaborative Vehicle Health Management (CVHM) to automatically optimize the D&P algorithms on a host vehicle, using the field data collected from peer vehicles encountered on the road. The carefully designed system architecture and learning algorithms enhance D&P performance without costly human intervention. The experimental results on battery remaining useful life prediction show the effectiveness of the proposed framework. This proposed framework has been implemented in a small test fleet as a proof-of-concept prototype.

## 1. INTRODUCTION

Diversified passenger vehicle usage leads to diversified vehicle system failure modes and aging processes. As a result, it is very challenging to achieve accurate and robust diagnostic and prognostic (D&P) performance for vehicle systems in the field. In the state-of-the-art practice of D&P algorithm development, a large amount of data has to be collected through fault injection on bench or test vehicles for diagnostics, or through accelerated ageing tests for prognostics. And a significant amount of algorithm tuning work has to be done by development engineers.

Motivated by this challenge, we propose a novel D&P framework called, Collaborative Vehicle Health Management (CVHM), where field data from peer vehicles are aggregated to automatically optimize the D&P algorithms for the host vehicle. This is an extension of the decade-long evolving research and development in the area of remote vehicle diagnostics (Millstein, 2002) (Kuschel, 2004) (Carr, 2005) (You, Krage, & Jalics, 2005) (Zoja, 2006) (Zhang, Grantt, Rychlinski, Edwards, Correia, & Wolf, 2009) (Byttner, Rögnavaldsson, Svensson, Bitar, & Chominsky, 2009). Three key enablers are needed to realize CVHM,

1. An onboard CVHM architecture that facilitates peer vehicle data aggregation, and host vehicle D&P algorithm adaptation
2. Intelligent data modeling and statistical decision making technologies that allow the extraction of fault signature, failure precursor, trending information, and other actionable knowledge to enhance the D&P performance.
3. A heterogeneous wireless communication solution that combines cellular network, and opportunistic V2V (vehicle-to-vehicle) communication to allow the exchange of large-volume data between vehicles in a cost-effective way.

In this paper, we present the latest development in the first two items above, using battery remaining useful life as the example application. The reader is referred to (Bai, Grimm, Talty, & Saraydar, 2011) for the background of item 3.

This paper is organized as follows. The proposed CVHM architecture is introduced in Section 2, followed by the development of the prognostic algorithms in Section 3. Section 4 discusses the system implementation. Section 5 presents the experimental results. Section 6 discusses future works.

---

Yilu Zhang et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.



## 2. COLLABORATIVE VEHICLE HEALTH MANAGEMENT SYSTEM ARCHITECTURE

A typical vehicle health management system architecture is illustrated in Fig. 1. Sensor information regarding particular vehicle subsystem is either directly collected by the VHM ECU that runs D&P algorithms or is transferred from other ECUs through an in-vehicle communication network. Note that, in real implementations, the VHM ECU may be implemented as a functional module within an ECU, such as a body control module (BCM), that executes control functions. The D&P module has various D&P algorithms for different targeted vehicle components or subsystems, such as battery, electrical power generation and storage (EPGS) system, fuel delivery system, etc. The D&P module processes the sensor information, and generates D&P results, including the detected anomalies, isolated faulty components, and the predicted remaining useful life (RUL) of related components. The D&P algorithms are usually developed, calibrated, and tested through a sophisticated vehicle development process. Once the vehicle is released for production, the D&P algorithms and the associated calibration values are usually fixed. If major updates on the onboard algorithms are needed, an ECU reprogramming can be done after the vehicle is usually called to a dealer service shop. Lately, the technology of remote ECU refresh is maturing, which may allow the ECU reprogramming to be done remotely through telematics connections.

The proposed CVHM system, as shown in Fig. 2, is built upon the existing VHM system architecture. The newly added V2X ECU provides the wireless communication

interface in order to exchange vehicle health related data between the host vehicle and peer vehicles. V2X represents vehicle-to-vehicle or vehicle to infrastructure. The V2X ECU stores the data in an onboard database. The VHM ECU has an algorithm adaptation module and a learning algorithm library, in addition to the regular D&P module. The algorithm adaptation module makes use of appropriate learning algorithms to process the vehicle health related data stored in the onboard database in order to tune and optimize the calibration values within the D&P module.

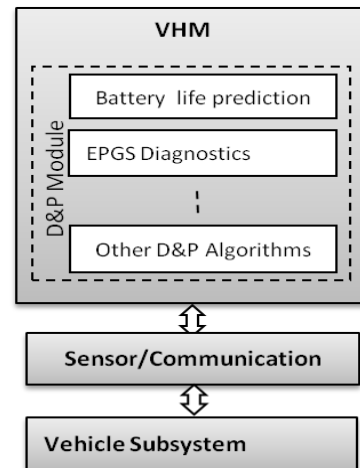


Fig. 1: A typical VHM system architecture in the state-of-the-art

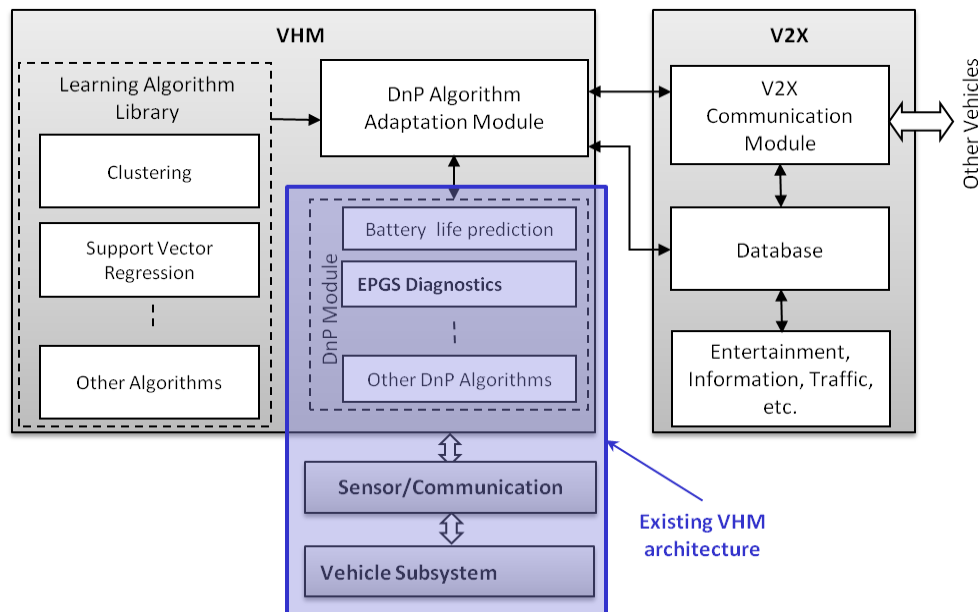


Fig. 2: Proposed CVHM system architecture

The advantage of CVHM can be understood based on the following example. A battery life prediction algorithm usually implements an ageing model that specifies how the battery internal resistance grows given the number of charge-discharge cycles. There are parameters in the ageing model that specifies the growth rate of the battery internal resistance, which is critical in battery life prediction. These parameters are typically calibrated using accelerated ageing test during the vehicle development process, and applied to across the board to all vehicles. However, it is difficult for a pre-calibrated model to account for the intrinsic diversity of usage patterns and environment impacts. The fact is that batteries for the same battery/vehicle model may have different life span that ranges from 1 year to 10+ years. At the same time, with large enough vehicle population, for any given vehicle, chance is high that there are peer vehicles with similar usage profiles that have been used for longer time, and therefore have gone further ahead in the ageing process. With CVHM, field data from these peer vehicles can be used to fine tune the growth rate in the battery ageing model, and consequently achieve higher prediction performance.

### 3. ALGORITHM DEVELOPEMNT

The general framework to develop model-based prognostics for remaining useful life (RUL) prediction involves the following steps.

First, one or more fault signatures are identified to characterize target system's state of health,  $Z = f(\text{SOH})$ . Depending on applications, these fault signatures may be assessed either directly or indirectly. For example, in the application of SLI (Starting, Light, Ignition) battery life prediction, multiple fault signatures have been proposed (Zhang, Grube, Shin, & Salman, 2008) (Zhang, Grube, Shin, & Salman, 2009) (Shin & Salman, 2010). Some of them can be directly measured by onboard sensors, *e.g.*, minimum cranking voltage. Some of them can be directly calculated from other sensor measurements, *e.g.*, cranking resistance can be calculated by  $dV/dI$ , where  $dV$  and  $dI$  are voltage and current changes in the beginning of the cranking process, respectively. There are also fault signatures that cannot be directly measured, and have to be estimated as the parameters in a system model, *e.g.*, battery capacity.

The second step is to establish the failure criteria for fault signatures with respect to specific applications. That is, if  $Z > Z_0$ , a system failure is declared, where  $Z_0$  is a threshold. For example, one of the main functions for SLI battery is to crank the engine. As battery ages, its SOH deteriorates, and so does its cranking capability. One of the fault signatures, cranking resistance, increases during the ageing process. When the cranking resistance reaches certain level, the engine can hardly be started. This is when a battery failure is declared. The failure criteria are highly application specific, and usually require careful calibration.

The third step is to establish a system-ageing model that specifies how the fault signatures evolve with respect to usage. That is,

$$Z = Z(L; \theta),$$

where  $L$  is a set of variables that characterize the usage profile of the target system, and  $\theta$  is a set of parameters that specify the detailed relationship between the usage and the fault signature evolution.

The CVHM framework follows the above general model-based prognostics framework. The main enhancement is that the system ageing model is updated as more data is made available from peer vehicles. In the next few sections, we take battery RUL prognosis as an example application to illustrate the development and implementation of the CVHM framework.

#### 3.1. Fault signature generation algorithms

Extensive previous research has been conducted, and multiple SLI battery fault signatures have been identified, including minimum cranking voltage, delta V, cranking power, voltage residual, and cranking resistance (Zhang, Grube, Shin, & Salman, 2008) (Zhang, Grube, Shin, & Salman, 2009) (Shin & Salman, 2010). A brief description of these fault signatures are listed Table 1. These fault signatures change along with the battery age. For instance, the cranking resistance increases in an accelerated ageing experiment, as shown in Fig. 3. It's worth noting that the battery capacity is also an effective signature. However, it is difficult to be estimated accurately online.

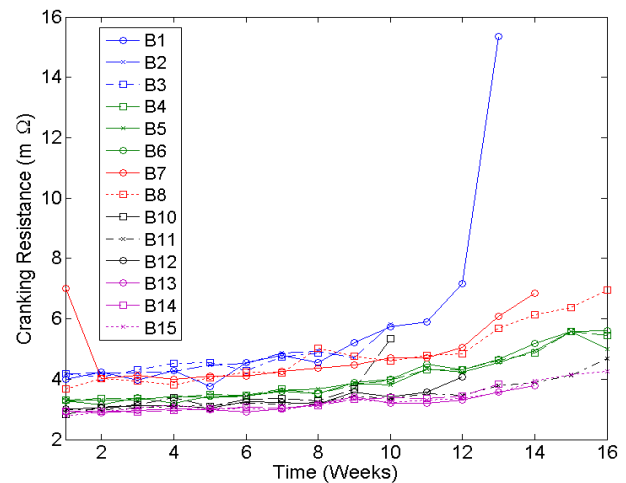


Fig. 3 Cranking resistance change during the accelerated ageing test for 14 batteries from JBI\_Aging\_2008 data set (battery 9 does not have data) in the conditions of 100% SOC and 25°C

Battery Fault Signatures	Formula
Minimum cranking voltage	$V_{min} = \min(V_{batt})$
Delta V: the difference between the first and second minimum voltage	$V_{delta} = V_{min1} - V_{min2}$ Where $V_{min1}$ and $V_{min2}$ are the first and second minimum voltages, respectively
Cranking power	$P = \int_T^{T+0.5} V_{batt} I_{batt} dt$ where $T = t$ when $I > 100A$
Voltage residual	$V_R = \frac{\sum_{i=1}^n (V_{batt}(i) - \hat{V}_{batt}(i))}{n}$ where $\hat{V}_{batt}$ is estimated by calibrated model from good batteries
Cranking resistance	$R = \frac{V_{batt0} - V_{min}}{I_{batt0} - I_{max}}$ where $I_{max} = \max(I_{batt})$

Table 1: Battery fault signatures

### 3.2. Prognosis algorithm with adaptation

While fault signatures indicate the current status of the fault, failure prediction requires an ageing model to depict how the fault signatures evolve as the battery ages. Multiple ageing models have been proposed in the literature. Some of them are physics-based models, considering either specific ageing mechanism of battery (Schiffer, Sauer, Bindner, Cronin, Lundsager, & Kaiser, 2007), or general ageing laws for mechanical or electro-chemical systems (Edwin, Chiang, Carter, Limthongkul, & Bishop, 2005). In reality, these models are more or less hybrids of empirical and physics-based models that have many model parameters fitted through experiments. Other models are purely data driven based on various linear or non-linear curve fitting techniques (Saha, Poll, & Christophersen, 2009). Due to the intrinsic complexity of the battery aging process, there is no clear winner in the proposed ageing models in terms of prediction accuracy. In this research, we adopted a few static parametric models, including polynomial curve fitting, exponential curve fitting, and support vector machine (Vapnik, 1998). There was not significant difference between these models in our experimental results. We present the algorithm development based on a 3rd order polynomial model due to its structural simplicity.

Each fault signature is modeled by the following equation:

$$\hat{y}(t) = p_1 t^3 + p_2 t^2 + p_3 t + p_4$$

where  $\hat{y}$  is predicted fault signature value,  $t$  is the battery age in terms of service time, and  $p_1, p_2, p_3$  and  $p_4$  are model parameters. Since both SOC and battery temperature can affect battery fault signature, different models have to be learned for different SOC and temperatures. The battery RUL is defined as

$$RUL = \arg \min_t [\hat{y}(t) = y_0] - t_{current}$$

where  $y_0$  is a predefined threshold, and  $t_{current}$  is the current battery age.

As discussed in Section 2, the ageing model calibrated with accelerated ageing test may not be able to characterize the ageing process in the field. In the proposed CVHM, the ageing model is adapted using the data from peer vehicles that have gone further in the ageing process.

Let  $y_H(t_j)$  be the fault signature value measured or estimated by the host vehicle at time instant  $t_j$ , where  $j = 1 \dots J$  and  $J$  is the current time index for the host vehicle. Let  $p_{H,1}, p_{H,2}, p_{H,3}, p_{H,4}$  be the ageing model parameters maintained by host vehicle, and  $p_{P_k,1}, p_{P_k,2}, p_{P_k,3}, p_{P_k,4}$  be the ageing model parameters used by peer vehicle  $P_k$ , where  $k = 1 \dots K$  and  $K$  is the number of peer vehicles. The model adaptation procedure is as follows.

1. Estimate host vehicle fault signature values using peer vehicles' ageing model parameters, which yields,

$$\hat{y}_{H,P_k}(t_j) = p_{P_k,1} t_j^3 + p_{P_k,2} t_j^2 + p_{P_k,3} t_j + p_{P_k,4}$$

where  $\hat{y}_{H,P_k}(t_j)$  indicates the estimate of host vehicle fault signature using the ageing model from peer vehicle  $P_k$ .

2. Calculate the corresponding estimation error for the ageing model from each peer vehicle  $P_k$  as,

$$R_{H,P_k} = \sum_{j=1}^J [\hat{y}_{H,P_k}(t_j) - y_H(t_j)]^2.$$

3. Pick  $N$  models with the smallest error. Without loss of generality, the corresponding peer vehicles can be represented as  $P_{k_1}, P_{k_2}, \dots, P_{k_N}$ . In the experiment presented in this paper,  $N$  is set to 3.

4. Calculate the adjusted host vehicle fault signature values,  $\bar{y}_H(t_j)$ , by averaging the fault signature values based on the selected peer vehicles' ageing models,

$$\bar{y}_H(t_j) = \frac{1}{N} \sum_{n=1}^N \hat{y}_{H,P_{k_n}}(t_j)$$

5. Update the host vehicle ageing model, using the adjusted fault signature values

$$\{p_{H,1}, p_{H,2}, p_{H,3}, p_{H,4}\} = \arg \min_{p_1, p_2, p_3, p_4} \sum_{j=1}^J [\bar{y}_H(t_j) - \hat{y}(t_j)]^2$$

where  $\hat{y}(t_j) = p_1 t_j^3 + p_2 t_j^2 + p_3 t_j + p_4$ .

The adjusted ageing model parameters  $\{p_{H,1}, p_{H,2}, p_{H,3}, p_{H,4}\}$  are used for future battery RUL prediction.

#### 4. SYSTEM IMPLEMENTATION

The proposed CVHM architecture has been implemented in a three-vehicle test fleet for the battery RUL prognosis application. To reduce the development cycle and cost, the test fleet is constructed in the way that one host vehicle implements the full CVHM architecture, and two peer vehicles implement only the V2X module. Each of the two peer vehicles maintain a database of battery D&P data from multiple batteries, which simulates the situation where data from multiple peer vehicles can be transferred to the host vehicle for D&P algorithm adaptation.

For the host vehicle prototype implementation, there are three major hardware components as shown in Fig. 4. The first one is a dSpace<sup>®</sup> MicroAutoBox (MAB) that has direct connection with the sensors on the battery. It employs the functions of data acquisition, signal pre-processing, and fault signature generation. During each vehicle cranking process, the MAB generates multiple battery-status related parameters, including battery temperature, SOC, cranking resistance, minimum cranking voltage, cranking powering, delta V, voltage residual.

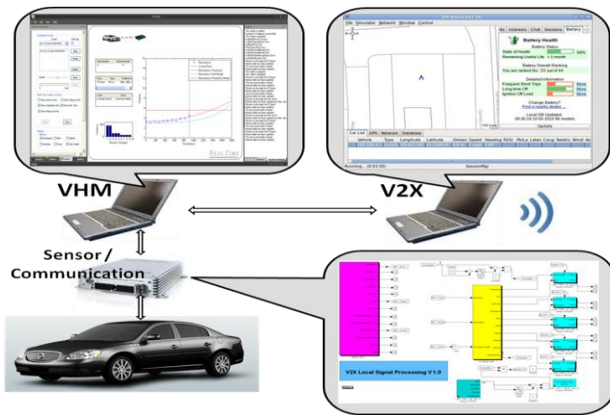


Fig. 4: Overview of system Implementation

The second major hardware component is a VHM laptop (an HP<sup>®</sup> 8440w laptop with Microsoft<sup>®</sup> Windows XP) that connects with the MAB through a Vector<sup>®</sup> CANCaseXL. The VHM laptop implements a VHM module that runs the adaptive D&P algorithms to predict battery RUL. The implementation of VHM module involves multiple operations, including the CAN (Control Area Network) communication with MAB, the D&P algorithms, the database manipulations, the communication with the V2X laptop, and a graphical interface for development users to conduct debugging and demonstration. A C++/MATLAB mixed programming technique is used to effectively accommodate different operation needs.

The third major hardware component is a V2X communication laptop (an HP<sup>®</sup> Compaq 6910P with the OS of Linux Ubuntu 10) that communicates with the VHM laptop through TCP/IP based connection. The V2X laptop implements the V2X module that interacts with peer vehicles and infrastructure through a wireless communication to exchange data. It maintains a MySQL<sup>®</sup> database server to organize the data as well as manage the retrieval requests from the VHM module. The details on this part of implementation will be covered in a forthcoming publication. The V2X laptop also serves as the driver interface module to provide battery health information to the end user.

#### 5. EXPERIMENTAL RESULTS

The CVHM system has been validated using the JBI\_Aging\_2008 data set. In this data collection effort, 15 batteries from different suppliers were aged from fresh to the end of life through an accelerated ageing process. The battery age varies from 8 to 16 weeks. During the ageing process, weekly cranking tests were conducted on a test vehicle for each battery after it was conditioned to 100% state of charge (SOC) and the temperature of 25°C. Battery current, battery voltage, and engine RPM were collected during cranking. After data cleaning, there are totally 1710 cranking data files that have adequate data for 14 batteries (battery 9 has no data). Table 2 summarizes the battery information from this data set.

Battery ID	Indices of Battery Types	Accelerated Ageing Life (weeks)
1	I	13
2		10
3		8
4	II	16
5		16
6		16
7	III	14
8		16
9		No Data
10	IV	10
11		16
12		12
13	V	14
14		13
15		16

Table 2: Summary of JBI\_Aging\_2008 data set

### 5.1. Fault signature

Fig. 5 shows the fault signatures of the batteries in the JBI\_Aging\_2008 dataset, including minimum cranking voltage, delta V, cranking power, voltage residual, and cranking resistance. Among these fault signatures, cranking resistance appears to be better SOH indicators than others, due to its consistency and monotonic correlation with the battery age. Therefore, we selected the cranking resistance as the fault signature in the rest of the experiments.

### 5.2. Remaining useful life (RUL) prediction

In order to evaluate CVHM-based battery RUL prediction, we conducted the experiment as follows. We randomly selected a battery, battery #6, from JBI\_Ageing\_2008 dataset, and loaded the cranking data from battery #6 to the local database on the host vehicle. At each ignition on, the cranking data at different battery age was fed to the VHM module in order to simulate the battery ageing process. The data from another randomly selected battery, battery #2, was used to calibrate the initial battery ageing model as described in Section 3.2. The data from the remaining 12 batteries were loaded in the two peer vehicles in order to simulate the fact that the peer vehicle population carries different batteries, and has gone through the full battery ageing process on those batteries. The host vehicle experienced multiple encounters with the peer vehicles, during which the battery data stored in peer vehicles were transferred to the host vehicle through V2V communication. The host vehicle used the newly acquired data to update the ageing model and the battery RUL prediction.

Fig. 6 illustrates the battery RUL prediction results during one particular ignition cycle. At this particular ignition cycle, the host vehicle battery has been in service for 540 days, assuming each week of accelerated ageing corresponding to about 90 days of real-world driving. The cranking resistance has increased from the initial value, but is still significantly lower than the end of life threshold indicated by the black horizontal line in Fig. 6. The initially calibrated ageing model, as shown by the blue line in Fig. 6, predicts the RUL is about 250 days, since the cranking resistance is predicted to pass the threshold in about 250 days. This prediction is very different from the actual cranking resistance data that are shown by the black cycles. At the same time, the host vehicle has access to the data from peer vehicles' batteries, of which the data from nearest neighbors are shown by the green crosses in Fig. 6. Following the model adjustment procedure presented in Section 3.2, an updated battery ageing model is obtained, and shown by the green line in Fig. 6. The updated ageing model traces the actual cranking resistance very well, and provides a fairly accurate RUL prediction. Table 3 presents more detailed RUL prediction results. After the first encounter between host vehicle and a peer vehicle, the peer vehicle transferred the data of four batteries to the host

vehicle. The updated ageing model had an RUL prediction error of 339 days. As more battery data was transferred, the RUL prediction error of the updated ageing model continuously reduced. So was the standard deviation of the prediction error, which suggests that the prediction is increasingly reliable.

## 6. DISCUSSIONS AND FUTURE WORKS

### 6.1. Preliminary penetration analysis

One of the key factors to the success of the CVHM framework is the access to peer vehicles' data. This is especially challenging in the early phase of CVHM deployment when the penetration of the CVHM system is low. We will try to answer the question that how many peer vehicles are needed to achieve specific RUL prediction performance.

The performance of RUL prediction can be measured by the accuracy and the precision (Vachtsevanos, Lewis, Roemer, Hess, & Wu, 2006). The accuracy specifies the difference between predicted value and the actual value. The precision specifies the spread of the predicted values. Many different metrics have been proposed (Saxena, et al., 2008). In this paper, we define the RUL prediction accuracy as  $E(t_{pf} - t_{af,h})$ , which is the expectation of the error between the predicted battery RUL,  $t_{pf}$ , and the actual RUL of the host vehicle battery,  $t_{af,h}$ . And the precision is defined as  $\sigma(t_{pf} - t_{af,h})$ , which is the standard deviation of the error.

With the adaptive prognostics proposed in Section 3.2,  $t_{pf}$  is obtained by the sample mean of the battery RUL from selected peer vehicles. That is  $t_{pf} = \frac{1}{n} \sum_{i=1}^n t_{af,i}$ , where  $t_{af,i}$  is the actual battery RUL for selected peer vehicle  $i$ . Assuming the batteries of the host vehicle and the selected peer vehicles have the same ageing behavior,  $t_{af,i}$  and  $t_{af,h}$  follow the independent and identical distribution (i.i.d.) with the expectation  $E_0$  and the standard deviation  $\sigma_0$ . According to (Spiegel, Schiller, & Srinivasan, 2009), we have,

$$E(t_{pf} - t_{af,h}) = E\left(\frac{1}{n} \sum_{i=1}^n t_{af,i} - t_{af,h}\right) = \frac{1}{n} \sum_{i=1}^n E(t_{af,i}) - E(t_{af,h}) = \frac{1}{n} \sum_{i=1}^n E_0 - E_0 = 0,$$

and

$$\sigma(t_{pf} - t_{af,h}) = \sigma\left(\frac{1}{n} \sum_{i=1}^n t_{af,i} - t_{af,h}\right) = \sqrt{\text{Var}\left(\frac{1}{n} \sum_{i=1}^n t_{af,i}\right) + \text{Var}(t_{af,h})} = \frac{\sqrt{n+1}}{n} \sigma_0.$$

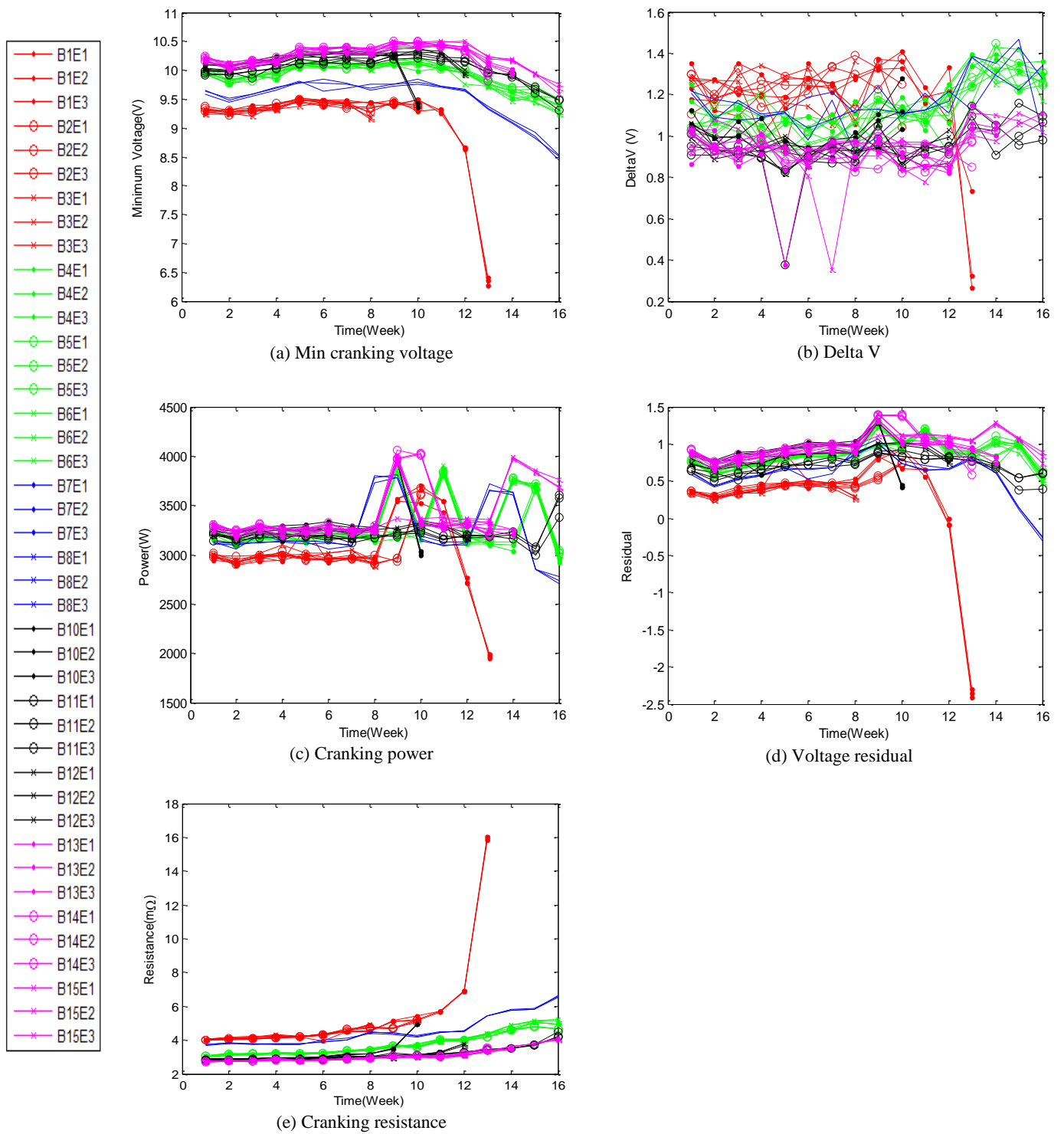


Fig. 5: Fault signatures at 100% SOC and 25C for batteries in JBI\_Ageing\_2008 data set. Same battery types share the same color in the figure.

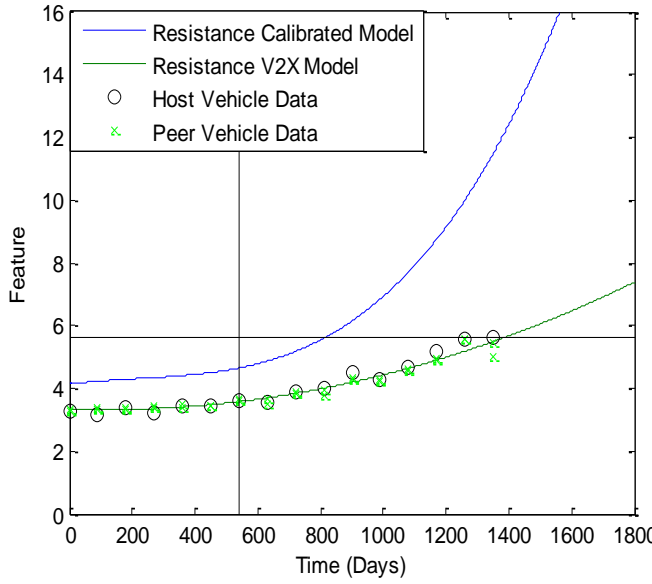


Fig. 6: Comparison of battery RUL prediction with pre-calibrated model and adaptive model

Number of peer vehicles' batteries	4	8	12	13
Actual RUL (Days) – Ground truth	964	964	964	964
Predicted RUL (Days)	625	1143	1040	992
Error (Days)	-339	+179	+76	-28
Sample Standard Deviation (Days)	460	220	139	71

Table 3: The relationship between the prediction power and the number of batteries whose data are transferred from peer vehicles. The results are based on the JBI\_Aging\_2008 dataset. Note that the prediction error and standard deviation are relatively high, due to the fact that the batteries in the JBI\_Aging\_2008 dataset represent 5 totally different battery types. In real applications, data from same battery type is available, and the prediction performance will be better.

In other words, under the i.i.d. assumption, the RUL estimation will have zero expected error, which is very desirable. And the error spread of the CVHM-based prediction is reduced by a factor of  $\frac{\sqrt{n+1}}{n}$  from the single vehicle battery RUL prediction, which shows why the CVHM framework enhances the prediction performance.

If we further assume the battery RUL distribution is normal, the prediction error will be within the error bound  $[-\Delta, +\Delta]$ , where  $\Delta = \Phi^{-1}(1 - \frac{\alpha}{2})\sigma(t_{pf} - t_{af,h})$  at the confidence level  $\alpha$ , and  $\Phi$  is the cumulative normal distribution function. To achieve a specific RUL error bound  $\Delta_0$ , we need to have  $\Delta \leq \Delta_0$ , which yields that the minimum number of selected peer vehicles is  $[\Phi^{-1}(1 - \frac{\alpha}{2})\sigma_0/\Delta_0]^2$ . To simplify the calculation, we approximate  $\frac{\sqrt{n+1}}{n}$  with  $\frac{1}{\sqrt{n}}$ .

Here is a hypothetic example. Suppose the standard deviation of battery life for the whole population of a certain battery type is 35 days, i.e.,  $\sigma_0 = 35$ . This means that the actual life of most of this battery population (95% if a normal distribution is assumed) fails within a range of +/-70 days around the average battery life, which is a fairly wide spread. To achieve the RUL prediction error bound of +/- 7 days ( $\Delta_0 = 7$ ) with 95% confidence ( $\alpha = 95\%$ ), the least number of selected peer vehicles is 100, according to the analysis above. As a comparison, Table 4 shows the number of potential peer vehicles with similar ageing behavior under different CVHM penetration rates.

Vehicle population (within a Metro area)	Penetration of CVHM systems	Percentage of selected peer vehicles (with similar ageing behavior)	No. of selected peer vehicles
200 K	0.5%	5%	50
200 K	1%	5%	100
200 K	2%	5%	200

Table 4: Number of potential peer vehicles with different CVHM penetration rate

### 6.2. Sophisticated ageing model

In a CVHM system, the ageing model is important for system efficiency and accuracy. As an initial attempt, we explored a few static parametric models in this project, such as 3rd order polynomial model, exponential model, and support vector machine, and obtained satisfactory experimental results. In the future, we plan to further investigate other physics-based models and pure data-driven models. We are currently in the process of developing a physics-based Lead-Acid battery ageing model. It models various aspect of battery behavior including electrical, thermal, and ageing. It covers major battery failure modes such as corrosion, sulfation, and water loss. It also models a very particular phenomenon in flooded Lead Acid batteries, called acid stratification, which is not a battery failure mode itself, but accelerates the battery ageing.

System ageing depends on not only the intrinsic system property but also on its usage. Therefore, system RUL prediction has to be able to capture and predict time-

dependent usage patterns, which is usually very difficult to do with physics-based models. On the data-driven side, time-series models (Brockwell & Davis, 1991) (Rasmussen & Williams, 2006) are among other candidate methods we are considering to capture more complex trends in field data.

#### ACKNOWLEDGEMENT

The authors would like to thank the V2X communication team members, Fan Bai and Bo Yu, for developing the V2X communication solution in this project.

#### REFERENCES

- Bai, F., Grimm, D., Talty, T., & Saraydar, C. (2011). Gossip Networks: The Enabler for Sparsely Populated VANETs. *SAE-2011-01-0046, SAE World Congress*. Detroit, MI.
- Brockwell, P. J., & Davis, R. A. (1991). *Time Series: Theory and Methods* (Second ed.). New York: Springer-Verlag Inc.
- Byttner, S., Rögnvaldsson, T., Svensson, M., Bitar, G., & Chominsky, W. (2009). Networked vehicles for automated fault detection. *Proceedings of IEEE International Symposium on Circuits and Systems*. Taipei, Taiwan.
- Carr, B. J. (2005). Practical application of remote diagnostics. *SAE World Congress*. Detroit, Michigan.
- Edwin, G. R., Chiang, Y.-M., Carter, W. C., Limthongkul, P., & Bishop, C. M. (2005). Microstructural Modeling and Design of Rechargeable Lithium-Ion Batteries. *Journal of The Electrochemical Society*, 152(1), A255-A263.
- Kuschel, J. O. (2004). Presenting a conceptual framework for remote vehicle diagnostics. *IRIS 27*.
- Millstein, S. (2002). vRM (vehicle Relationship Management). *Convergence: Transportation Electronics Conference*. Detroit, Michigan.
- Rasmussen, C. E., & Williams, C. K. (2006). *Gaussian Process for Machine Learning*. Cambridge, Massachusetts: The MIT Press.
- Saha, B. G., Poll, S., & Christophersen, J. (2009, February). Prognostics Methods for Battery Health Monitoring Using a Bayesian Framework. *IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT*, 58(2), 291-296.
- Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., et al. (2008). Metrics for Evaluating Performance of Prognostics Techniques. *Proceedings of 1st International Conference on Prognostics and Health Management (PHM08)*. Denver, CO.
- Schiffer, J., Sauer, D. U., Bindner, H., Cronin, T., Lundsager, P., & Kaiser, R. (2007). Model prediction for ranking lead-acid batteries according to expected lifetime in renewable energy systems and autonomous power-supply systems. *Journal of Power Sources*, 168, 66-78.
- Shin, K., & Salman, M. (2010). Evidence Theory Based Automotive Battery Health Monitoring. *SAE Int. J. Passeng. Cars - Electron. Electr. Syst.*, pp. 10-16.
- Spiegel, M. R., Schiller, J. J., & Srinivasan, R. A. (2009). *Probability and Statistics*. New York: McGraw-Hill Companies Inc. .
- Vachtsevanos, G., Lewis, F. L., Roemer, M., Hess, A., & Wu, B. (2006). *Intelligent fault diagnosis and prognosis for engineering systems*. Hoboken, New Jersey: John Wiley & Sons, Inc.
- Vapnik, V. N. (1998). *Statistical Learning Theory*. New York, US: John Wiley & Sons, Inc.
- You, S., Krage, m., & Jalics, L. (2005). Overview of remote diagnosis and maintenance for automotive systems. *SAE World Congress*. Detroit, MI.
- Zhang, X., Grube, R., Shin, K., & Salman, M. (2008). Automotive Battery State-of-Health Monitoring: a Battery Cranking Voltage based Approach. *Proceedings of the 2008 Integrated System Health Management Conference*. Covington, KY.
- Zhang, X., Grube, R., Shin, K., & Salman, M. (2009). A parity-relation based approach to starting, lighting and ignition battery state-of-health monitoring: algorithm development. *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes (SAFEPROCESS)*. Barcelona, Spain.
- Zhang, Y., Grantt, G. W., Rychlinski, M. J., Edwards, R. M., Correia, J. J., & Wolf, C. E. (2009). Connected vehicle diagnostics and prognostics, concept and initial practice. *IEEE Transactions on Reliability*, 58, 286-294.
- Zoia, D. E. (2006). OnStar e-mail service hits million mark. [Online] [wardauto.com,http://wardsauto.com/ar/onstar\\_email\\_million/](http://wardsauto.com/http://wardsauto.com/ar/onstar_email_million/).

#### BIOGRAPHIES

**Yilu Zhang** received his B.S., and M.S. degrees in electrical engineering from Zhejiang University, China, in 1994, and 1997, respectively; and his Ph.D. degree in computer science from Michigan State University, East Lansing, Michigan in 2002. He joined General Motors Global R&D center at Warren, Michigan in 2002, and currently holds a position of Staff Researcher. His research interests include statistical pattern recognition, machine learning, signal processing, and their applications, including human machine interactions, and integrated vehicle health management. Dr. Zhang served as an Associate Editor of International Journal of Humanoid Robotics from 2003 to 2007, the Publication Chair for IEEE 8th International



Conference on Development and Learning 2009, and the Chair of Battery Management System Workshop in conjunction with PHM Society Annual Conference 2011. In 2008, Dr. Zhang received the "Boss" Kettering Award - the highest technology award in GM - for his major contribution to remote vehicle diagnostics technology.

**Xinyu Du** received the B.Sc. and M.Sc. degrees in Automation from Tsinghua University, Beijing, China, in 2001 and 2004, respectively, and the Ph.D. degree in electrical engineering from Wayne State University, MI, USA, in 2012. He has been working in Electrical and Controls Integration Lab, General Motors Global R&D Center, Warren, MI since 2010, and currently holds the position of senior software engineer with Patni Americas Inc. His research interests include type-2 fuzzy control, fuzzy discrete event system and hybrid system, MEMS device fabrication and FEM simulation, data mining over vehicular network and vehicle battery health management. He is currently a member of IEEE CIS GOLD sub-committee and was a member of IEEE fuzzy system competitions technical committee in 2009 and 2011. He served as the Lead Guest Editor of Advances in Fuzzy Systems. He received the best student paper finalist award from the 24th NAFIPs annual conference in 2005 and the Ralph H. Kummeler Award for distinguished achievement in graduate student research from Wayne State University in 2010.

**Mutasim Salman** is a Lab. Group manager and a Technical Fellow in the Electrical, Controls and Integration Lab. of GM Research and Development Center. He has the responsibility of development and validation of algorithms for state of health monitoring, diagnosis, prognosis and fault tolerant control of vehicle critical systems. He pioneered the work on integrated chassis control in the late eighties that led to the production of GM "Industry First" Stabilitrak1 and then to Stabilitrak3. He had an extensive experience in hybrid vehicle, modeling, control and energy management strategies. He has several GM awards that includes 3 GM prestigious Boss Kettering, 3 McCuen, and 2 President and Chairman Awards. Mutasim received his bachelor's degree in Electrical Engineering from University of Texas at Austin; M.S. and PhD in Electrical Engineering with specialization in Systems and control from University of Illinois at Urbana- Champaign. He also has an Executive MBA. He is IEEE senior member. He holds 21 patents and has coauthored more than 42 refereed technical publications and a book. He joined the GM R&D staff in 1984.

# Fault Diagnosis of Gas Turbine Engine LRUs Using the Startup Characteristics

Kyusung Kim<sup>1</sup>, Onder Uluyol<sup>2</sup>, Girija Parthasarathy<sup>3</sup>, and Dinkar Mylaraswamy<sup>4</sup>

*Honeywell Aerospace, Golden Valley, MN, 55422, USA*

<sup>1</sup>*Kyusung.kim@honeywell.com*

<sup>2</sup>*Onder.uluyol@honeywell.com*

<sup>3</sup>*Girija.parthasarathy@honeywell.com*

<sup>4</sup>*Dinkar.mylaraswamy@honeywell.com*

## ABSTRACT

This paper introduces a feature-extraction method to characterize gas turbine engine dynamics. The extracted features are used to develop a fault diagnosis and prognosis method for startup related sub-systems in gas turbine engines - the starter system, the ignition system and the fuel delivery system.

The startup of a gas turbine engine from ignition to idle speed is very critical not only for achieving a fast and efficient startup without incurring stall, but also for health monitoring of many subsystems involved. During startup, an engine goes through a number of phases during which various components become dominant. The proposed approach physically monitors the relevant phases of a startup by detecting distinct changes in engine behavior as it manifests in such critical variables as the core speed and the gas temperature. The startup process includes several known milestones, such as starter-on, light-off, peak gas temperature, and idle. As each of these is achieved, different engine components come into play and the dynamic response of the engine changes. Monitoring engine speed and exhaust gas temperature and their derivatives provides valuable insights into engine behavior.

The approach of the fault diagnosis system is as follows. The engine startup profiles of the core speed (N2) and the gas temperature are obtained and processed into a compact data set by identifying critical-to-characterization instances. The principal component analysis is applied to a number of parameters, and the fault is detected and mapped into three engine component failures which are the starter system failure, the ignition system failure, and the fuel delivery system failure.

---

Kyusung Kim et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

In this work, actual engine test data was used to develop and validate the system, and the results are shown for the test on engines that experienced startup related system failures. The developed fault diagnosis system detected the failure successfully in all three component failures.

## 1. INTRODUCTION

The gas turbine engine is one of the most vital aviation components. While the heart of this propulsion system is the gas producer that converts fuel into mechanical energy, several LRUs (Line Replaceable Units) contribute to the overall health and remaining useful life of the propulsion function. Although some LRUs may not be considered to be engine OEM parts, they nevertheless contribute to the prognostic health of the propulsion system. Consequently, any accurate estimate of propulsion remaining useful life calculation from a CBM (Condition Based Maintenance) perspective must account for all such LRUs (Parthasarathy et al., 2011).

Current LRU fault detection is achieved using built-in-tests (BIT). Unfortunately, these BIT implement simple threshold checks (i.e., hard faults) without taking a systems perspective of the propulsion system. Significant maintenance effort is expended to troubleshoot and isolate in-range (i.e., soft) faults. As a result, when the component fails, it is too late and manifests as an engine shutdown or loss of power control at the propulsion function level. A failed LRU will drive maintenance costs and operational interrupts in two ways: 1) an LRU failure is misdiagnosed as an engine problem and the engine is removed, and 2) the engine must be removed to access certain LRUs.

Most engine fault diagnosis/prognosis is performed using measurement data from engines that are in steady-state conditions. The steady-state data is used for several reasons; most notably, system transients can confuse fault

diagnosis/prognosis methods, giving incorrect results. A more robust approach to developing fault diagnosis/prognosis methods that explicitly accounts for transient data is required (Surender et al., 2005). Furthermore, most turbine engine fault diagnosis/prognosis methods are developed with engine performance models that have been validated only under steady-state conditions or with actual engine data at steady-state conditions. Engine models that accurately represent the system in transient conditions are difficult to develop. Nevertheless, developing fault diagnosis/prognosis methods designed to operate on system data during transient as well as steady-state operation has several important advantages; 1) Certain system faults have a distinct signature during system transient conditions that would not normally be discernible during steady-state conditions. 2) The effect of feedback control action is less dominant during transient conditions than during steady-state conditions. Because feedback control suppresses the effect of sensor and system faults, faults are more evident during transient conditions. 3) Certain engine component incipient faults are manifest only during transient conditions such as startup and shutdown (e.g., starter and igniter system faults) (Uluyol et al., 2006).

Thus, for a vehicle health monitoring system that is comprehensive in its scope, and timely and accurate in its diagnosis, high fidelity engine models and a large amount of high-speed data both in steady-state as well as in transients are needed. However, limited computational resources available on-board, and the limited bandwidth capacity and the high cost of real-time data transmission place serious barriers to fulfilling that need (Kim et al., 2005).

The approach presented in this paper seeks to overcome these barriers by separating the initial feature extraction stage of diagnostics algorithms from the modeling and trending stages. The first stage, which includes the detection of time instances that are critical to diagnosis and control, is performed on board, while the latter stages are performed on a ground station. This paper presents an approach that permits a much greater insight into the engine health than is possible with a couple of snapshots at takeoff and cruise, while keeping the data size much smaller than that of the complete high speed data. The fault diagnosis method for startup related components in turbine engines is developed using continuous time series data. The engine startup data set recorded with high speed sampling rate is analyzed to discover the best conditions for detecting the target component failure. Engine startup profiles of the core speed (N2) and the measured gas temperature (MGT) are then transformed into a compact set by a series of data processing steps. The processing is based on statistical characteristics analysis, and principal component analysis (PCA). The fault is detected and mapped into three engine component failures which are the starter system failure, the ignition system failure, and the fuel delivery system failure.

The remainder of this paper is organized as follows: In Section 2, the functionality of startup related components is described. In Section 3, the engine startup process is described. Section 4 describes the engine startup profile and the critical-to-characterization (CTC) instances. Section 5 presents the developed fault diagnosis/prognosis system with the steps of the algorithm logic. Section 6 shows the results of the developed diagnosis system. The data set used to develop the system is the actual engine data recorded in the test cell. Finally, Section 7 presents the summary and conclusions.

## 2. ENGINE STARTUP SYSTEM AND COMPONENTS

Gas turbine engines are complicated pieces of machinery, so fault diagnosis of these machines is enhanced by a detailed understanding of the equipment. The engine of interest in this work is a turbo-shaft engine.

The startup system consists of the ignition system, starter motor and the fuel system. The engine ignition system requires an external source of power. The ignition system provides igniters and exciter output circuits, and each igniter and circuit releases sufficient energy for all ground and air starting requirements. The ignition channels are powered by 28 V dc when the starter is energized. It supplies energy for spark at the igniter plugs. Adequate energy is supplied by the ignition system throughout its input voltage operating range to obtain successful engine starts.

For starting on the ground, the automatic start sequence is enabled by the pilot placing the power level angle (PLA) in the IDLE position, and holding the momentary start switch. Given sufficient battery power, the gas producer will begin to accelerate with the ignition enabled. As the gas producer accelerates, the ECU commands introduction of fuel to the engine. Light-off is declared if a rise in MGT is detected based on the MGT rate of change or MGT increase since introduction of fuel, or if Ng is greater than some pre-defined level. There is a start abort if any of the following are detected: excessive MGT, hung start, no light-off, or no Ng rotation.

The fuel system on the turbine engine provides fuel to the engine for proper combustion under all circumstances. The fuel system comprises the FMU (Fuel Metering Unit), fuel pump, high-pressure filter, min-flow valve, and fuel manifold assembly. Fuel metering functions are provided by the FMU. In the installed condition, there is no accessible means for adjusting the FMU. The high-pressure fuel filter is installed upstream of the FMU. It is designed and constructed to minimize the release of contaminants upstream of the metering valve and the fuel manifold.

## 3. ENGINE STARTUP PROCESS

The key features of the transient startup process are captured and made available as enriched inputs to LRU fault isolation

algorithm. A typical engine speed plot during startup is shown in Figure 1.

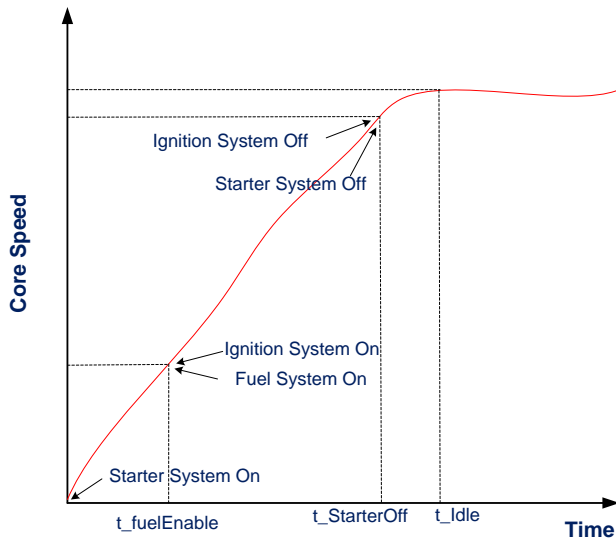


Figure 1. Typical Engine Start Profile

Startup includes several milestones such as starter on, light-off, peak temperature, and idle. As each of these states is achieved, different components come into play, and the dynamic response of the engine changes. The gas producer core speed (NGG) and measured gas temperature (MGT) are two of the most informative measurements for detecting or verifying whether these states are achieved. Monitoring their derivatives also provides valuable insights for engine behavior.

In general, the automatic start is performed by latching a combined starter/igniter relay and starter/ignition systems are on. The maximum engine speed gradient occurs when the engine speed has its highest rate of change during startup—usually a few seconds after the starter is switched on. According to the engine control logic, the control system shall deliver regulated fuel flow at the fuel metering unit delivery port when core speed exceeds some pre-defined level of full speed. Light-off occurs when ignition successfully completes and the combustor is able to sustain combustion. The maximum temperature gradient that corresponds to the highest rate of change in MGT follows the light-off several seconds behind during startup. The power section then begins to provide rotational energy to the system. Peak temperature occurs when the engine reaches its highest temperature during startup. At some pre-defined level of engine speed, the starter system and the ignition system are disabled through the ignition/starter relay driver. Finally, the ground idle occurs when the engine reaches its idle speed.

Figure 2 shows the time periods when the function of startup related LRUs is active during the startup process and they will be used to select the proper startup features. For example, the starter system anomaly can be detected and differentiated from the anomalies due to ignition and fuel systems, if the features are selected at ‘Fuel Enable’. For the ignition system anomaly detection, the features at ‘light-off’ and ‘peak MGT dot’ are more proper, whereas the features at ‘peak MGT’ and ‘idle’ are more proper for the fuel system anomaly detection.

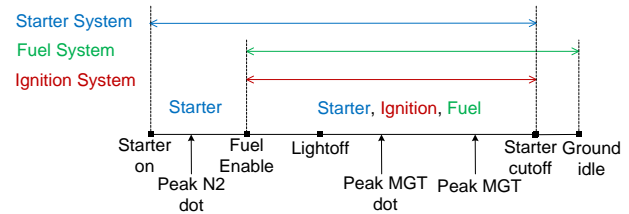


Figure 2. Startup Partition According to LRUs Functional Activity

By storing and analyzing engine sensor data taken during these key conditions of engine startup, the system and method is able to accurately characterize the performance of the engine during startup. This information is the basis for LRU prognostics to determine when faults in the start transient regime are occurring or likely to occur. Furthermore, the approach provides this ability to characterize the engine startup performance using only the sensor data taken during the key conditions.

#### 4. STARTUP PROFILE

In this section, we discuss how the startup can be characterized using a small number of data points rather than equally spaced time series data.

The startup of a gas turbine engine from ignition to idle speed is very critical not only for achieving a fast and efficient startup without incurring stall, but also for health monitoring of many subsystems involved. The state of the art in monitoring engine startup is that engine parameters are sampled at regular frequencies and compared against fixed thresholds on these parameters. Often, the thresholds are set arbitrarily---monitoring parameters at 10%, 20%, 30% engine speed, etc. Sometimes the thresholds are set by experts or based on design specifications. In either case, startup monitoring does not capture the changes in engine response accurately and in a timely manner, since the changes in engine response manifest as an engine achieves certain startup phases, and not necessarily as some arbitrary thresholds are reached.

During startup, an engine goes through a number of phases in which various components become dominant. In the

absence of very detailed and costly engine models, the phases can be determined by monitoring the dynamic response of the engine. Our approach is the combination of monitoring physically relevant phases of a startup and monitoring the engine control schedule. The physically relevant phases can be obtained by detecting distinct changes in engine behavior as it manifests in such critical variables as core speed and exhaust gas temperature. This approach is superior to monitoring predetermined thresholds since the time the data should be captured is determined on the fly (hence, it varies from one flight to another). The engine control schedule can be obtained by the engine control logic. Some of the control logic provides additional insights of the engine operating conditions. For example, the engine control logic schedules the time when the fuel shall start to provide to the engine. This time of instance is very important since the performance of ignition system and fuel delivery system can be evaluated from this point of time. Unless you don't have a very accurate fuel flow measurement sensor (in fact, this is not the case almost always), this information cannot be obtained.

As described in the previous section, startup includes several known milestones, such as light off, peak MGT, and idle. As each of these is achieved, different components come into play and the dynamic response of the engine changes. The engine core speed (N2) and MGT are two of the most informative sensors. Monitoring their derivatives also provides, as we shall see below, valuable insights for engine behavior. Based on the derivatives of N2 and MGT, we can identify very precisely the time instances of light off, peak N2 dot, and peak MGT dot (Uluyol et al., 2005).

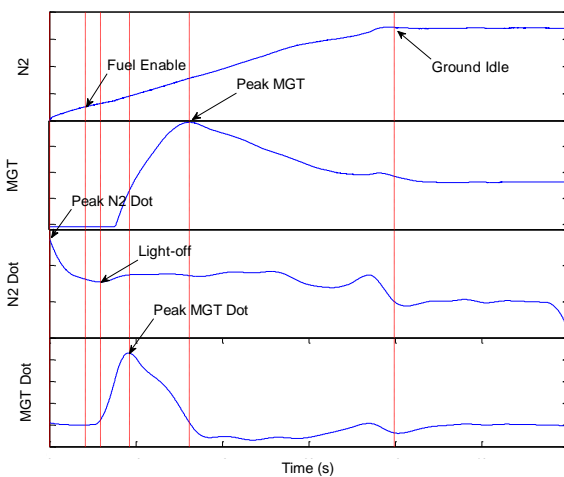


Figure 3. Features Extracted from the Startup Algorithm

When applied to actual startup data, the CTC instances cut across N2 and MGT profiles as well as other engine variables of interest depending on the failure type. An example of indication of CTC instances over the N2 and

MGT startup profile plotted with actual data is shown in Figure 3. This figure helps the chronological understanding of the CTC instances occurrence. Notice that the CTC instances are nicely, but not equally, distributed between the start and idle speed. The distance between the lines changes as the startup profile changes. However, the simple patterns that each variable forms retain their shape, thereby allowing an automatic and consistent feature extraction.

### 5. FAULT DIAGNOSIS SYSTEM

The fault diagnosis system proposed in this work is presented in Figure 4. N2 and MGT, which are the two most important engine performance parameters, are processed at each flight to detect any startup related LRU anomalies.

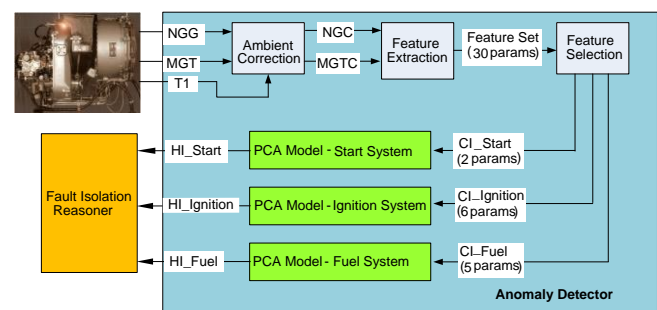


Figure 4. Schematic description of the algorithm

As emphasized earlier, the one of the ideas of developing fault prediction method in this work is based on the data reduction. The N2 and MGT startup profiles are continuous time series data. Considering the engine startup transient time, which typically takes 40-50 seconds, they consist of a large number of samples; the number depends on the sampling rate (for example, 2000-2500 samples with 50 HZ sampling rate, which was used in this work). Rather than analyzing whole continuous profiles having a large number of samples, the processing of a few points that represent the whole profile is much more efficient. In fact, there are two perspectives of data dimensionality reduction: technical perspective and practical perspective. In many problems, reducing the number of input variables can sometimes lead to improved performance for a given data set, even though some information is being discarded. The fixed quantity of data is better able to specify the mapping in the lower dimension space and this more than compensates for the loss of information (Bishop 1995). In the practical perspective, there are several more advantages to condensing the data: 1) It minimizes the cost and space for data collection and storage; 2) Computationally faster data processing makes for timely prognostic decisions; and 3) Not all of the engine data collection system can record continuous high speed data. Thus, a fault diagnosis system based on condensing the data set requires minimal modification of the existing data acquisition system. The

data reduction processing, which reduces the data size from the continuous time series data consisting as many as 2500 samples into 2 - 6 samples, is accomplished in two steps; from the second step to the third step as described below.

The first step in the algorithm covers data extraction from the engine and standardization of the extracted data. Since the startup profiles of N2 and MGT vary depending on the ambient conditions, the abnormal engine startup can result not only from the malfunctioning engine but also from ambient conditions. Correcting engine parameters against the standard condition is necessary to decouple the effect from the varying ambient conditions. The correction of N2 and MGT is done using ambient temperature (T1).

In the second step, we obtain the snapshot data at points that best represent the characteristics of a continuous N2 and MGT startup profile. The six CTC conditions, which are Peak N2 dot, Fuel Enable, Light Off, Peak MGT dot, Peak MGT, and Idle, are selected to represent the startup profile. The snapshot data set of five parameters---Time, N2, MGT, N2 dot, and MGT dot---are obtained at the six CTC conditions, resulting in 30 parameters per startup.

As discussed in Section 3, the time periods when the function of startup related LRUs is active during the startup process are different depending on LRUs, the startup features shall be selected at different CTC instances for each LRU. The selection of proper features are done in the third step, and the resulting output from this step is the down-selected startup features, which is represented by CI (condition indicator) in Figure 4. For the starter system anomaly detection, two CIs are selected at Fuel Enable condition. For the ignition system anomaly detection, six CIs are selected at Light-off and Peak MGT dot condition. For the fuel delivery system anomaly detection, five CIs are selected at Peak MGT condition.

The fourth step is to detect any anomalies related to starter system, ignition system, and fuel delivery system. The anomaly detection is done based on Principal Components Analysis. The output of this step is the HI (health indicator) indicating existence of any anomalies in the three LRUs. The PCA model establishes correlations between the features. The anomaly detection involves measuring the multivariate distance away from the center of the correlation observed from the training set. If the distance exceeds a given threshold, then an anomaly is flagged.

The anomalies detected in the previous steps indicate faults in the startup process that encompass several components, including the engine. The last step is to isolate the possible root cause of anomalies further to subsystems or components, and to distinguish engine faults that manifest symptoms during the startup process. As shown in Figure 5, the fault isolation reasoner is composed of starter system anomaly reasoner, ignition system anomaly reasoner, and fuel delivery system anomaly reasoner. It also includes the

battery fault isolation logic. The inputs to the fault isolation reasoner are the HIs generated from the anomaly detector, additional measurements of oil temperature and fuel temperature, and the output from a different algorithm detecting fuel system fault. The outputs of the fault isolation reasoner are HIs related to starter motor, battery, igniter, fuel delivery, and engine. The logic of reasoners to isolate the root cause of each LRUs is presented in Section 6.

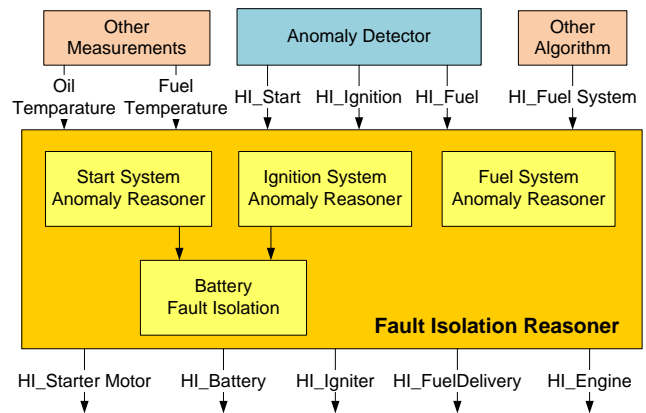


Figure 5. Fault Isolation Reasoner

## 6. FAULT DIAGNOSIS RESULT

The proposed fault diagnosis system was developed and tested with actual engine data collected from a test cell. The tests were done at the various conditions such as different altitude, Mach no., ambient temperature, oil/fuel temperatures. The startup test includes the initial static start (both cold and hot starts), aborted take-off ground start, and in-flight restart. The sampling rate of this data set is 50 Hz. The extracted parameters are N2, MGT, and T1. The ambient corrected values of N2 and MGT are computed from the empirical correction models, which are functions of T1.

Keeping the size of the data needed for the fault diagnosis much smaller than that of the complete high-speed data is advantageous and much more efficient. The rationale for extracting snapshot data is that a continuous startup profile can be represented by several points without losing its characteristics. Thirty snapshot datum per startup are obtained, each having six CTC conditions: Peak N2 dot, Fuel Enable, Light Off, Peak MGT dot, Peak MGT, and Idle for the five parameters of Time, N2, MGT, N2 dot, and MGT dot where Time, N2, and MGT are measured values and N2 dot and MGT dot are computed values. Table 1 summarizes the 30 parameters showing the CTC conditions in the column level and the engine variables in the row level. In Table 1, the features selected for the starter system anomaly detection are marked in blue and they are the Time

at Fuel Enable, and N2 dot at Fuel Enable. The features selected for the ignition system anomaly detection are marked in green and they are the time interval between Light-off and Fuel Enable, N2 at Light-off, N2 at Peak MGT dot, MGT at Peak MGT dot, N2 dot at Peak MGT dot, and MGT dot at Peak MGT dot. The features selected for the fuel system anomaly detection are marked in yellow and they are time interval between Peak MGT and Light-off, N2 at Peak MGT, MGT at Peak MGT, N2 dot at Peak MGT, and MGT dot at Peak MGT. As shown in Figure 2, the starter system anomaly detection is executed at Fuel Enable condition, the ignition system anomaly detection is executed at Peak MGT dot condition, and the fuel system anomaly detection is executed at Peak MGT condition.

Conditions	Parameters				
	Time	N2	MGT	N2 dot	MGT dot
@ Peak N2dot					
@ Fuel Enable					
@ Light-off					
@ Peak MGT dot					
@ Peak MGT					
@ Idle					

Table 1. Startup Feature Selection (Blue – starter system anomaly detection, Green – igniter system anomaly detection, Yellow – fuel system anomaly detection)

The result of starter system anomaly detection is shown in Figure 6 - Figure 8. Figure 6 shows two startup features selected for the starter system anomaly detection. The upper plot shows the Time at Fuel Enable and the lower plot shows N2 dot at Fuel Enable. The x-axis represents each startup. The normal startup is marked with blue x and the abnormal startup is marked with red dot. Figure 7 shows the PCA model output for starter system anomaly detection. The red horizontal line represents the threshold for the anomaly and blue x represents normal case and red dot represents the abnormal case. From Figure 6, the features themselves are not so distinguishable between the normal and abnormal cases. But as shown in Figure 7, the PCA model generates a clear indication of the anomaly by showing the big variance from the normal correlation among the features. Figure 8 is the N2 startup profile showing differences in the case of typical normal startup (blue curve) and the starter system anomaly (red curve). The two curves are similar to each other except around the Fuel Enable instance; Time at Fuel Enable is much larger in the case of anomaly. This clearly demonstrates that the starter system anomaly shall be detected at Fuel Enable instance. The root cause of the starter system anomaly could be various. According to Figure 6 and Figure 8, the major anomaly signature is the large value of Time to Fuel Enable. The possible root causes of delayed Fuel Enable are the

starter motor fault, the battery fault, the engine drag, and the engine rub. Figure 9 shows the logic to isolate the root cause of starter system anomaly. The ambiguity set is composed of the starter motor fault, the battery fault, the engine drag, and the engine rubs. When the starter system anomaly is triggered, two additional tests are to be done to isolate the root cause. If the oil temperature is too low and below a certain limit, then the probable cause of the delayed Fuel Enable is the engine drag. If the steady state scalar is higher than a certain level, then the Fuel Enable may be delayed due to the engine rubs. If none of those two tests are true, then the probable root causes of starter system anomaly are either the battery deterioration or the starter motor deterioration. The logic to isolate the battery problem is discussed later in this section.

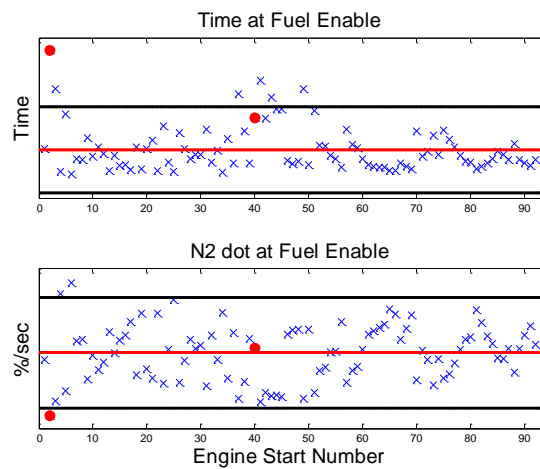


Figure 6. Startup Features for Starter System Anomaly Detection

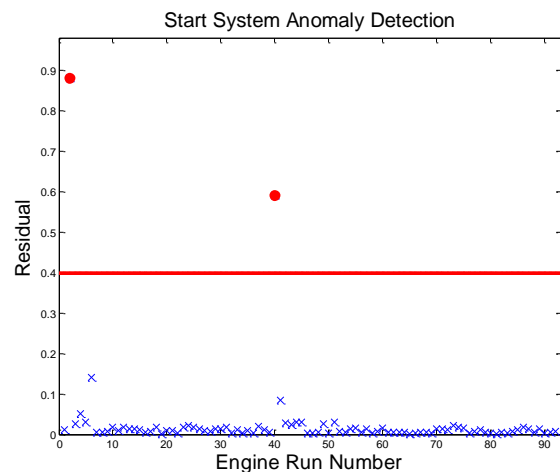


Figure 7. PCA Model Output for Starter System Anomaly Detection

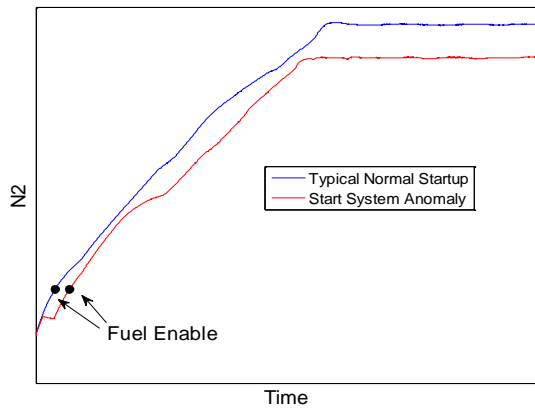


Figure 8. N2 Startup Profile comparing Normal Startup (Blue) and Starter System Anomaly (Red)

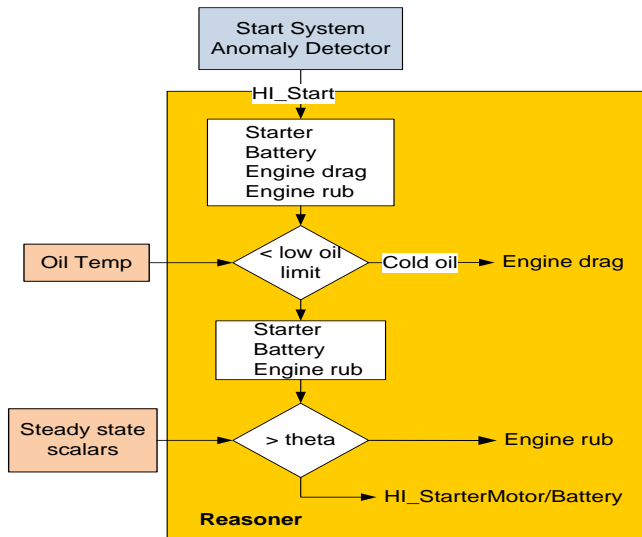


Figure 9. Starter System Anomaly Reasoner

The result of ignition system anomaly detection is shown in Figure 10 - Figure 12. Among six startup features selected for the ignition system anomaly detection, Figure 10 presents the features that show the most distinguishable signatures between the abnormal and normal startups, which are the time intervals between Light-off and Fuel Enable and N2 at Peak MGT dot. Figure 11 shows the PCA model output for ignition system anomaly detection. Figure 12 is the N2 and MGT startup profiles showing differences in the case of typical normal startup (blue curve) and the ignition system anomaly (red curve). The two curves are similar to each other in the early stage of startup but show the big difference around Peak MGT dot instance. This clearly demonstrates that the ignition system anomaly shall be detected at Peak MGT dot instance. The root cause of the ignition system anomaly could be various. According to Figure 10 and Figure 12, the major anomaly signature is the large value of the time interval between Light-off and Fuel

Enable. The possible root causes of delayed Peak MGT dot are the igniter fault, the battery fault, and the fuel/air mixture problem. The fuel/air mixture problem usually occurs at higher altitude resulting in denser fuel in the combustor chamber. Figure 13 shows the logic to isolate the root cause of ignition system anomaly. The ambiguity set is composed of the igniter fault, the battery fault, and the fuel/air mixture problem. There exists another algorithm to diagnose various fuel system faults (Mylaraswamy et al., 2011). This algorithm is based on the performance of control loops by assessing the controller dynamics. This algorithm is called the fuel scout algorithm and it can isolate the fault according to the various fuel system components such as the stepper motor, metering valve, RVDT sensor, fuel manifold sensor, and fuel nozzle. When the ignition system anomaly is triggered, the output of the fuel scout algorithm is referred to confirm if the anomaly is due to the fuel/air mixture problem. If the fuel scout algorithm does not trigger, then the probable root causes of ignition system anomaly are either the battery deterioration or the igniter deterioration. Figure 14 shows the logic to isolate the battery problem, the starter motor problem, and the igniter problem. If the starter system anomaly reasoner concludes that the anomaly is due to either starter motor or battery, and the ignition system anomaly reasoner concludes that the anomaly is due to either igniter or battery, then the probable root cause of both anomalies is the battery, because the starter motor and the igniter are powered by the same battery. If the starter system anomaly reasoner outputs HI of starter motor and battery is true, whereas the ignition system anomaly reasoner outputs HI of igniter and battery is false, then the root cause of the starter system anomaly is the starter motor deterioration. If the starter system anomaly reasoner outputs HI of starter motor and battery is false, whereas the ignition system anomaly reasoner outputs HI of igniter and battery is true, then the root cause of the ignition system anomaly is the igniter deterioration.

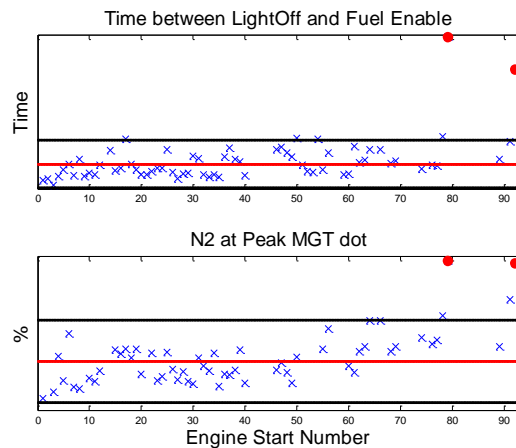


Figure 10. Subset of Startup Features for Ignition System Anomaly Detection



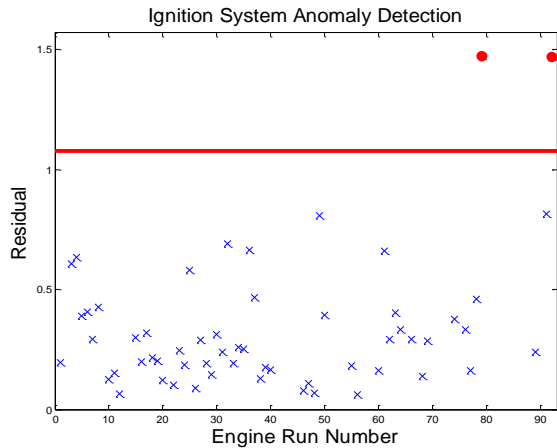


Figure 11. PCA Model Output for Ignition System Anomaly Detection

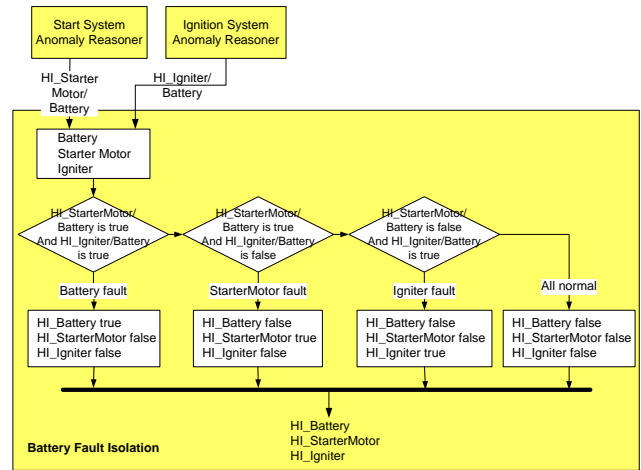


Figure 14. Battery Fault Isolation

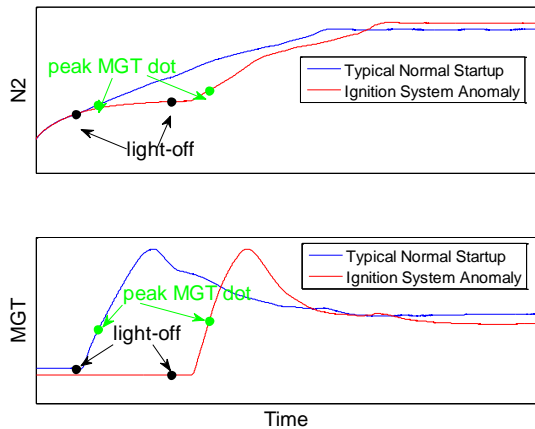


Figure 12. Startup Profiles comparing Normal Startup (Blue) and Ignition System Anomaly (Red)

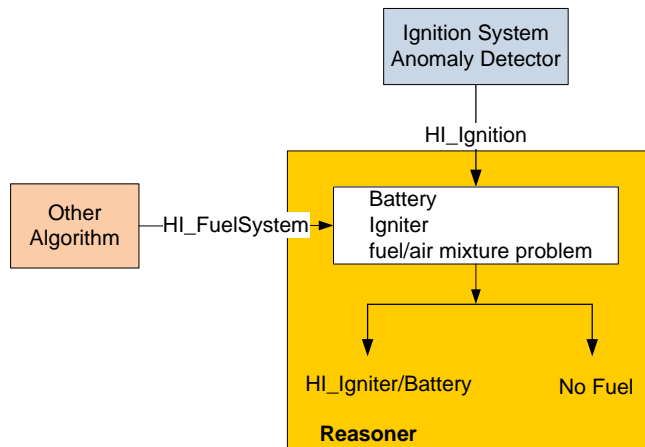


Figure 13. Ignition System Anomaly Reasoner

The result of fuel system anomaly detection is shown in Figure 15 - Figure 17. Among six startup features selected for the fuel system anomaly detection, Figure 15 presents the features that show the most distinguishable signatures between the abnormal and normal startups, which are the time intervals between peak MGT and Light-off and N2 at Peak MGT. Figure 16 shows the PCA model output for fuel system anomaly detection. Figure 17 is the N2 and MGT startup profiles showing differences in the case of typical normal startup (blue curve) and the fuel system anomaly (red curve). The two curves are similar to each other in the early stage of startup but show the big difference around Peak MGT instance. This clearly demonstrates that the fuel system anomaly shall be detected at Peak MGT instance. The root cause of the fuel system anomaly could be various. According to Figure 15 and Figure 17, the major anomaly signature is the large value of the time interval between Peak MGT and Light-off. The possible root causes of delayed Peak MGT are the fuel system deterioration, the engine deterioration, and the low fuel temperature. When the fuel is too cold and the amount of fuel delivered to the combustor chamber is not sufficient enough the secondary fuel nozzle begins to open. The size of secondary fuel nozzle is bigger so the size of the fuel droplet is bigger. Since the fuel is very cold, the colder and bigger fuel sprayed into combustor chamber results in the cool-down and the possible blown-out. Figure 18 shows the logic to isolate the root cause of fuel system anomaly. The ambiguity set is composed of the fuel system deterioration, the engine deterioration, and the low fuel temperature. If the fuel temperature is below a certain limit, then the probable cause of the delayed Peak MGT is the cold fuel. Similarly in the case of ignition system anomaly reasoner, when the fuel system anomaly is triggered, the output of the fuel loop scout algorithm is referred to confirm if the anomaly is due to the fuel system deterioration. If the fuel loop scout algorithm does not trigger, then the probable root cause of fuel system anomaly is the engine deterioration.

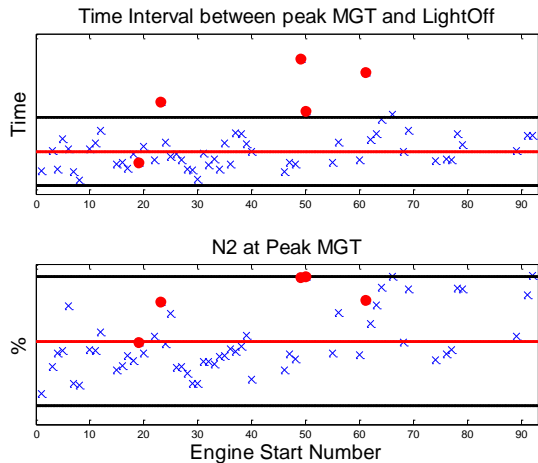


Figure 15. Subset of Startup Features for Fuel System Anomaly Detection

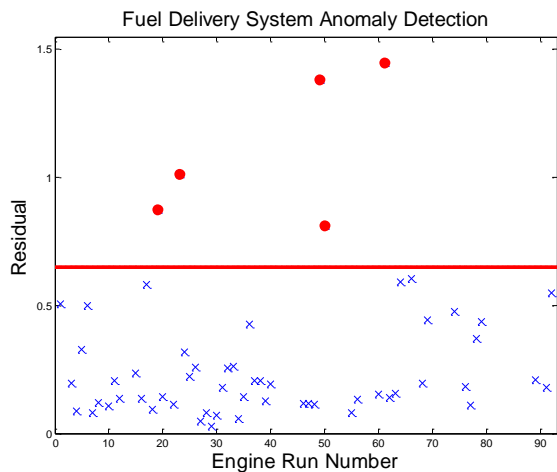


Figure 16. PCA Model Output for Fuel System Anomaly Detection

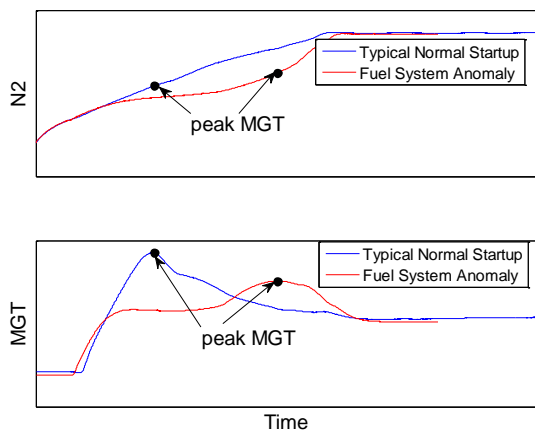


Figure 17. Startup Profiles comparing Normal Startup (Blue) and Fuel System Anomaly (Red)

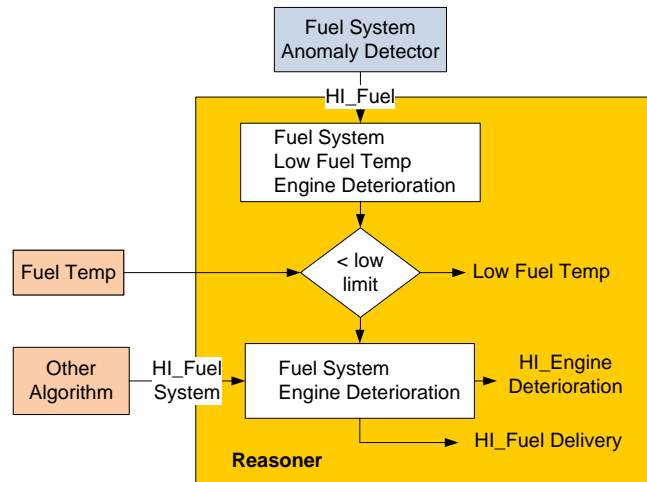


Figure 18. Fuel System Anomaly Reasoner

## 7. CONCLUSIONS

This paper introduces a method for gas turbine LRU anomaly detection during the engine startup. The approach seeks to strike a balance between the need for a large amount of high-speed data for accurately characterizing the engine condition not only at steady-state but also at transients, and the limited computational resources available on-board and the difficulties associated with storing and transmitting data. Extracting features based on actual engine dynamics and the engine control logic can be done with very minimal computational resources that are already available on most aircraft.

The time that snapshot data is taken is as important as the engine variable that is captured in the snapshot. We have shown that the conditions of Fuel Enable is more useful for the starter system anomaly detection, the Peak MGT dot condition is more useful for the ignition system anomaly detection, and the Peak MGT condition is more useful for the fuel system anomaly detection. The logic to isolate the anomaly of each LRU is also presented.

The approach has been applied on actual engine data collected in test cell. The measurements of interest are the N2 and MGT during engine startup. The developed system detects the anomalies related to the starter system, ignition system and fuel system. The main contributions of this paper are;

- This paper identifies the CTC conditions by the combination of the engine control logic and the engine dynamics.
- This paper provides the method to condense the data required to characterize engine dynamics from several hundred seconds of high speed data to about two dozen data points per startup, which has tremendous implications in engine health monitoring. Implementing

the approach on-board allows real-time data transfer and makes timely prognostics possible.

- This paper is focused on the different stages of the engine startup process, which enables to cover multiple LRUs which are critically associated to the engine startup. This paper identifies the specific time intervals when the function of a certain LRU is dominant, and the fault detection of that particular LRU is done during the identified time intervals.
- Identifies three different time intervals when the fault detection of three LRUs are done, the three LRUs include start system, ignition system and fuel system.
- Identifies three sets of fault features that exhibit the symptoms of three LRUs most effectively.
- There exist a number of causes that show the anomaly in the fault features. This is called the ambiguity set. This paper identifies the ambiguity set of each three LRUs and isolates the root cause of each anomaly.

#### REFERENCES

- Surender, V., and Ganguli, R., (2005). Adaptive myriad filter for improved gas turbine condition monitoring using transient data. *ASME J. Eng. Gas Turbines Power*, vol. 127(2), pp. 329-339.
- Uluyol, O., Kim, K., and Nwadiogbu, E., (2006). Synergistic use of soft computing technologies for the fault detection in gas turbine engines. *IEEE Trans. Syst., Man Cybern.*, vol. 36(4), pp. 476-484.
- Bishop, C. (1995). *Neural networks for pattern recognition*. Oxford: Clarendon Press
- Kim, K., Uluyol, O., and Ball, C., (2005). Fault Diagnosis and Prognosis for Fuel Supply System in Gas Turbine Engines. *Proceedings of ASME IDETC 2005*. September 24-28, Long Beach, CA.
- Uluyol, O., Kim, K., and Ball, C., (2005). On-board Characterization of Engine Dynamics for Health Monitoring and Control. *Proceedings of ASME Turbo Expo 2005*. June 6-9, Reno, NV.
- Parthasarathy, G., Mylaraswamy, D., Uluyol, O., and Kim, K., (2011). Readiness Approach for Propulsion Engine LRUs. *Proceedings of MFPT 2011*. May 10-12, Virginia Beach, VA.
- Mylaraswamy, D., Parthasarathy, G., Kim, K., and Uluyol, O., (2011). Low-cost Embedded Scouts for Engine Health Monitoring. *ISABE Conference 2011*. September 12-16, Gothenburg, Sweden.

# Correction of Data Gathered by Degraded Transducers for Damage Prognosis in Composite Structures

K. R. Mulligan<sup>1</sup>, N. Quaegebeur<sup>2</sup>, P. Masson<sup>3</sup>, and S. Létourneau<sup>4</sup>

<sup>1,2,3</sup> *Université de Sherbrooke, Sherbrooke, QC, J1K 2R1, CANADA*

*Kyle.Mulligan@USherbrooke.ca*

*Nicolas.Quaegebeur@USherbrooke.ca*

*Patrice.Masson@USherbrooke.ca*

<sup>4</sup> *NRC-IIT, National Research Council of Canada, Ottawa, ON, K1A 0R6, CANADA*

*Sylvain.Letourneau@nrc-cnrc.gc.ca*

## ABSTRACT

This paper presents an approach for the correction of data gathered for damage prognosis (DP) in composite structures. The validation setup consists of surface-bonded piezoceramic (PZT) transducers used in a Structural Health Monitoring (SHM) system with simulated bonding layer damage using Teflon masks. The modal damping around PZT mechanical resonance is used as a metric to assess and compensate for the degradation of the adhesive layer of the transducers. Modal damping is derived from electrical admittance curves using a lumped parameter model to monitor the degradation of the transducer adhesive layer. A Pitch-Catch (PC) configuration is then used to discriminate the effect of bonding degradation on actuation and sensing. It is shown that below the first mechanical resonance frequency of the PZT, degradation leads to a decrease in the amplitude of the transmitted and measured signals. Above resonance, in addition to a decrease in signal amplitude of the transmitted and measured signals, a slight linear phase delay is also observed. A Signal Correction Factor (SCF) is proposed to adjust signals based on adhesive degradation evaluated using the measured modal damping. The benefits of the SCF for prognostics feature generation are demonstrated in the frequency domain for the  $A_0$  mode.

## 1. INTRODUCTION

Structural Health Monitoring (SHM) denotes in-situ and continuous on-line monitoring (Mickens, Schulz, Sundaresan, & Ghoshal, 2003) for detection and interpretation of adverse changes within a structure (Kessler & Pramila, 2007).

K. R. Mulligan et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

SHM is applicable to a number of fields aiming at reducing system life-cycle costs and down-time and increasing safety (Giurgiutiu & Bao, 2004). In the aerospace domain, this is done by integrating Non-Destructive Evaluation (NDE) techniques as autonomous systems into airframes (Kessler & Pramila, 2007). Increasing confidence in SHM solutions could also contribute to reduce maintenance costs by minimizing scheduled downtime through decreasing the frequency and duration of these maintenance intervals (Kapoor, Boller, Giljohann, & Braun, 2010). Such failures depend on the host structure material which is typically aluminum or a Carbon Fiber Re-inforced Polymer (CFRP).

Data gathered from SHM systems for airframe diagnostics are used to detect and localize fatigue damage only. SHM systems based on Guided Wave (GW) propagation using piezoceramic transducers are among the various existing systems that could be used for data gathering. Piezoceramic transducers are advantageous due to their cost-effectiveness and lightweight and can be used for quick and continuous structural inspections (Quaegebeur, Masson, Langlois-Demers, & Micheau, 2010).

Prognostic enhancements are currently being investigated for SHM systems to improve condition-based monitoring. Damage prognosis (DP) is defined as an estimate of an engineered system's Remaining Useful Life (RUL) (Farrar & Lieven, 2007). These enhancements aim to monitor tendencies within SHM data to classify the various airframe failure modes (Kessler & Pramila, 2007) and then estimate the RUL of the host structure.

The use of CFRPs for host structure fabrication has increased as reported by (Kessler & Pramila, 2007). For CFRPs, the two main failure modes due to impacts include intra-ply matrix cracking and inter-ply delaminations. Intra-ply matrix

cracking occurs on the host structure surface due to low impact energies prior to inter-ply delaminations. Inter-ply delaminations are the result of either singular high energy impacts or repeated low velocity impacts which also cause intra-ply matrix cracks to spread and weaken the host structure as described in (Iarve, Gurvich, Mollenhauer, Rose, & Dávila, 2011). Prediction using model-based prognostics of inter-ply delamination size versus increasing impact projectile velocity is reported in (Mueller et al., 2009) where a numerical model is used to simulate an impact using a steel ball projectile (Choi, 1990). Their work did not follow ASTM standards that dictate a 16 mm hemispherical impact head must be used to impact composite coupons with a mass and height based on the coupon thickness such that realistic and repeatable damages (over a number of coupons) are induced (ASTM, 2007). Preliminary work described in (Ahmad & Gupta, 2010) for simulating and predicting the failure behavior of composite panels follows the ASTM standard for damage initiation but also presents only numerical results. Further numerical work for predicting crack growth and RUL are presented in (Coppe, Pais, Kim, & Haftka, 2010). In this work, crack growth RUL estimation is predicted using material properties and the stress intensity factor which is a complicated function of applied loading, boundary conditions, crack location, geometry, and material properties.

In the aforementioned numerical work, features are extracted from the simulations to form a feature set. The most representative and discriminate features from the set are then used as inputs to predictive models such as: K-Nearest Neighbor (KNN), Neural Networks, and Decision Trees. For GW, discriminative features are extracted from time, frequency, and energy domain signals (Kessler & Pramila, 2007). There is little knowledge however on which features can predict gradual tendencies from damage initiation which is important for RUL estimation and at what point these features can be used for RUL prediction after the damage initiation phase commences. Although much numerical work is presented in literature for damage prognosis and generating feature sets, there is little numerical work and nearly no experimental work in RUL estimation. Experimentally, little work has been put forth in proposing a robust data gathering methodology in a laboratory environment. Most successful RUL estimation approaches in various domains use a hybrid system that implements a combination of model and data driven based prognostics (Byington, Roemer, & Gallie, 2002).

An important consideration when using a piezoceramic transducer SHM system for data gathering prior to prognostic model development, is that impacts inflicted upon aerospace structures causes both structural damage and damage to the SHM system. More specifically, it is shown in (Park, Farrar, Lanza di Scalea, & Coccia, 2006) that impacts in regions around piezoceramic SHM systems mainly damage the bonding layer between the transducers and the host structure sur-

face. Degradation of the bonding layer in turn affects the signals generated and received by these transducers shown in (Mulligan, Quaegebeur, Ostiguy, Masson, & Létourneau, 2012) for a glass plate as the host structure. Numerical simulations were used to determine a damage metric based on modal damping of the transducer determined from electrical admittance curves to assess the level of bonding layer degradation. A Signal Correction Factor (SCF) has been proposed to correct for changes in signal amplitude and phase caused by degradation which is measured from changes in modal damping. Numerical results of the modal damping metrics capability of measuring the amount of bonding layer degradation are verified experimentally due to the transparency of the glass host structure as chemical degradation was used.

In this paper, the modal damping metric is used to assess the amount of bonding layer degradation on a CFRP host structure numerically. Experimental measurements are used to validate numerical results by controlling the bonding coverage area of piezoceramic lead-zirconate-titanate (PZT) transducers using Teflon masks. Following this, two transducers are bonded to an aerospace grade CFRP plate in a Pitch-Catch (PC) configuration. One transducer is perfectly bonded to the structure as a reference whereas the other is bonded using Teflon masks in order to simulate a damage in the adhesive. Following the Pitch-Catch measurements, the damaged transducer is removed, the surface of the plate is cleaned, and a new transducer is bonded using a different Teflon mask. The Pitch-Catch measurements are repeated. During acquisition, the reference and damaged transducer take turns as an actuator or sensor and data are post processed to investigate the effect of degradation on actuation and sensing. A SCF is proposed to adjust the amplitude and phase of the measured signals based on bonding layer degradation using the modal damping metric. The SCF is used to adjust the  $A_0$  dispersion curve to account for bonding layer degradation to demonstrate its importance in ensuring a robust database is gathered for later prognostic model development.

## 2. BONDING LAYER DEGRADATION MODES DUE TO IMPACTS

The effect of structural damage due to impacts on the bonding layer between a PZT transducer and the host structure has not been explicitly studied. Much work has however been done on the effect of impacts on adhesive bonded joints. The adhesive layer in composite lap joints with traction and impact loading remains intact while failure occurs at the adherends. Further damage propagation in the adherends causes the adhesive to gradually debond at the extremities of the lap joint (Galliot, Rousseau, & Verchery, 2012). Although a major portion of the impact energy is absorbed by the adherends, the remaining energy is transferred to the adhesive layer itself. With increasing impact energies, the adhesive layer is susceptible to failure modes including: a reduction in initial

Item	Material	Thickness (mm)	Young's Modulus (GPa)	Poisson's ratio	Density (kg/m <sup>3</sup> )
PZT	PZT 5A	0.5	65.0	0.31	7750
CFRP	16 plies prepreg [0,90]8s	2.45	74.5	0.219	1488
Adhesive	Permatex <sup>®</sup> cyanoacrylate	0.1	2.4	0.3	1100

Table 1. Properties of the PZT transducer, CFRP, and adhesive layer.

contact areas through particle deformation and delamination due to shearing motion (Islam & Chan, 2004). The remaining impact energy does not however affect the Young's modulus for brittle adhesives (such as cyanoacrylate) (Sugaya, Obuchi, & Chiaki, 2011). The lack of resistance to impact damage of adhesives is primarily related to their density. Although density does not change with increasing impact energy, there is a lesser effect of damage on the adhesive layer if it has a high density core due to reduced buckling (Nguyen, Jacombs, Thomson, Hachenberg, & Scott, 2005).

### 3. PIEZOCERAMIC DEBONDING METRICS

For SHM system health verification, variations in the electrical admittance have been shown to play a meaningful role in the detection of transducer debonding from host structure surfaces under environmental and mechanical loading conditions (Park et al., 2006) (Mulligan et al., 2012). The electrical admittance of a piezoceramic transducer  $\gamma(\omega)$  is defined as the ratio between the resulting current  $I$  and voltage  $V$  at a given angular frequency  $\omega$ . The theoretical admittance is approximated using a lumped parameter system using a simple RC series model when neglecting the mechanical resonance of the PZT transducer. This model is shown (Mulligan et al., 2012) to provide a good approximation of the low frequency admittance but performs poorly for higher frequencies due to the occurrence of resonances which cannot be modeled. Therefore, around and above resonance, a basic LCR circuit shown in Fig. 1 must be used with series resistance  $R_s$ , electrostatic capacitance  $C_s$ , electrical compliance  $C_p$ , mechanical mass  $L_p$ , and mechanical damping  $R_p$ , where  $R_s$  and  $R_p$  account for energy dissipation (Kim, Grisso, Kim, Ha, & Inman, 2008).

$$\xi = \frac{1}{2R_p} \sqrt{\frac{L_p}{C_p}} \quad (1)$$

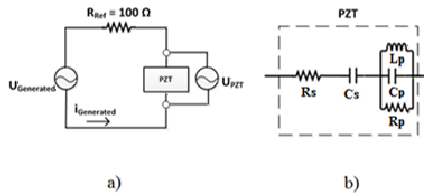


Figure 1. Admittance measurement circuit (a). RLC model of an unloaded PZT that accounts for mechanical resonance and damping (b).

From the electrical admittance curves, the modal damping  $\xi$  described by Eq. (1), which has been shown to describe gradual transducer debonding on a glass host structure, is extracted from estimation of the lumped parameter system (Mulligan et al., 2012). The constants of the lumped parameter system are determined by minimizing the distance between modeled and measured admittances for both the real and imaginary parts. Minimization algorithms are used to find the minimum of constrained multi-variable functions by substituting values for all variables using an initial estimate over a number of iterations and are available within the MATLAB framework (*fmincon* & *ga*).

The modal damping metric can be used to assess the level of adhesive coverage degradation and using the amplitude and phase change curves constructed for a particular host structure, a Signal Corrector Factor (SCF) can be developed as described in the next section.

## 4. EXPERIMENTAL AND NUMERICAL QUANTIFICATION OF PIEZOCERAMIC BONDING LAYER DEGRADATION

### 4.1. Numerical assessment of the degradation process

The first step in this study is to investigate the degradation process of bonded PZTs and relate it to the modal damping metric extracted from the electro-mechanical problem in section 3. To do this, a numerical model has been developed (shown in Fig. 2) using commercial FEM software COMSOL. An axisymmetric 2D model in the frequency domain has been used for simplicity and to reduce computation time. To mimic the experimental setup presented later in section 4.2, a 5 mm circular PZT is attached to a 2.5 mm quasi-isotropic carbon fiber plate using a 0.1 mm thick bonding layer whose properties (shown in Tab. 1) are representative of cyanoacrylate. The carbon fiber plate is modeled as an isotropic structure using equivalent properties (shown in Tab. 1) obtained from mechanical testing of the orthotropic plate used experimentally and Perfectly Matched Layers (PML) of 20 mm are used to avoid boundary reflections.

The model is meshed using 15 000 triangular elements for a total number of 60 000 degrees of freedom (DOF) such that a maximum mesh size of 0.1 mm is ensured and that at least 10 nodes exist per minimum resolvable wavelength. A voltage of 1V is simulated at the upper electrode of the PZT and current is estimated at the same electrode to extract the electrical admittance. The modal damping  $\xi$  is estimated using methods described in section 3. In this numerical work, the variation of adhesive coverage degradation mode is tested. The

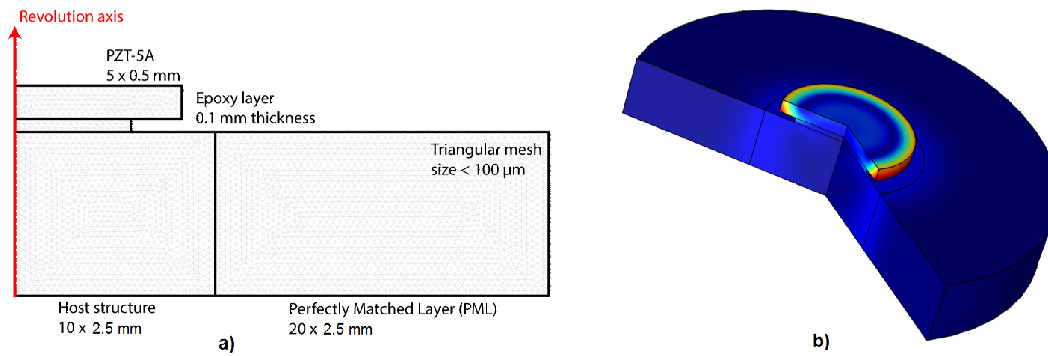


Figure 2. Illustration of the FEM mesh used for computation (a) and 3D reconstruction of the displacement field (b).

influence of adhesive coverage is estimated by changing the adhesive surface from 100 % (perfectly bonded adhesive) to 25 % (nearly debonded) by steps of 10 %. The results of the simulation are presented together with experimental results in section 4.3.

#### 4.2. Experimental procedure

Experiments are carried out to relate the degradation process for bonded PZTs to the modal damping metric extracted from the electro-mechanical problem through gradual degradation. Transducers are bonded to the surface of a carbon fiber plate with dimensions 52 cm x 52 cm using 0.01 ml of cyanoacrylate measured from a syringe and a static load of 125 kPa. Adhesive coverage reduction was controlled using Teflon inserts during the PZT bonding process as in (Lanzara, Yoon, Kim, & Chang, 2009). Teflon is ideal for controlling the adhesive coverage under the PZT due to its low modulus of elasticity (Dupont, 2012). Almost no energy is transferred through the Teflon from the PZT into the structure leaving the remaining bonding layer as the conductive path. Adhesive coverage reduction using Teflon is chosen in this work as the primary adhesive failure mode because adhesive coverage is easily controlled experimentally compared to changes in the bonding layer Young’s modulus which is degraded using ultraviolet (UV) light or temperature cycling and measured using traction testing or a rheometer which both require access to the adhesive under the PZT (Lotters, Olthuis, Veltink, & Bergveld, 1997). The Young’s modulus effect is considered as negligible (Skaja, Fernando, & Croll, 2006).

Voltage bursts are generated at the PZT, and the impedance is measured using a voltage divider circuit with a 100 Ω 5 % resistor. The input signals are generated using an HP33120A generator with a sampling frequency of 10 MHz. The generated signals are amplified using a MusiLab UA-8400 amplification system. The signal acquisition is performed using a high impedance National Instruments® PCI-5105 12-bit DAQ board configured through a custom LabVIEW interface. The signals are recorded at a fixed sampling frequency

of 10 MHz. All measurements are averaged 500 times in order to increase SNR and low-pass filtered at 1.5 MHz. The transfer function for the admittance over the entire frequency range below 1 MHz is obtained using broadband generation of guided waves through sub-band decomposition with N = 11 sub-bands for reconstruction between 10 kHz and 1 MHz as presented in (Quaegebeur, Masson, Micheau, & Mrad, 2012). A sub-band frequency step of 100 kHz is selected to ensure perfect reconstruction. The modal damping is extracted from the admittance curve as described in (Mulligan et al., 2012).

#### 4.3. Numerical and experimental results of bonding layer degradation

The numerical and experimental results for modal damping for 4 degradation trials and the unbonded case are presented in Fig. 3. Simulations show that modal damping decreases with degradation over a range of 85 %. Experimentally, the same

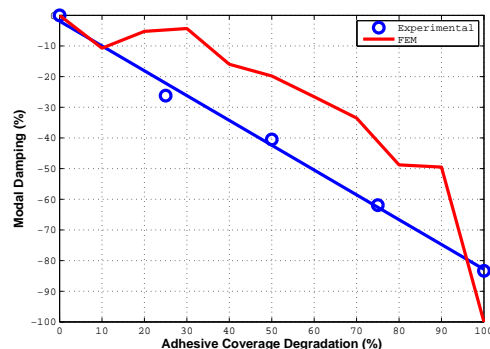


Figure 3. Numerical and experimental results for modal damping over adhesive coverage degradation.

tendencies are observed as in the simulated curve with good agreement. For a linear curve fit to the experimental measurements a coefficient of determination ( $R^2$ ) of 0.9941 is observed designating small linear discrepancy. Due to the degradation of the bonding layer simulated by the insertion

of Teflon, the PZT transducer is enabled to move more freely with less constraint and therefore damping decreases (Nader, Silva, & Adamowski, 2004).

Metrics based on capacitance and the resonance frequency of the PZT are investigated in (Park et al., 2006) and (Mulligan et al., 2012). The capacitance metric performs well in detecting complete transducer debonding but performs poorly in measuring gradual adhesive degradation. This is largely due to resonance peaks that disrupt a linear tendency for the capacitance to increase with debonding. The resonance frequency metric is robust to such peaks and shows a linear tendency to decrease with bonding layer degradation. The sensitivity of the metric is shown in (Mulligan et al., 2012) to be quite low with a range of only 10 %. The modal damping metric is a combination of the capacitance and resonance frequency metrics and is robust to resonance peaks while providing a range of 100 %.

## 5. EFFECTS OF DEGRADATION ON ACTUATION AND SENSING OF GUIDED WAVES

### 5.1. Experimental setup

In order to relate bonding degradation to the actuation and sensing performance of SHM systems that use PZT transducers, the influence of adhesive layer degradation in a Pitch-Catch configuration is proposed in this section.

For the experimental setup, two transducers are separated by a distance of 20 cm, and fixed to a carbon fiber plate using cyanoacrylate. One transducer is used as a reference, while the other one is used for evaluation of both a degraded actuator and sensor. The Pitch-Catch configuration is surrounded by an absorbing perimeter to attenuate edge reflections. As in section 4.2, Teflon is used to control the amount of adhesive coverage under the PZT transducer as a degradation technique. Between each data acquisition trial, the degraded transducer is removed, the surface is cleaned, and a new transducer is bonded to the plate surface in the same location using a different Teflon mask.

The broadband generation of guided waves through sub-band decomposition technique presented in (Quaegebeur et al., 2012) is used to estimate the transfer functions for amplitude and phase versus frequency below 1 MHz. For a given adhesive coverage degradation, the reference transducer first generates the broadband signals and the degraded transducer is used as a sensor response to estimate the effect of degradation on sensing. Then the degraded transducer is used as an actuator and the reference transducer as a sensor. To determine the effect of degradation on actuation, the amplitude and phase are then frequency averaged for both actuation and sensing for three frequency ranges: below ( $< 250$  kHz), around (250 kHz - 700 kHz), and above ( $> 700$  kHz) mechanical resonance. The adhesive coverage area is known from the Teflon

mask areas. The error bars are calculated from the standard deviation of the magnitude of the FFT over each frequency range. The sample size for each standard deviation is approximately 1400 frequency points. The amplitude and phase are presented in Fig. 4 for sensing and in Fig. 5 for actuation.

### 5.2. Effect of sensor degradation

The amplitude change for a PZT used as a sensor versus adhesive coverage degradation below mechanical resonance is presented in Fig. 4 (a). With degradation, an overall consistent tendency for the amplitude to decrease or increase is not observed. Initially, the amplitude increases 25 % from the perfectly bonded case after 50 % adhesive coverage degradation. Following this, the amplitude decreases 60 % between 50 % - 75 % adhesive degradation. It is also important to note that the error bar at 50 % degradation is 60 % larger than for the other degradation trials. Increasing error bars with adhesive coverage degradation was observed in (Mulligan et al., 2012) with increasing bonding layer degradation largely due to shifting resonance peaks into lower frequency ranges from around and above resonance. These shifting resonance peaks can provide false indications that the amplitude for a specific frequency increases with degradation because the change in amplitude in the figure is determined as an average over the entire frequency range. Smaller frequency range selections can avoid such ambiguities.

The amplitude change versus adhesive coverage degradation around resonance is presented in Fig. 4 (b). In this case, the amplitude decreases with adhesive degradation over a range of 80 %. Again a large error bar is observed at 50 % degradation. The error bar is however 80 % smaller than that of the 50 % degradation trial in low frequency.

The amplitude change versus adhesive coverage degradation above resonance is presented in Fig. 4 (c). As in the around resonance range, the amplitude decreases over a range of 70 %. The error bar in the 25 % degradation trial is now 5 % larger than that at 50 % degradation. This further suggests that resonance peaks are shifting out of high frequency into the other frequency ranges.

The phase change versus adhesive coverage degradation below, around, and above resonance is presented in Fig. 4 (d), (e), and (f). In Fig. 4 (d), the phase of the sensor signal below resonance is presented. With degradation, the phase increases slightly ( $5^\circ$ ) from the perfectly bonded case. In Fig. 4 (e), the phase of the sensor signal around resonance is presented. The results show that the phase does not change significantly. For frequencies above resonance shown in Fig. 4 (f), the phase increases slightly ( $3^\circ$ ) with degradation.



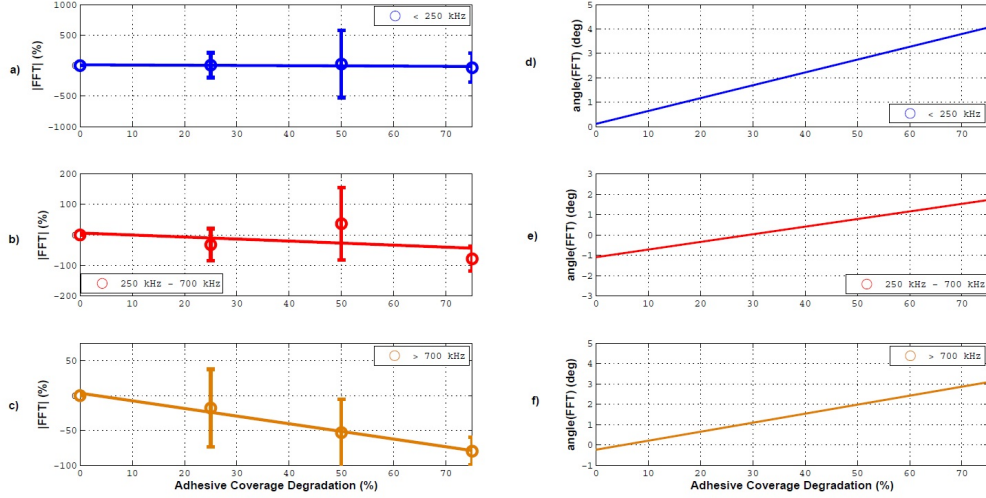


Figure 4. Changes in the amplitude (a,b,c) and phase (d,e,f) of sensor signals with adhesive coverage degradation for three frequency ranges (< 250 kHz: a,d; 250 kHz - 700 kHz: b,e; > 700 kHz: c,f).

### 5.3. Effect of actuator degradation

The amplitude change versus adhesive coverage degradation for a PZT used as an actuator is presented in Fig. 5. In each case, for amplitude (Fig. 5 (a), (b), and (c)), the actuator mimics the same tendencies as the sensor with bonding layer degradation over the same frequency ranges. The amount of amplitude change is in the same range as in the sensor case for all frequency ranges. For the phase, presented in Fig. 5 (d), (e), and (f), each range shows an overall increase with bonding layer degradation. The low and high frequency range increases are identical to those of the sensor case. Around resonance, an increase of  $3^\circ$  is observed. It has been found in literature that bonding layer degradation affects generated signal phase in low frequency which could lead to false indication on structural conditions but the effect has not been studied in ranges higher than resonance (Overly, Park, & Farinholt, 2009).

### 5.4. Signal correction factor (SCF) for Pitch-Catch measurements

Using the amount of degradation obtained through the modal damping metric and curves that describe changes in amplitude and phase from the perfectly bonded state, a Signal Correction Factor (SCF) is proposed and assessed in the following. The SCF removes biasing in SHM gathered signals from bonding degradation to potentially improve structural damage and RUL estimates (Mulligan, Masson, Létourneau, & Quaegebeur, 2011).

Variations in a signal measured by a transducer within a SHM system are caused by degradation of the host structure  $\Delta s^{\text{damaged}}(t)$  and degradation of the transducer bonding layer  $\Delta s^{\text{degraded}}(t)$ . The measured signal  $s'(t)$  can be viewed as the sum of the two components caused by damage on top of a

pristine signal  $s(t)$  shown in Eq. (2).

$$s'(t) = s(t) + \Delta s^{\text{degraded}}(t) + \Delta s^{\text{damage}}(t) \quad (2)$$

The SCF is derived from the modal damping metric which assesses the level of bonding layer degradation. The change in amplitude and phase for a given degradation and frequency range of the generated signal which are measured empirically, are then used to adjust the captured signal to compensate for the bonding layer degradation measured by the modal damping metric. Next, the changes in amplitude and phase are used as inputs to a formulation for removing the PZT degradation effect to reveal captured signal changes due only to structural damage which is shown in Eq. (3), where the Fast Fourier Transform (FFT) of the measured signal  $s'(t)$  is multiplied by an amplitude constant  $k$  and a phase shift  $e^{j\phi}$ . Taking the inverse (IFFT) of the transformed signal leaves the pristine signal changed by structural damage  $\Delta s^{\text{damaged}}(t)$ . The SCF is therefore defined as  $SCF = ke^{j\phi}$ .

$$s(t) + \Delta s^{\text{damage}}(t) = IFFT[k e^{j\phi} FFT(s'(t))] \quad (3)$$

The SCF works on the principle that the changes to measured signals due to bonding layer degradation are measurable. Once the amplitude and phase curves with bonding layer degradation are developed either numerically or experimentally for a given material, the modal damping metric is used to assess the level of degradation. The level of degradation is the abscissa on the amplitude and phase curves and with the appropriate adjustment constants, the SCF can be used.

In order to assess the SCF for an extended frequency do-

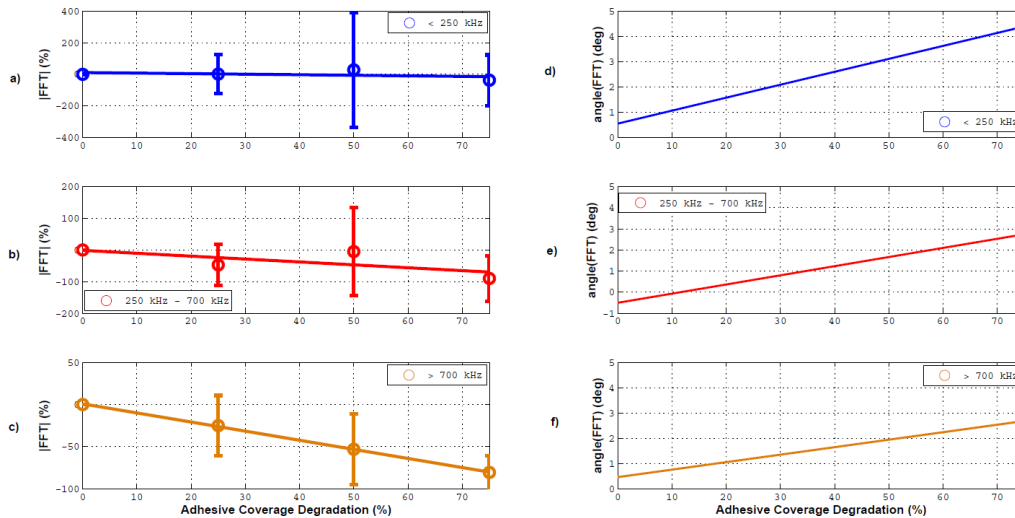


Figure 5. Changes in the amplitude (a,b,c) and phase (d,e,f) of actuator signals with adhesive coverage degradation for three frequency ranges (< 250 kHz: a,d; 250 kHz - 700 kHz: b,e; > 700 kHz: c,f).

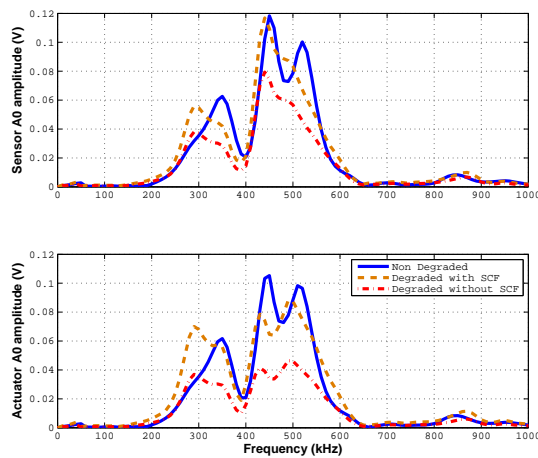


Figure 6. Degraded, non-degraded, and adjusted  $A_0$  modes using the SCF for the sensor and actuator.

main and provide a practical example of feature generation for prognostic model development, the effect of the SCF applied to the  $A_0$  modes is presented in Fig. 6 for a degraded sensor and actuator with 25 % bonding layer coverage degradation. The amplitude of the mode  $A_0$  is extracted separately from the dispersion curves constructed from the signals using sub-band reconstruction (Quaegebeur et al., 2012) over the frequency range below 1 MHz.

In the sensor case, the SCF performs well in adjusting the amplitude of the  $A_0$  mode over the entire frequency range. Around resonance, there is a frequency shift of 10 kHz between the non-degraded and degraded signals that is not accounted for in the SCF. There is also attenuation and defor-

mation of the resonance peaks. These alterations were not observed in (Mulligan et al., 2012). The main difference in the experimental method in this study compared to (Mulligan et al., 2012) is the use of Teflon in the degradation process. Teflon could add damping to the signals received by the PZT and also alter its resonance frequency.

In the actuator case, the SCF performs well in adjusting the amplitude of the  $A_0$  mode over the entire frequency range as in the sensor case. The frequency shift is also observed between the non-degraded and degraded signals although it is a slightly larger shift of 20 kHz compared to the sensor case. Further investigation is required in which the Teflon is removed after the adhesive cures under the PZT. Good performance of the SCF in low frequency is critical in damage imaging (EUSR (Giurgiutiu & Bao, 2004) and Excitelet (Quaegebeur et al., 2012)) and feature generation algorithms as they strongly depend on estimating the Time of Flight (ToF) and amplitude of the  $A_0$  wave packet.

## 6. CONCLUSIONS AND DISCUSSIONS

This paper presented an approach for the correction of data gathered for damage prognosis (DP) in composite structures. To do this, numerical and experimental investigations of the modal damping metric to assess and compensate the degradation of the adhesive layer of surface-bonded PZT transducers for SHM applications is presented. Modal damping curves suggest that the amount of bonding layer degradation is measurable. Bonding layer degradation leads to a reduction in the actuation and sensing amplitude and at higher frequencies, a delay in the time signal is observed. With this information, a Signal Correction Factor (SCF) is used to adjust signals generated or sensed from PZT transducers with a degraded bond-

ing layer. A practical example was demonstrated for the  $A_0$  sensor and actuator signal amplitudes which can be useful in feature generation for prognostics model development.

#### ACKNOWLEDGMENT

This study has been conducted with the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) and the National Research Council of Canada (NRC).

#### REFERENCES

- Ahmad, S., & Gupta, N. K. (2010). Probabilistic analysis of composite panels under low velocity impact. In *Proceedings of the IMPLAST conference*.
- ASTM. (2007). *Standard test method for measuring the damage resistance of a fibre-reinforced polymer matrix composite to a drop-weight impact event* (Tech. Rep. Nos. D7136/D7136M-07). American Society for Testing and Materials.
- Byington, C. S., Roemer, M. J., & Gallie, T. (2002). Prognostic enhancements to diagnostic systems for improved condition-based maintenance. In *Aerospace conference proceedings*.
- Choi, H. (1990). *Damage in graphite/epoxy laminated composites due to low-velocity impact*. Unpublished doctoral dissertation, Stanford University.
- Coppe, A., Pais, M. J., Kim, N.-H., & Haftka, R. T. (2010). Identification of equivalent damage growth parameters for general crack geometry. In *Annual conference on prognostics and health management*.
- Dupont. (2012). *Teflon PTFE fluoropolymer resin: Properties handbook* (Tech. Rep.). Author.
- Farrar, C. R., & Lieven, N. A. J. (2007). Damage prognosis: The future of structural health monitoring. *Phil. Trans. R. Soc. A*, 365, 623–632.
- Galliot, C., Rousseau, J., & Verchery, G. (2012). Drop weight tensile impact testing of adhesively bonded carbon/epoxy laminate joints. *International Journal of Adhesion and Adhesives*, 35, 68–75.
- Giurgiutiu, V., & Bao, J. J. (2004). Embedded-ultrasonics structural radar for in situ structural health monitoring of thin-wall structures. *Structural Health Monitoring*, 3(2), 121–140.
- Iarve, E. V., Gurvich, M. R., Mollenhauer, D. H., Rose, C. A., & Dávila, C. G. (2011). Mesh-independent matrix cracking and delamination modeling in laminated composites. *Int. J. Numer. Meth. Engng*, 88, 749–773.
- Islam, R. A., & Chan, Y. C. (2004). Effect of drop impact energy on contact resistance of anisotropic conductive adhesive film joints. *J. Mater. Res.*, 19(6), 1662–1668.
- Kapoor, H., Boller, C., Giljohann, S., & Braun, C. (2010). Strategies for structural health monitoring implementation potential assessment in aircraft operational life extension considerations. In *Proceedings of the 2nd international symposium on NDT in aerospace*.
- Kessler, S. S., & Pramila, R. (2007). Pattern recognition for damage characterization in composite materials. In *Proceedings of the aiaa sdm conference*.
- Kim, J., Grisso, B. L., Kim, J. K., Ha, D. S., & Inman, D. J. (2008). Electrical modeling of piezoelectric ceramics for analysis and evaluation of sensory systems. In *IEEE sensors applications symposium*.
- Lanzara, G., Yoon, Y., Kim, Y., & Chang, F.-K. (2009). Influence of interface degradation on the performance of piezoelectric actuators. *Journal of Intelligent Material Systems and Structures*, 20(14), 1699–1710.
- Lotters, J. C., Olthuis, W., Veltink, P. H., & Bergveld, P. (1997). The mechanical properties of the rubber elastic polymer polydimethylsiloxane for sensor applications. *J. Micromech. Microeng.*, 7, 145–147.
- Mickens, T., Schulz, M., Sundaresan, M., & Ghoshal, A. (2003). Structural health monitoring of an aircraft joint. *Mechanical Systems and Signal Processing*, 17(2), 285–303.
- Mueller, I., Larrosa, C., Roy, S., Mittal, A., Kuldeep, L., & Chang, F.-K. (2009). An integrated health management and prognostic technology for composite airframe structures. In *Annual conference on prognostics and health management*.
- Mulligan, K. R., Masson, P., Létourneau, S., & Quaegebeur, N. (2011). An approach to compensate for the degradation of the monitoring system in damage detection. In *Proceedings of the Canadian Institute for NDE*.
- Mulligan, K. R., Quaegebeur, N., Ostiguy, P.-C., Masson, P., & Létourneau, S. (2012). Comparison of metrics to monitor and compensate for piezoceramic degradation in structural health monitoring. *Structural Health Monitoring*.
- Nader, G., Silva, E. C. N., & Adamowski, J. C. (2004). Effective damping value of piezoelectric transducer experimental techniques and numerical analysis. In *Abcm symposium series in mechatronics*.
- Nguyen, M. Q., Jacombs, S. S., Thomson, R. S., Hachenberg, D., & Scott, M. L. (2005). Simulation of impact on sandwich structures. *Composite Structures*, 67, 217–227.
- Overly, T. G., Park, K., & Farinholt, M. (2009). Piezoelectric active-sensor diagnostics and validation using instantaneous baseline data. *IEEE Sensors Journal*, 9(11), 1414–1421.
- Park, G., Farrar, C. R., Lanza di Scalea, F., & Coccia, S. (2006). Performance assessment and validation of piezoelectric active-sensors in structural health monitoring. *Smart Materials and Structures*, 15(6), 1673–1683.
- Quaegebeur, N., Masson, P., Langlois-Demers, D., & Micheau, P. (2010). Dispersion-based imaging for structural health monitoring using sparse and compact arrays. *Smart Materials and Structures*, 20(1), 1–12.
- Quaegebeur, N., Masson, P., Micheau, P., & Mrad, N. (2012). Broadband generation of ultrasonic guided waves using sub-band decomposition. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*.
- Skaja, A., Fernando, D., & Croll, S. (2006). Mechanical property changes and degradation during accelerated weathering of polyester-urethane coatings. *Journal of Coatings Technology and Research*, 3(1), 41–51.
- Sugaya, T., Obuchi, T., & Chiaki, S. (2011). Influences of loading rates on stress-strain relations of cured bulks of brittle and ductile adhesives. *Journal of Solid Mechanics and Materials Engineering*, 5(12), 921–928.

# Application of Symbolic Regression to Electrochemical Impedance Spectroscopy Data for Lubricating Oil Health Evaluation

Carl Byington<sup>1</sup>, Nicholas Mackos<sup>2</sup>, Garrett Argenna<sup>3</sup>, Andrew Palladino<sup>4</sup>, Johan Reimann<sup>5</sup>, and Joel Schmitigal<sup>6</sup>

<sup>1,2,3,4,5</sup>*Impact Technologies, Rochester, NY, 14623, USA*

*Carl.Byington@impact-tek.com*  
*Nicholas.Mackos@impact-tek.com*  
*Garrett.Argenna@impact-tek.com*  
*Andrew.Palladino@impact-tek.com*  
*Johan.Reimann@impact-tek.com*

<sup>6</sup>*US Army TARDEC, Warren, MI, 48092, USA*

*Joel.A.Schmitigal.civ@mail.mil*

## ABSTRACT

The authors have applied an advanced set of auto-regressive tools for identifying potentially complex, linear and non-linear relationships in data, wherein the underlying physical relationships are not well described. In this paper these tools and techniques are described in detail, and the results of the application of these tools to evaluation of diesel engine lubricating oil health (based on electrochemical impedance spectroscopy data) is detailed. It is demonstrated that highly accurate models can be constructed which take as input features derived from diesel engine lubricating oil electrochemical impedance spectroscopy data and output estimates of traditional laboratory based oil analysis parameters. The electrochemical impedance spectroscopy and laboratory analytical data used are from a field deployment of oil condition sensors on several long-haul class 8 diesel trucks. The dataset was divided into training and test datasets and goodness of fit metrics were calculated to evaluate model performance. Models were successfully generated for nitration, soot content, total base number, total acid number, and viscosity.

## 1. INTRODUCTION

An on-line oil condition monitoring device for application to vehicular diesel engines provides significant benefit over traditional oil sampling methods. The online nature of the monitoring device eliminates the long delays associated with traditional laboratory analysis and prevents the possibility of sampling errors. Knowledge of the actual

condition of the oil at a particular time also allows for the real time adjustment of oil drain intervals – either extending to take advantage of additional remaining useful life or shortening to prevent engine damage due to abnormal fluid conditions or contaminations. Maintenance actions can also be planned and carried out opportunistically.

It has long been known that electrochemical impedance spectroscopy (EIS) can provide valuable insight into the condition of lubricating oils and their additive packages (Byington et al 2010, Moffatt et al 2012). In order to mature this understanding research within this field has focused on characterizing the relationship between lubricating oils and electrochemical impedance spectroscopy. Lvovich V F. and Smiechowski M. F. (2011, 2008, 2006, 2005, 2002, 2001) are the primary contributors to this characterization and have produced several well behaved models of the relationship. While these models provide tremendous insight into lubricant chemistry, they are based on empirical data from laboratory grade instrumentation and known oil formulations and contaminants. For on-line lubricant monitors, the oil formulation and contamination is unknown and therefore samples must be drawn and traditional oil analysis performed. These traditional laboratory tests typically output lubricant chemical and mechanical properties such as Total Acid Number (TAN), Total Base Number (TBN), percent soot content, viscosity, and degree of nitration, among others.

The work presented in this paper extends the scope of previous modeling research by establishing a direct map between on-line oil sensor features and the underlying oil chemistry assessed through traditional laboratory analysis. While correlations have been observed between these on-line EIS data and those values which represent the output of traditional laboratory oil analysis (Mackos et al 2008),

---

Byington et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

models have not been developed to explicate this relationship. While EIS data alone can be used to generate lubricant remaining useful life estimates, using models to estimate traditional laboratory oil analysis parameters provides additional benefits; for example, historical condemnation thresholds established using these traditional oil analysis parameters can be leveraged.

The physics of the relationship between measured electrochemical data and laboratory test outputs like TAN and TBN is vastly complex. Therefore the explicit definition of transfer functions to translate EIS data into the desired laboratory test features is difficult and impractical. Several methods exist for modeling complex scientific data. When expert knowledge of the parametric relationships between measured data are known, fixed-form models can be applied. In the case that relationships are not well understood, numerical models are often used. E.g neural networks, naïve bayes classifiers. These models however do not explain discovered relationships intuitively and thus do not easily distill data into scientific knowledge. Instead, the authors have pursued the application of symbolic regression techniques which require no a priori knowledge of the functional relationship between the inputs and desired outputs of such a model and result in a closed form solution which may describe physical and chemical relationships more clearly.

The authors have been working with US Army Tank Automotive Research, Development and Engineering Center (TARDEC) to develop next generation hardware for online oil condition monitoring. As part of this effort, and with the cooperation of the National Automotive Center, several sets of existing oil condition monitoring hardware were deployed on long haul class 8 trucks. A periodic oil sampling and laboratory analysis plan was also implemented. These laboratory analyticals and EIS data were used to evaluate the capability of symbolic regression techniques to generate models for estimating TAN, TBN, nitration, soot content, and viscosity.

## 2. SYMBOLIC REGRESSION OVERVIEW

The main objective of this effort was to correlate laboratory generated tribology results with sensor generated electrochemical impedance spectroscopy data. While there is prior understanding of the chemical and physical nature of oil and how it interacts with contaminants and other breakdown processes, this understanding has never directly resulted in models that correlate tribology data to EIS data. Given the ground truth information this is a supervised learning problem and since the tribology data is not discretized, a regression method is appropriate (rather than classifier methods such as logistic regression, neural networks, support vector machines, etc.). Multi-variant Linear regression is the obvious and standard method; if the specific model for optimization is known then symbolic

regression is unnecessary. If however the model is unknown, the application of linear regression is a labor intensive process, to include adding and subtracting features, increasing and decreasing the complexity of features included, cross-validation, and regularization. The application of Symbolic Regression, and the toolsets which were employed, effectively automate these processes. Symbolic regression also provides significant benefit over linear regression when the ultimate goal is to deploy the models in an embedded environment. Like linear regression a closed form equation is generated, however the operations for inclusion in the solutions identified can be defined ahead of time; in this manner any limitations of the embedded platform can be accounted for. Solutions of varying levels of complexity can also be generated and evaluated to trade off performance in terms of accuracy and computational complexity.

The Symbolic Regression algorithm described in this section is used to identify general and potentially complex relationships, in this case between the online oil-condition monitor observations and associated laboratory generated oil chemical and mechanical properties. The Symbolic Regression algorithm (Koza, 1992) is a generalization to the standard regression problem formulation in that it requires very few assumptions regarding the underlying regression model and the output of the algorithm is a closed form expression that can easily be implemented on an embedded platform. The produced closed form expressions can be non-linear and have temporal dependencies and as a result important information such as leading fault (temporal) or cyclic degradation (non-linear) can be identified using this technique. In short, the Symbolic Regression technique is an excellent choice when faced with complex problems where many of the underlying physical behaviors of a system are not well described.

The Symbolic Regression tool used for this analysis relies on Genetic Programming (Koza, 1998) to search for the best functional/algebraic map between the produced oil condition monitor features and the oil analysis results reported by the laboratory. The Genetic Programming algorithm evaluates a pool of symbolic expressions represented by a collection of parse trees (one such tree is depicted in Figure 1) and iteratively applies candidate selection, cross-over and mutation operations to generate the most effective expressions. The fitness of each expression can be evaluated using many different metrics; however, for the analysis performed in this work, the mean absolute error was used. As with most data driven modeling tools, special attention must be paid to avoid over-fitting the derived model to the provided data. For Symbolic Regression the over-fitting problem is addressed at two different levels. First, the algorithm provides a Pareto front of optimal solutions that allows the researcher to select the ideal solution in terms of functional complexity and performance. For instance, if a simple solution performs only slightly

worse than a much more complex expression, the Symbolic Regression tool will provide both solutions to the researcher who can then select the correct solution in terms of complexity and performance. By providing this functionality, it is possible to eliminate overly complex expressions that tend to be highly tuned to the training data. The second technique used to prevent over-fitting is the standard cross-validation approach. That is, the generated expressions are optimized to fit a training set, say 80% of the original data, but when evaluating the performance of the expressions the remaining 20% of the data is used. This simple approach reduces the likelihood of over-fitting.

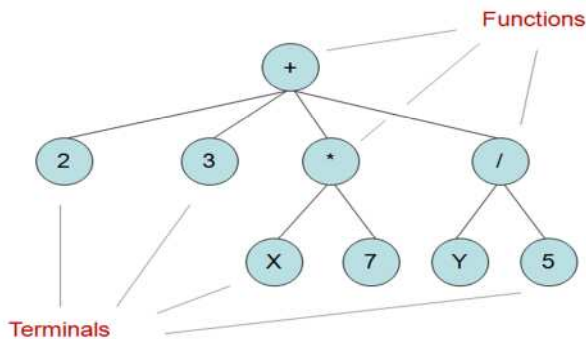


Figure 1. An example of a parse tree corresponding to the expression  $2+3+x*7+Y/5$ .

In addition to the ease of implementing the derived expressions on an embedded platform, it is also possible to analyze the individual terms in each of the expressions to determine what their impact may be on the overall model response. This sensitivity analysis step provides insight into how important each term is, and also into what features should be generated by the oil condition monitor.

It is worth noting that the Symbolic Regression analysis is only performed during the development of the oil assessment model. That is, the symbolic regression process will not be running on the sensor itself; only the functional output of the symbolic regression process would be considered for embedded implementation. It should also be noted that Symbolic Regression analysis tools are freely available to developers through the Eureqa software developed by a group of researchers at Cornell University (Schmidt & Lipson, 2009). This tool is mature and allows users to distribute the search task over a large number of computers through Amazon Cloud Services <http://aws.amazon.com/>.

### 3. APPLICATION TO DIESEL ENGINE LUBRICATING OIL

#### 3.1. Description of the Dataset Used

The underlying technology for the oil condition monitor detailed herein is electrochemical impedance spectroscopy (EIS), wherein the fluid under test is subjected to a dynamic

electrical signal and the fluid's effects on the signal are measured and correlated to various chemical and physical phenomena. The oil condition monitor's embedded algorithm trends temperature-normalized and filtered electrochemical impedances measured at a high frequency (HF), medium frequency (MF), and low frequency (LF).

As previously described, a field deployment on several long haul class-8 trucks was used to generate the necessary EIS and laboratory analytical data for this effort. Across these installations, online oil condition monitoring devices collected data continuously for several months, resulting in a dataset which spanned more than ten oil changes.

Throughout most of the test period, oil samples were taken from the vehicles and sent to a third party laboratory for analysis. Three of the trucks in the installation were selected for inclusion in the symbolic regression study based on the quality and consistency of their corresponding data sets.

In the following section, the output of the models generated through the application of the previously described symbolic regression techniques are presented against laboratory analytical data for comparison.

#### 3.2. Symbolic Regression Results

Symbolic regression models were created for the following laboratory generated analyticals: nitration, TBN, TAN, Soot and viscosity. These models are represented by closed form mathematical expressions suitable for implementation in embedded hardware. An example of the kind of expressions that comprised these models is given in Eq. (1) below, wherein *Feature 1* is one of the electrochemical features generated by the oil condition monitor and *X*, *Y*, and *Z* are constants.

$$TBN \text{ est.} = \log(\text{Feature 1} - XeY) - Z \quad (1)$$

To ensure that the model did not over-fit the data the model performance metrics were computed by performing cross-validation using 50% of the data. For each laboratory analytical a single model was created based on data from all of the trucks so that the repeatability of the model across different oil condition monitoring hardware and different vehicles could be evaluated.

In Figure 2 below, the model based Nitration estimate is represented by black data points. The vertical lines indicate when an oil change occurred. As expected, the nitration level dropped after each oil change. The squares indicate the nitration measurement made off-line through laboratory analysis using oil samples drawn from each truck during the test. The squares are plotted along the x-axis according to when the sample was drawn. Also note that more online EIS results have been acquired than laboratory analytical data at the time of writing of this paper. This is especially the case for truck 3, for which oil samples have not yet been received for operation after Jan 21<sup>st</sup>.

The model performs well across all three trucks and across the entire test. The only laboratory measurement that did not line up well with model results was the first sample drawn in March on truck 1. It is more likely that the laboratory measurement is wrong rather than the sensor model estimate since it is unlikely that the nitration level increased, decreased and increased again across one oil cycle; this is not typical of nitration trending. The sensor installed on truck 2 also underestimated nitration on the fifth oil cycle.

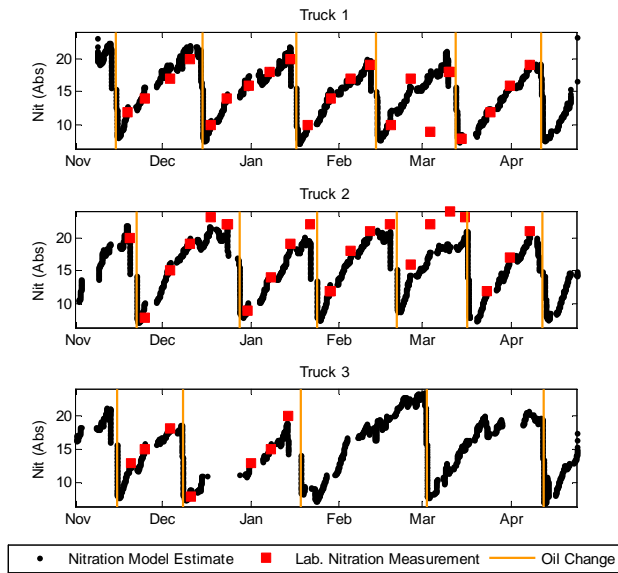


Figure 2: Nitration estimate plotted with Lab Nitration Measurements

Statistical results are summarized in Table 1 and a histogram of the differences between the model predicted values and the ground truth is depicted in Figure 3.

Standard Deviation of the Residual	1.7934 (Abs)
Mean of the Residual	-0.0380 (Abs)

Table 1: Nitration Model Performance Metrics

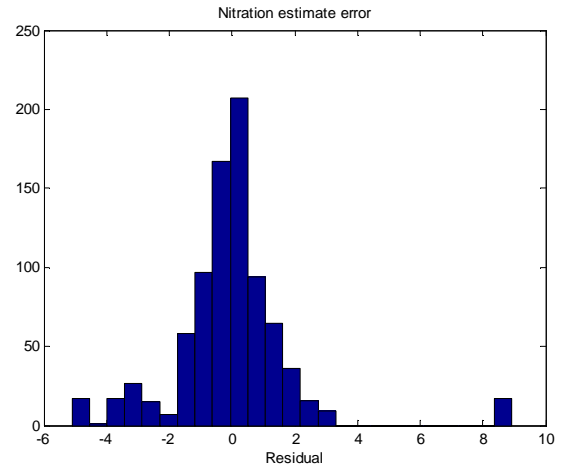


Figure 3: Histogram capturing error between the model generated Nitration value and the laboratory results

The model appears to perform well given that the Nitration values observed in the laboratory data ranged from 6 to 24 (Abs). That is, the standard deviation of the modeling error is 9.9% of the range of the laboratory measurements.

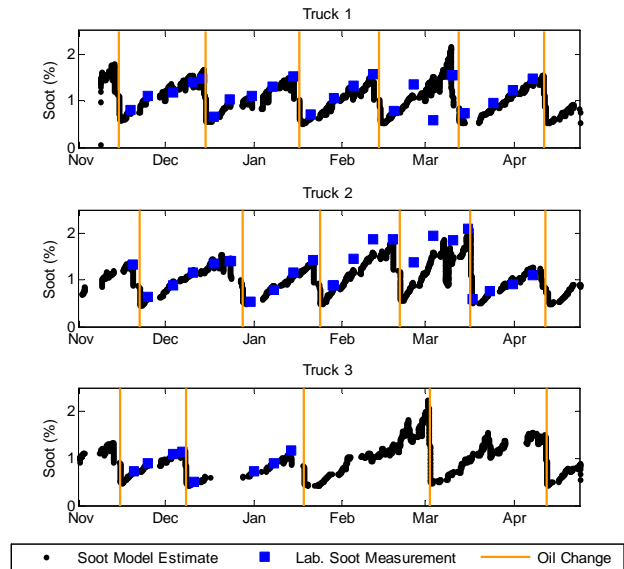


Figure 4: Soot estimate plotted with Lab Soot Measurements

Figure 4 shows the results from the same data set but applying the soot estimation model and comparing with laboratory soot measurements.

The model performs well across all three trucks and across the entire test. The first sample drawn in March on truck 1 continues to line up poorly with inline data. This means it is most likely due to a misrepresentative oil sample being drawn / analyzed. The sensor installed on truck 2 also underestimated soot content on the fourth oil cycle.

The statistical results are summarized in Table 2 and a histogram of the differences between the model predicted values and the ground truth is depicted in Figure 5.

Standard Deviation of the Residual	0.0722(%)
Mean of the Residual	-0.1373 (%)

Table 2: Soot Model Performance Metrics

The model appears to perform well given that the Soot values observed in the laboratory data ranged from 0.5 to 2 (%). That is, the standard deviation of the modeling error is 4.5% of the range of the laboratory measurements. Based on this observation, the soot estimation model was the highest performer among the 5 models calculated.

Figure 6 shows the results from the same data set but applying the TBN estimation model and comparing them with laboratory TBN measurements.

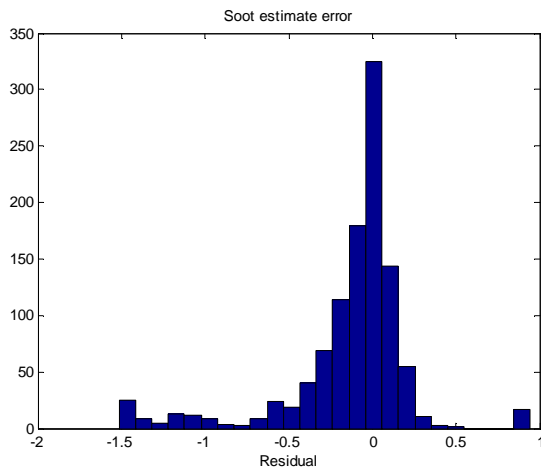


Figure 5: Histogram capturing error between the model generated Soot value and the laboratory results

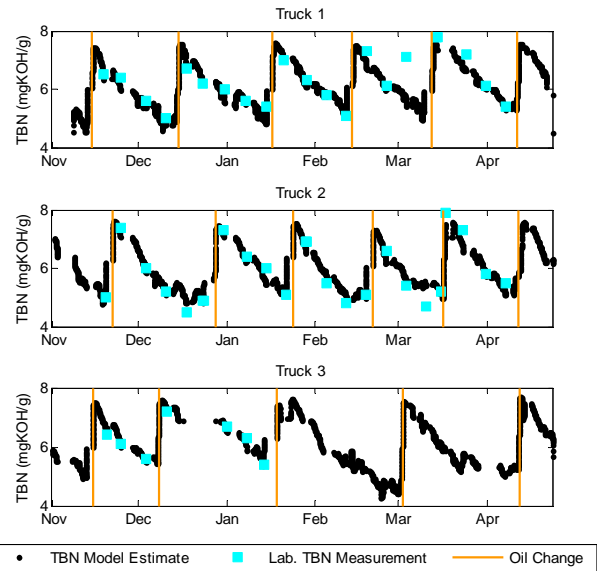


Figure 6: TBN estimate plotted with Lab TBN Measurements

The model performs well across all three trucks and across the entire test. There are a greater number of extreme outliers than the Soot and Nitration models produce but still a healthy performance within one standard deviation as is shown in Table 3 below.

Standard Deviation of the Residual	0.3220 (mgKOH/g)
Mean of the Residual	-0.1940 (mgKOH/g)

Table 3: TBN Model Performance Metrics

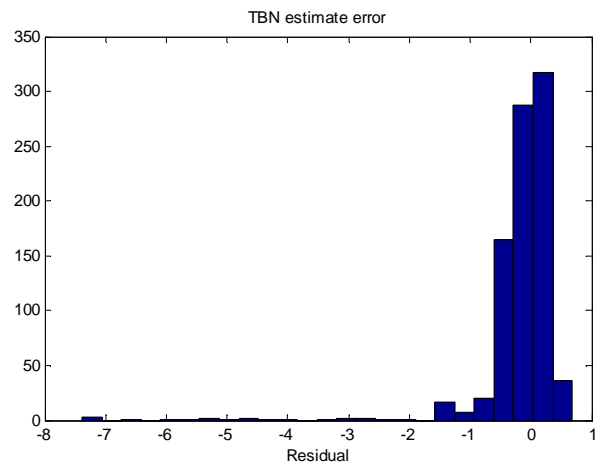


Figure 7: Histogram capturing error between the model generated TBN value and the laboratory results

Given that the TBN values observed in the laboratory data ranged from 2.8 to 7.9 (mgKOH/g), the standard deviation of the modeling error is still only 6.3% of the range of the



laboratory measurements. Therefore while there are a greater number of residual outliers than Nitration for example, it still out performs the Nitration model the majority of the time.

Figure 8 shows the results from the same data set but applying the TAN estimation model and comparing with laboratory TAN measurements.

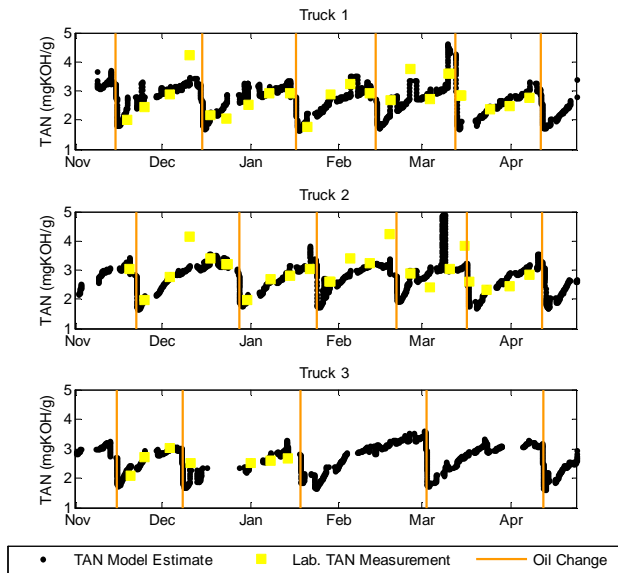


Figure 8: Soot estimate plotted with Lab. Soot Measurements

TAN appears to be the worst performer of the five models created. However the model does appear to show a correlation. The statistical performance shows that the model does not perform well enough to be relied upon.

Standard Deviation of the Residual	1.2993 (mgKOH/g)
Mean of the Residual	0.5829 (mgKOH/g)

Table 4: TAN Model Performance Metrics

Considering that the laboratory data ranged from 1.79 to 4.22 (mgKOH/g), the standard deviation of the modeling error is over 50% of the range of laboratory measurements. In other words the confidence bounds of the estimate extend to over half the range of typical data.

Finally, Figure 10 shows the results from the same data set but applying the Viscosity estimation model and comparing with laboratory Viscosity measurements.

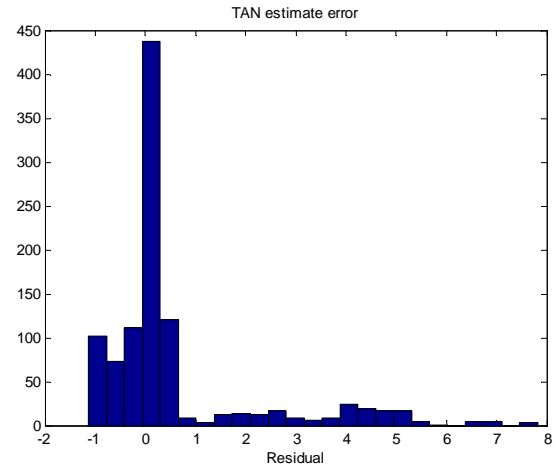


Figure 9: Histogram capturing error between the model generated TAN value and the laboratory results

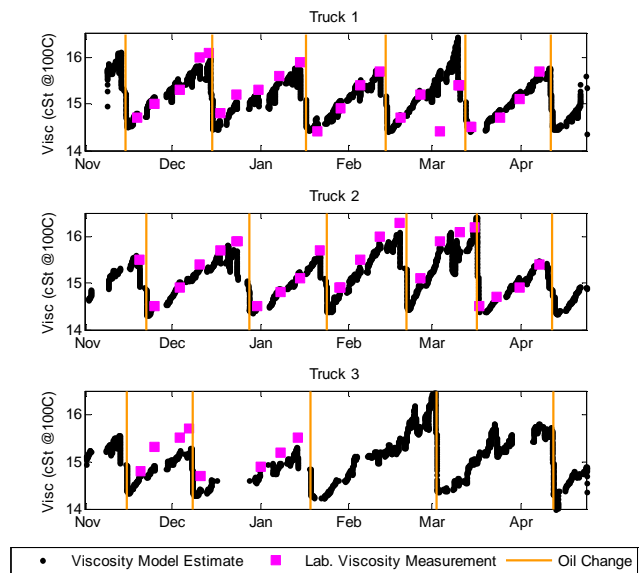


Figure 10: Viscosity estimate plotted with Lab. Viscosity Measurements

The model performs well across all three trucks with perhaps a slightly weaker performance implemented on truck 3 data. The statistical results are summarized in Table 5 and Figure 11.

Standard Deviation of the Residual	0.1188 (cSt @100C)
Mean of the Residual	-0.0673 (cSt @100C)

Table 5: Viscosity Model Performance Metrics

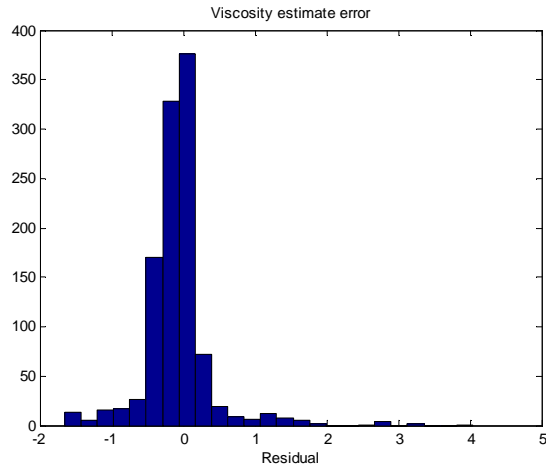


Figure 11: Histogram capturing error between the model generated Viscosity value and the laboratory results

The model appears to perform well given that the Viscosity values observed in the laboratory data ranged from 14.4 to 16.3 (mgKOH/g). That is, the standard deviation of the modeling error is 6.3% of the range of the laboratory measurements.

### 3.3. RUL Estimation Plan

The data shown in Figure 2, Figure 4, Figure 6, Figure 8 and Figure 10 can be reconditioned to display the features vs. hours on oil by identifying top-ups and oil changes and adjusting the time on oil accordingly. The resulting reconditioned data for Nitration, Soot, TBN and Viscosity estimates are shown from Figure 12 through Figure 15. The bands of data are represented by a family of feature curves. The traditional oil analysis results are also plotted on each plot as colored squares. Each deterministic curve, after filtering for noise, is monotonically increasing and can be fitted to a general function form,  $n^{\text{th}}$  order or exponential, depending on the feature type. As one can readily see, while there is significant spread of the values, the trend on each is clear and a regressive model can be used to predict a future threshold exceedence on any parameter.

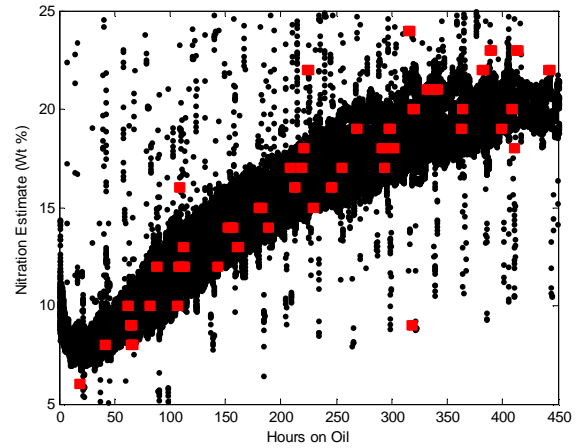


Figure 12: Hours on Oil vs. Nitration Estimate

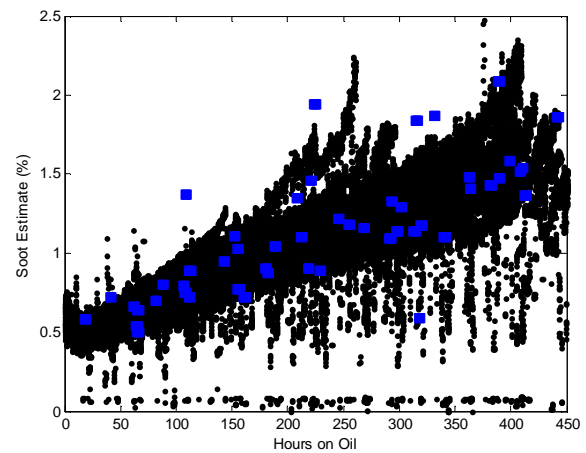


Figure 13: Hours on Oil vs. Soot Estimate

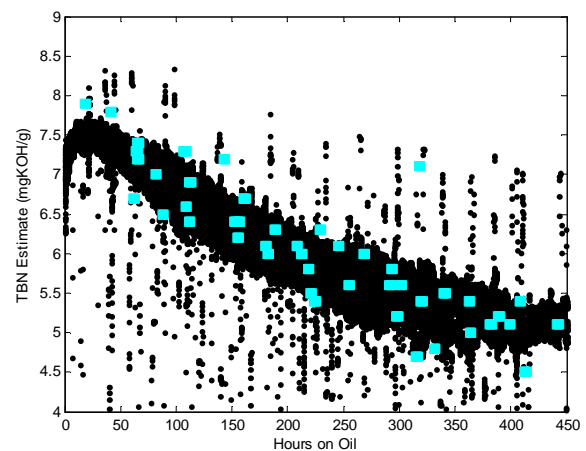


Figure 14: Hours on Oil vs. TBN Estimate

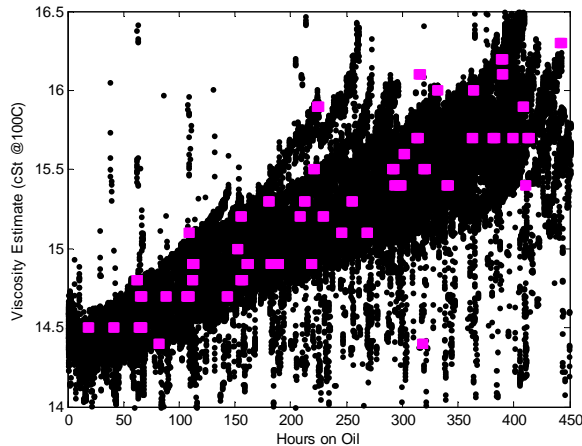


Figure 15: Hours on Oil vs. Viscosity Estimate

Given the nature of the data and in order to better approximate the uncertainty band for each considered feature, a Monte Carlo method was chosen to estimate remaining useful life probabilistic outputs. The approach starts with identifying the core parameter drivers of the model and assigning an initial distribution to each variable. The drivers can be identified by performing a sensitivity analysis to quantify the influence of a parameter with respect to the probabilistic outputs, such as the remaining useful life distribution. A Monte Carlo simulation is then performed and consists of randomly sampling from the initial distributions and running the models into the future over a predefined operating profile.

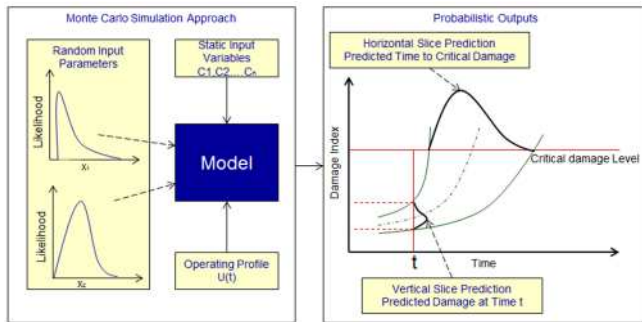


Figure 16: Monte Carlo Probabilistic Method

Figure 16 is an illustration of the approach. By sampling the initial parameter distribution, a family of model curves is generated and can be used to calculate vertical or horizontal slice predictions. A horizontal slice is generally taken at the critical damage level and will generate a distribution on time to critical damage which also represents the remaining useful life probabilistic output. A vertical slice is taken at any point in time and represents a distribution on predicted damage at specified time  $t$ .

It was also determined to limit the number of input parameter distributions to three or less. The more distributions are being sampled from, the more simulations are needed to obtain a better approximation on the uncertainty spread. Different sampling methods, such as importance sampling, can be applied to reduce the simulation time and still output a reasonable approximation of the spread.

One of the advantages of this approach is the ability to optionally update the initial input distributions given ground truth information. The underlying assumption is that if the module has access to accurate oil analyticals, these results can be used to update the initial distributions. By producing more accurate initial conditions for the prognosis model, the system is capable of improving the subsequent prognosis results.

The RUL determination is a direct adaptation of the authors prior work in health-based prognostics. The prior demonstrated method uses Particle Filters to perform feature trend predictions (Zhang et al., 2008). Particle filtering is an application of Bayesian state estimation that calculates an a posteriori probability density function (PDF) of a state of a system based on a priori observations or measurements. If the calculation of the future state of the system is extended in multiple steps with the use of a model, the particle filtering algorithm can perform long term predictions. In this case, the system observations are initially used to build a PDF of the “present” or “current” system condition, as illustrated conceptually in Figure 17.

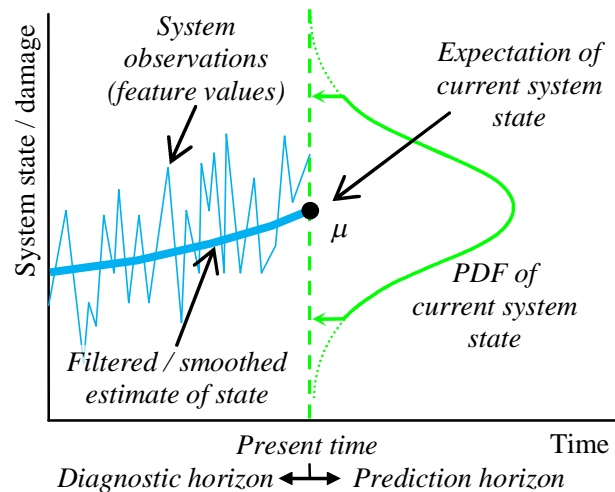


Figure 17. Determination of the state of a system as a PDF based on feature values

This PDF is then sampled into “particles” representative of potential system states with individual weights. Using the model, the prognostic algorithm simulates the progression of the weights in time to do a prediction of possible future system states, as illustrated in Figure 18.

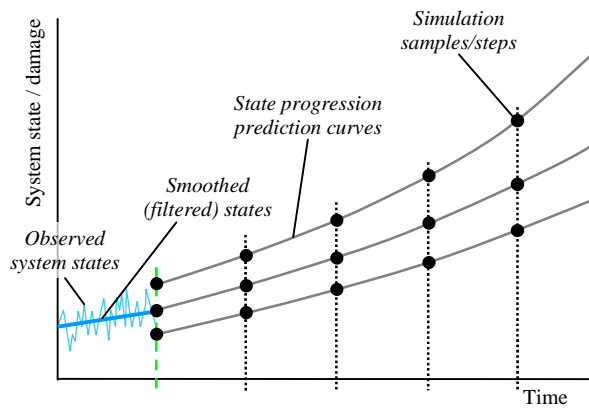


Figure 18. System state prediction and progression curves

Just as with the initial state, future states of the system can be represented by PDFs. Once the progression of the system state has been determined, the algorithm can be used to predict the time required for the system to reach a condition of interest, such as a need for maintenance. The condition predicted is represented by a “prediction threshold” line. Because there is uncertainty in the future system states (as represented by the different state progression curves), there is also uncertainty in the predicted time to reach the threshold. This uncertainty in time is represented also by a PDF, referred to as the “time-to-threshold” (TTT) PDF. The definition of prognostic confidence is tied to how the area of the TTT PDF is divided. To determine the minimum time remaining to reach the prediction threshold, called the “just-in-time” point, a confidence specification is required. Figure 19 illustrates how a 95% prediction confidence is used to determine the just-in-time point. This approach has been successfully used in a range of different mechanical and electrical prognosis problems.

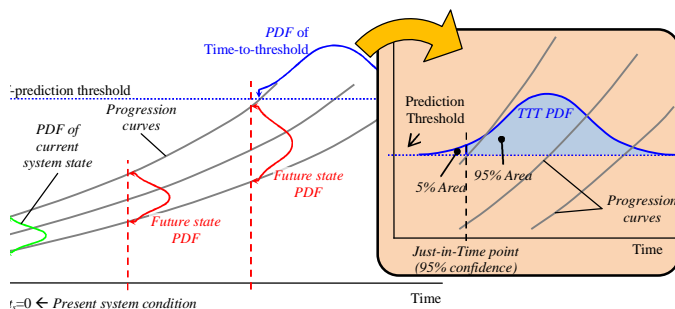


Figure 19. Determination of the prediction time to reach a prognostic threshold with a given prognostic confidence (the inlay box provides an example using 95% confidence)

#### 4. CONCLUSION

Of the five models created, all but one performed well enough for embedded implementation in a next generation

online oil condition monitor, with the highest performing estimate being soot estimation, and TBN and Viscosity models a close second.

It should be noted however that the laboratory data ranges mentioned for each analytical also represent boundaries within which the model can be implemented. If, for example, soot content extends above 2%, the model can no longer be trusted to perform with the stated accuracy. Once data is acquired outside the range of this data set, the models can be matured to handle the increased range and the performance of the model will have to be reassessed.

Another limitation of the model is the singular oil type in use during this field trial. To remove potential installation or vehicle specific artifacts contained in the models they will need to be verified against more diverse data sets.

Lastly the approach and framework has been offered to extend this regressive analysis approach to perform real-time prediction of oil RUL. Several specific methods were offered to handle the uncertainty and also produce a useful prediction in automated software. Realization of this technology will not only allow for improved equipment protection and enhance the underlying oil-wetted component effective reliability with its ability to look for contaminants and aging/wear out mechanisms, but it will also allow both oil sampling/lab tests and oil changes to be performed on a predictive condition-based schedule. Thus, this technology has the ability to provide significant return of value to the operator and maintainer as well as provide environmental/green movement benefits with the reduction in oil usage and subsequent disposal.

#### ACKNOWLEDGEMENT

The authors would like to acknowledge several organizations that contributed to this work in some substantial and positive manner. Specifically:

- Lubrizol Corporation, for their support of this field trial, and provision of oil condition monitoring hardware which generated a portion of the data used in this analysis; also, for their technical contribution to the development of the next generation oil condition monitoring hardware.
- US Army TARDEC for their financial and technical guidance in support of the next generation oil condition monitoring hardware development efforts, and the field trial which generated the presented data.
- Southeastern Freight Lines for their support of the field trial and provision of the vehicles onto which the oil condition monitoring devices were installed.
- ATA and the Technology & Maintenance Council for their support of the field trial.

## NOMENCLATURE

<i>EIS</i>	Electrochemical Impedance Spectroscopy
<i>TAN</i>	Total Acid Number
<i>TBN</i>	Total Base Number
<i>TARDEC</i>	Tank Automotive Research, Development and Engineering Center

## REFERENCES

- Moffatt, J., Byington, C.S., Minnella, C. M., "Lubricant Condition Assessment System (LUCAS), an Enabler for Condition Based Maintenance Best Practice", American Helicopter Society Annual Forum, Fort Worth, TX, May 2012.
- Byington C., Palmer C., Argenna G., Mackos N. "An Integrated, Real-Time Oil Quality Monitor and Debris Measurement Capability for Drive Train and Engine Systems" American Helicopter Society 66th Annual Forum and Technology Display, 2010
- Mackos N., Baybutt M., Palmer C., Tario J.; "Providing Embedded, In-situ Oil Quality Monitoring for Improved Maintenance and On-Board Diagnostics in Trucking and Automotive Applications" SAE Int. J. Commer. Veh. 1(1):260-267, 2008.
- Koza J. R., Genetic Programming: On the Programming of Computers by Means of Natural Selection. (MIT Press, Cambridge, MA, 1992).
- Koza J.R. Genetic Programming, MIT Press, ISBN 0-262-11189-6, 1998
- Schmidt M., Lipson H. (2009) "Distilling Free-Form Natural Laws from Experimental Data," Science, Vol. 324, no. 5923, pp. 81 - 85.
- Lvovich V F., Electrochemical Impedance Spectroscopy Characterization of Electrorheological Fluids, Crane Aerospace and Electronics, Elyria, Ohio, May 9<sup>th</sup> 2011
- Lvovich V F., Smiechowski M. F., Non-Linear Impedance Analysis of industrial lubricants, *Electrochim. Acta*, 53, pp. 7375-7385, 2008.
- Lvovich V F. and Smiechowski M. F., Impedance Characterization of Industrial Lubricants, *Electrochimica Acta*, vol. 51, no. 8-9, pp. 1487-1496, 2006.
- Smiechowski M. F., Lvovich V F., Characterization of Carbon Black Colloidal Nanoparticles by Electrochemical Impedance Spectroscopy, *J. Electroanal. Chem.*, 577(1), pp. 67-78, 2005.
- Smiechowski M. F., Lvovich V F., Electrochemical Monitoring of Water-Surfactant Interactions in Industrial Lubricants, *J. Electroanal. Chem.*, 534 (2), pp.171-180, 2002.
- Smiechowski M. F., Lvovich V F., On-Line Electrochemical Sensors for Monitoring Time-Dependent Water-Polymer Interactions in Industrial Lubricants. *Chemical and Biological Sensors and Analytical Methods*, Proceedings Volume 2001-18, M.Butler, P. Vanysek, N. Yamazoe Eds., The

Electrochemical Soc., Inc., Pennington, NJ, pp. 442-453, 2001.

- Lvovich V F., Riga A. T. and Cahoon J., Characterization of Organic Surfactants and Dispersants By Frequency-Dependent Dielectric Thermal Analysis and Electrochemistry, *Materials Characterization by Dynamic and Modulated Thermal Analytical Techniques*, ASTM Special Technical Publication 1402, A. Riga and L. Judovits, Ed., American Society for Testing and Materials, West Conshohocken, June 2001, 157-173.
- Lvovich V F., Boyle F., "Method for On-Line Monitoring of Condition of Non-Aqueous Fluids", U.S. Patent Application 20070151806, granted March 2008.
- Pérez A., Hadfield M., "Low-Cost Quality Sensor Based on Changes in Complex Permittivity" *Sensors* 2011, Volume 11, doi:10.3390/s111110675
- Zhang, B., Sconyers, C., Byington, C.S., Patrick, R., Orchard, M.E., and Vachtsevanos, G.J. "Anomaly Detection: A Robust Approach to Detection of Unanticipated Faults". International Conference on Prognostics and Health Management, Denver, Colorado, October 6-9, 2008.
- Byington, C. S., Patrick, R., Smith, M. J., Vachtsevanos, G. J., "Integrated Software Platform for Diagnostics and Prognostics with Air Vehicle HUMS, 7<sup>th</sup> DSTO International Conference on Health and Usage Monitoring, Melbourne, Australia, March 2011.

## BIOGRAPHIES

**Carl Byington, P.E.** is a registered Professional Engineer and prior Owner and Technical Director at Impact Technologies, now a Sikorsky Innovations Company. He possesses over 22 years of design, analysis, and testing experience with fluid power, thermal, mechanical, and electrical systems. He has performed as the Principal Investigator (PI) for innovative Prognostics and Health Management (PHM) and Condition-based Maintenance (CBM) technologies on over 70 programs for the military, NASA, and other customers. Carl is a member of ASME, IEEE, AIAA, SAE, STLE and AHS professional societies. He is also a past Technical Program Chairman of the MFPT Society and the PHM Society for the 2010 and 2011 meetings. Since 2003, he has been a Lead Instructor for the annual "PHM/CBM Design" Short Course offered by Impact to government and industry participants. In prior employments, Carl was the CBM Department Head at the Penn State Applied Research Lab and performed propulsion research at NASA Langley Research Center. He possesses degrees in Mechanical Engineering from the University of Pennsylvania and Aeronautical Engineering from the George Washington University. He has produced well over 100 publications related to advanced sensing techniques; signal processing; vibration monitoring; diagnostics and control; prognostics; data fusion; equipment health

management; and condition-based maintenance. He is also coauthor on 5 US awarded or pending patents related to predictive monitoring technologies including one related to this oil condition monitoring measurement method.

**Nicholos Mackos** is a mechanical engineer and project manager at Impact Technologies, A Sikorsky Innovations Company. He has worked in the area of advanced diagnostics and prognostics development for 7 years, with a specific focus on sensing hardware development. Nicholos has co-authored many technical publications concerning advanced, smart-sensor development and the application of electrochemical impedance spectroscopy to condition monitoring for industrial and automotive lubricating oils. Nicholos obtained his B.S in Mechanical Engineering from Rochester Institute of Technology, and also holds a Masters Certificate in Applied Project Management from Villanova University.

**Garrett Argenna** is a Project Engineer at Impact Technologies, A Sikorsky Innovations Company, where he develops diagnostic and prognostic systems for military and commercial applications. His experience includes analyzing and interpreting a wide range of sensor data, developing prognostic health assessment algorithms, designing experiments and data acquisition systems. His specific areas of expertise include identifying and developing appropriate sensing methodologies to monitor degradation leading to failure modes, and developing algorithms to autonomously correlate sensor information with the current health of systems. Garrett received his B.S. in Mechanical Engineering with Honors from the Rochester Institute of Technology.

# Detecting injector deactivation failure modes in diesel engines using time and order domain approaches

Mitchell Lebold<sup>1</sup>, Scott Pflumm<sup>2</sup>, Jeffrey Banks<sup>3</sup>, Jonathon Bednar<sup>4</sup>, Karl Reichard<sup>5</sup>, Kenneth Fischer<sup>6</sup> and Joseph Stempnik<sup>7</sup>

<sup>1,2,3,4,5</sup> *The Pennsylvania State University  
State College, PA 16804-0030  
Telephone: (814) 865-8958  
lebold@arl.psu.edu  
jsp116@arl.psu.edu  
jcb242@arl.psu.edu  
jpb5007@arl.psu.edu  
kmr5@arl.psu.edu*

<sup>6,7</sup> *US Army RDECOM - TARDEC  
6501 E. 11 Mile Rd.  
RDTA-RS MS #204  
Warren, MI 48397-5000  
ken.fischer@us.army.mil  
joe.stempnik@us.army.mil*

## ABSTRACT

This paper documents the investigation of fuel injector fault detection methods for a seven liter diesel engine. This effort was conducted for the Tank Automotive Research Development Engineering Center (TARDEC) Condition Based Maintenance (CBM) team. The task was to develop algorithms capable of real-time detection of injector misfire events. The purpose of this task was to enable TARDEC's Engine Control Management (ECM) research and development efforts to evaluate the technical feasibility of integrating automated on-board condition monitoring algorithms with future ECM monitoring and control operations.

---

Mitchell Lebold et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Disclaimer: Reference herein to any specific commercial company, product, process, or service by trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the Department of the Army (DoA). The opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or the DoA, and shall not be used for advertising or product endorsement purposes.

Recommended Approval for Public Release - Distribution is Unlimited (Distribution A)

During this investigation, it was shown that multiple techniques can correctly detect and identify injector cylinder misfiring. Each individual technique has its own advantages, and this investigation focused on low computational power signal processing methods that would be suitable for embedding in an engine controller or processor. The list below includes six injector fault analysis approaches that were evaluated for this effort:

1. Injector signal based analysis
2. Vibration based analysis
3. Time domain speed encoder analysis
4. Order domain speed encoder analysis
5. FFT classifier selection techniques
6. Time domain classification techniques

This paper highlights the first four diagnostics techniques. On-platform tests are suggested for technique validation and future development of these initial findings.

## 1. INTRODUCTION

The Tank Automotive Research Development Engineering Center (TARDEC) Condition Based Maintenance (CBM) team investigated the development of a Vehicle Health Management Systems (VHMS) for ground combat platforms. The mission of the program was to direct the development of technologies to provide the US Army with an improved diagnostic, predictive and sustainment capability for ground vehicles.

In 2010, TARDEC conducted diesel engine dynamometer testing on a seven liter diesel engine. As part of the dynamometer test schedule, an engine testing tool was used to selectively deactivate individual fuel injectors over a range of speed and load conditions. This set of tests formed the basis for the data set used in this investigation.

This effort investigated multiple approaches and documented the advantages and limitations with varying degrees of robustness, cost and complexity in terms of instrumentation, hardware and software requirements. From failure mode, effects and criticality analysis (FMECA) studies of diesel engines shown by Banks, Hines, Lebold, Campbell, Begg and Byington (2001) can be used to detect failures that will produced the greatest value for achieving high operational availability and reducing maintenance costs. For this effort, we focused on detecting injector (or combustion) related faults. We emphasize the distinction that unlike the work developed in the past by Murphy, Lebold, Reichard, Galie, and Byington (2003) the approaches described here focused on commonly installed sensors. Moreover, the goal of this work was to develop simple data collection/processing schemes for embedded solutions.

The six injector fault analysis techniques studied as part of this effort can be grouped into three domains: time, frequency, and classification. Each technique is summarized below.

#### Time Domain Techniques:

- **INJECTOR SIGNAL BASED ANALYSIS**
  - Begin with calculating the timed based injector pulse profile.
  - Once the pulse profile is characterized, determine injector faults by categorizing pulse profile energy.
- **VIBRATION BASED ANALYSIS**
  - Investigate an alternative accelerometer-based approach utilizing only analog acquired signals.
  - Similar to the injector signal analysis, this approach also determines injector faults by the presence of energy in the vibration signal during the specific cylinder firing event window.
- **TIME DOMAIN SPEED ENCODER ANALYSIS**
  - Investigate fault detection approaches using the crankshaft speed encoder signal acquired from the digital timer board DAQ system.
  - Determine fault based on the instantaneous speed profile of the crank shaft.

#### Order Domain Techniques:

- **ORDER DOMAIN SPEED ENCODER ANALYSIS**
  - Investigate the order domain analysis of the crankshaft speed encoder signal acquired from the timer board DAQ system.
  - Determine fault based on the magnitude and phase at particular crankshaft orders.

#### Classification Domain Techniques:

- **GENERAL FAULT CLASSIFICATION AND REASONING ALGORITHMS**
  - Investigate fault detection methods using classification/reasoning algorithms on FFT frequency bin data.
- **TIME BASED CLASSIFICATION APPROACH**
  - Investigate fault detection methods using classification/reasoning algorithms on time domain data.

For this paper, only the time and order domain techniques will be discussed. The following sections will discuss the process for calculating injector signal profile and the four diagnostic techniques in further detail.

## 2. ENGINE INJECTOR ANALYSIS BACKGROUND

The purpose of injector timing analysis is to quantify the pulse duration and delay of the injection of fuel into the engine with respect to the piston's orientation. This is usually referenced to the piston's Top Dead Center (TDC) location of 0 degrees. Top dead center is the position of a piston in which it is farthest from the crankshaft.

In graphic terms the goal of this task is shown in Figure 1. The subsequent sub-sections describe the process by which this end result is obtained.

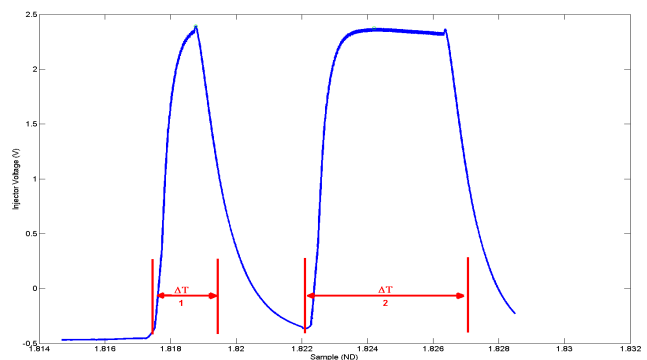


Figure 1. Graphic Illustration of Fuel Injector Timing Analysis Objective

The algorithm developed to perform this task calculates this pulse width for every injector actuation event that occurs



during the entire ten second recording. The algorithm is designed to differentiate and properly account for injector pulses consisting of single, double or triple pulse profiles.

The data in Figure 2 illustrates single, triple and double pulse profiles respectively. Single pulses were typically observed at relatively low speed around 700 revolution per minute (RPM) and low load (30 lbs-ft) conditions during the initial warm-up of the engine. Triple pulses were observed as speed-load conditions were increased. Typically the right-most pulse within the three pulse profile would gradually widen as speed-load continued to increase and eventually ‘absorb’ the center pulse, thereby resulting in the two pulse profile at the highest speed - load (2200 RPM - 560 lbs-ft) condition.

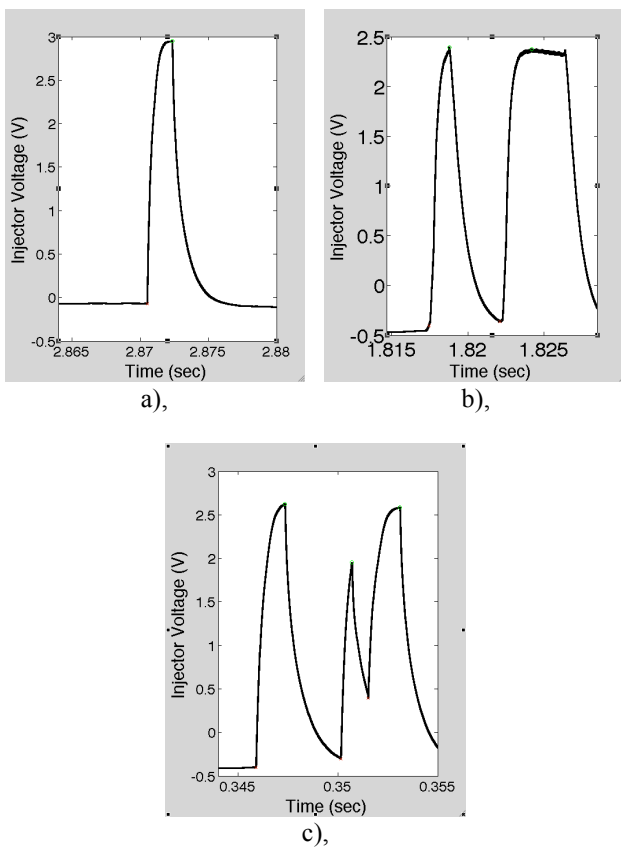


Figure 2. (a): Single Injector Pulse; (b) Triple Pulse; (c) Double Pulse

The algorithm stores the computed pulse duration ( $\Delta T$ ) and the relative delay/offset from Top Dead Center (TDC) for each respective pulse profile event occurring during the record set. In broad terms the analysis is divided into three sub-tasks:

1. Compute detection window
2. Determine true top dead center
3. Process injector pulses

While this processing technique is the basis for the first time-domain analysis approach, it is also useful for insuring that the engine controller is performing properly and that the engine is load balancing across the engine properly. The following section discusses the first time domain technique, which parses the data similarly to the approach just mentioned and looks at the energy within each combustion event window.

### 3. INJECTOR SIGNAL BASED ANALYSIS

This time-domain technique for injector fault detection looks at the presence of energy in the injector signal during the injection firing event window. Figure 3 illustrates the method by which signal energy is determined to exist in the detection window.

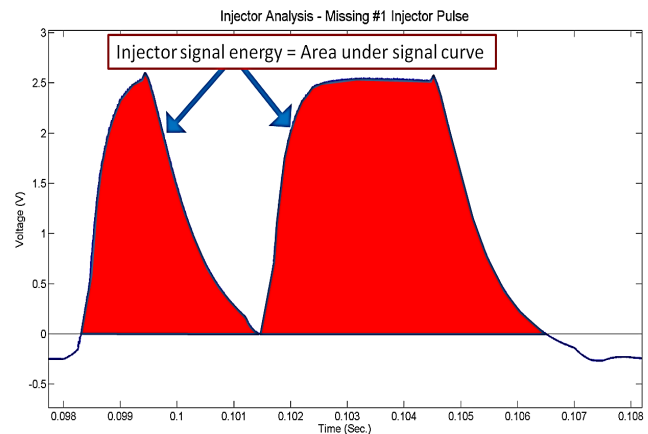


Figure 3. Detecting Injector Signal Energy in the Window of Interest

Figure 4 is a graphic representation illustrating how the algorithm identifies a cylinder three injector misfire. This plot is predicated on the fact the algorithm has identified TDC of cylinder one. Once the cylinder one TDC is located, the algorithm measures the injector signal energy in each detection window. If the signal energy does not exceed the threshold condition, the algorithm identifies the respective injector as a misfire fault.

The injector signal based detection approach resulted in 100% detection rate for all of the specific load, speed and fault conditions tested. No false positives or negatives were generated using this approach. The sensing requirements for this approach are an injector signal and a once per revolution signal (1PPR/TDC signal). The signal processing approach works under varying conditions regardless of the number of injector pulses produced per cylinder, (commonly one, two or three pulses). Furthermore, due to the low computation processing power needed for this approach, this simplistic technique can be easily embedded into an engine controller or small microprocessor.

The next method applies the same signal processing methodology to accelerometer (vibration) data collected from a location mounted on top of the engine heads.

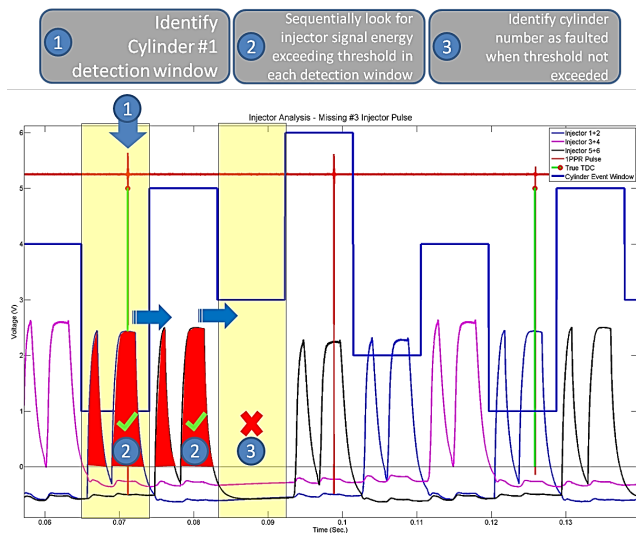


Figure 4. Detecting a Representative Cylinder Misfire Event

#### 4. VIBRATION BASED ANALYSIS

The motivation for investigating a vibration-based fault detection approach is to provide an alternative method for detecting combustion related faults. Therefore, such an approach could be used for engines without injector firing signals or engines having mechanical injectors. Besides developing a technique using vibration sensors, the task also investigates the minimum number of accelerometers needed to correctly detect a fuel injector misfire in any given cylinder. Like the previous detection approach, this investigation was conducted on an engine dynamometer test stand while seeding one injector fault electronically.

The data in Figure 5 illustrates a representative plot over two revolutions of the crankshaft at an operating condition of 30 lbs-ft @ 700 RPM. The alternating yellow and white regions correspond to crankshaft angle of revolution during which fuel injectors are injecting fuel into respective cylinders, designated in firing order sequence: #1-5-3-6-2-4.

The vertical axis overlays the injector profiles for each of the six cylinders and the signal acquired from the accelerometer mounted on the number three cylinder head. Note the clear relationship between the two injector pulses per revolution and the corresponding two pulses detected by the accelerometer.

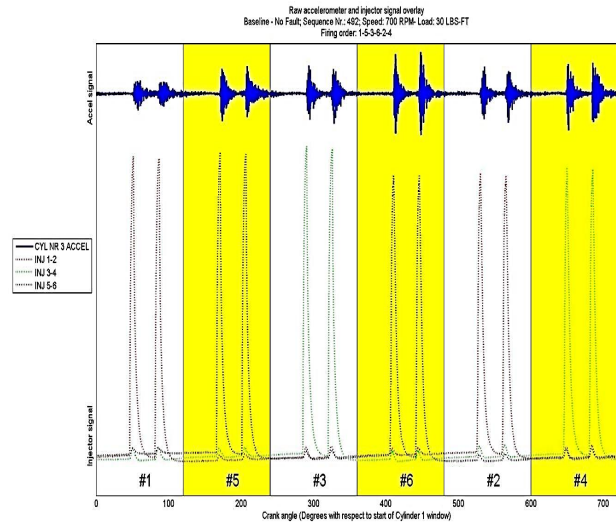


Figure 5. Accelerometer and Injector Signals: Baseline – No Fault (30 lbs-ft@700 RPM)

The data in Figure 6 shows a representative plot with an injector misfire event in cylinder one at the same operating conditions (30 lbs-ft @ 700 RPM). As a note, this snapshot of data was collected during the start of engine warm-up, hence the single injector profile.

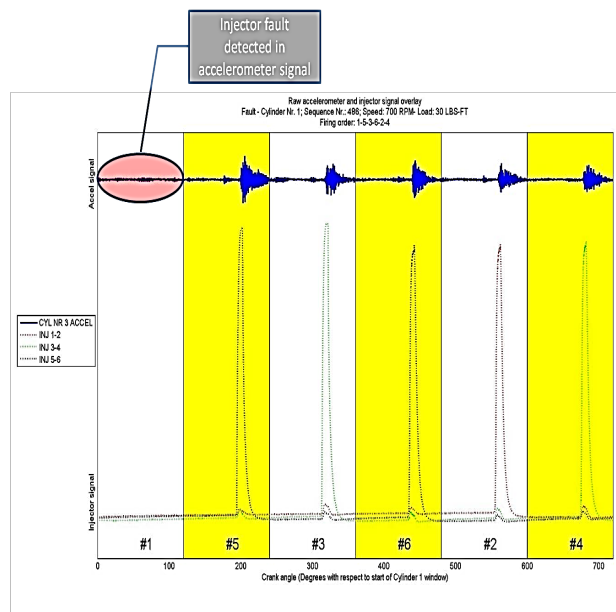


Figure 6. Accelerometer and Injector Signals: Fault Cylinder One (30 lbs-ft@700 RPM)

Here again, accelerometer three is the channel used to detect the vibration. This plot demonstrates that this centrally positioned accelerometer is capable of detecting injector misfire events at this operating condition on the dynamometer test stand.

The data in Figure 7 illustrates the signal obtained from accelerometer three at a relatively higher speed and load operating condition (560 lbs-ft @ 2200 RPM). The correlation between the injector pulses and vibration signal is not as apparent as it was for the 30 lbs-ft @ 700 RPM case but with simple low-pass filtering the firing event faults can still be detected.

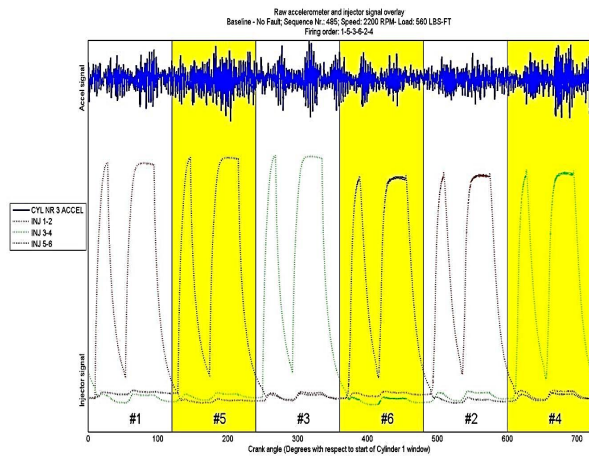


Figure 7. Accelerometer and Injector Signals: Baseline-No Fault (560 lbs-ft@2200 RPM)

Nonetheless, as illustrated in Figure 8, accelerometer three with low-pass filtering is capable of detecting a cylinder one injector misfire event. Accelerometer three demonstrated similar results in its ability to detect misfire faults in the other five cylinders.

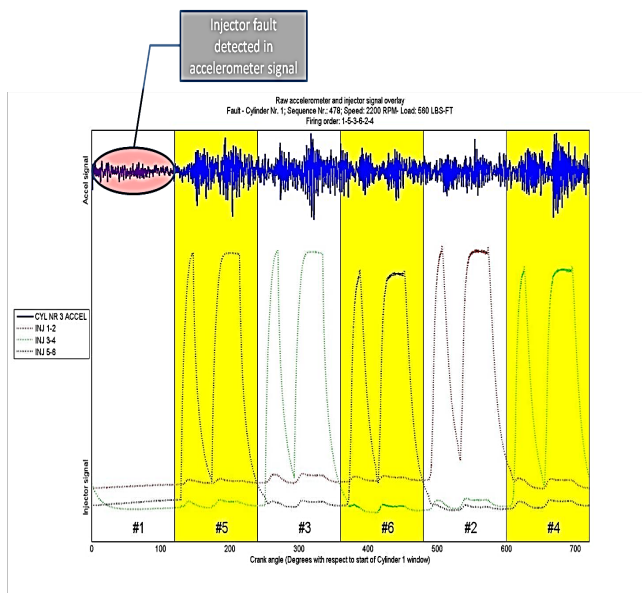


Figure 8. Accelerometer and Injector Signal Overlay: Fault Cylinder One (560 lbs-ft@ 2200 RPM)

The data in Figure 9 demonstrates that the signal obtained from accelerometer three is capable of detecting injector misfires across each of the six cylinders, respectively. As is shown in Figures 5 - 8, the horizontal axis in Figure 9 indicates crank angle revolution over two complete revolutions. The vertical axis indicates the signal obtained by accelerometer number three for six respective faulted cylinders in firing order, as read from top to bottom, (#1-5-3-6-2-4). Because the vertical axis is arranged in order of firing sequence with respect to crank angle, the highlighted green diagonal box creates an 'identity matrix'. Stated differently, using injector misfire three as an example, by visually tracing an imaginary vertical line beginning on the horizontal axis at approximately 300° up to the point where the line intersects the diagonal green box, the reader will observe the amplitude of the accelerometer signal is noticeably smaller relative to the preceding (left most) and subsequent (right most) portions of the signal. Tracing an imaginary horizontal line from this green box to the left until intersecting the vertical axis, the reader will observe the vertical axis number label corresponding to the cylinder that experienced the injector misfire.

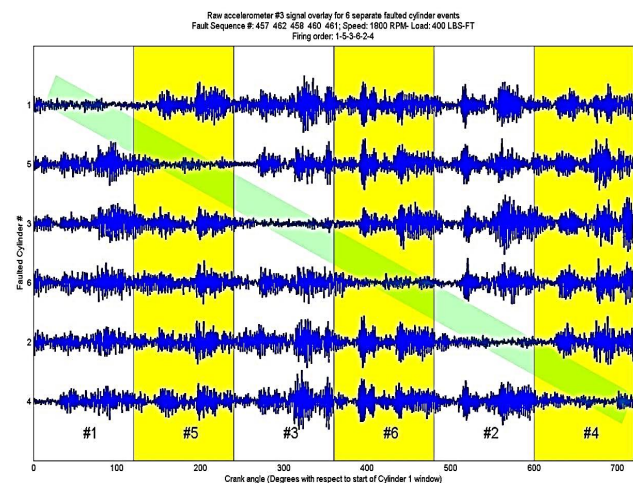


Figure 9. Accelerometer Three Detection of Injector Misfires Occurring in Each of the Six Cylinders: (560 lbs-ft@2200 RPM)

The vibration based detection approach resulted in a 100% detection rate for all the specific load, speed and fault conditions tested. The signal requirements for this approach are vibration data from one accelerometer located in an optimum location and a once per revolution signal (1PPR or TDC signal). This signal processing approach also works under varying conditions regardless of the number of injector pulses per cylinder.

As observed in the plots, the approach requires only one accelerometer to detect all cylinder faults. Having stated this, this approach would have to be validated on vehicles operating during extreme conditions to determine effects of

externally induced vibrations. If validated, this simple approach can also be easily embedded into an engine controller or small embedded processor based system. It is worth noting that different engines would each require their own respective signal amplitude threshold levels in order to trigger a fault detection alert. This paper demonstrates the feasibility of the approach. Amplitude thresholds were not part of this phase of the work. They will be established and validated for integration on an engine specific basis.

The next detection method discussed looks at using a commonly installed speed sensor for detecting faults by means of measuring changes in the crankshaft speed.

### 5. TIME DOMAIN SPEED ENCODER ANALYSIS

When an engine misfires, there will be a sudden decrease in engine speed immediately after the misfire event. Therefore, when a misfire event occurs, a torsional vibration is excited in the crankshaft and will show up in the instantaneous speed of the engine. The task of this analysis was to detect injector faults based on changes in crankshaft speed. For the purpose of this testing, the crankshaft was equipped with two speed encoders, a one pulse per revolution (1 PPR) and a 360 PPR encoder, respectively. The one PPR encoder provided an index signal for detecting each revolution of the crankshaft. The 360 PPR encoder enabled calculation of crankshaft angular velocity at a resolution of 1° of revolution.

In the hypothetical ideal scenario, crankshaft speed would be constant with no fluctuations for a given steady state operating condition. However in practice, each firing event/impulse creates a sinusoidal oscillation of crankshaft speed. The blue line in Figure 10 illustrates this speed oscillation. The green, magenta and red lines indicate the injector signals for the respective cylinders. Keeping in mind the x-axis represents 720° degrees (2 complete revolutions of the crankshaft), the alternating yellow and white regions corresponds to the firing order of the respective cylinders #1-5-3-6-2-4.

Two additional comments are pertinent to interpreting the following plots:

1. The speed plot is a composite mean of 50 two-revolution cycles of the crankshaft. Averaging over multiple cycles over a given steady state operating condition results in the averaging out of extraneous signal noise. The capability to synchronize multiple cycles and thereby enable multi-cycle averaging is accomplished by combining the one PPR encoder channel as an indexing signal with the injector one-two channel (green) to identify a consistent crank angle reference point. This reference point corresponds to the top dead center (TDC) position of cylinder one at the beginning of the cylinder one power stroke.

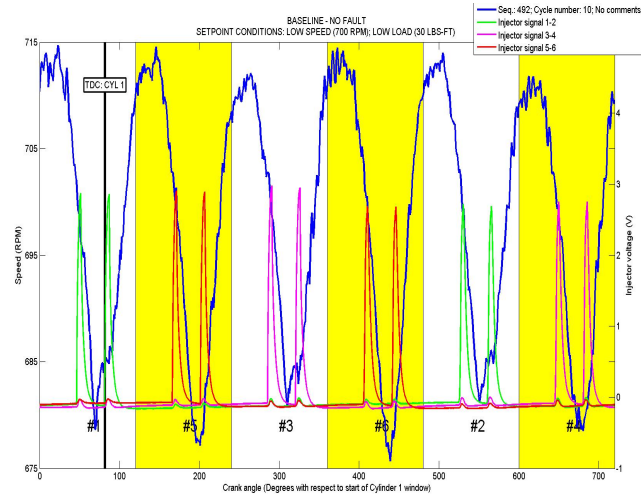


Figure 10. Sinusoidal Oscillation of Crankshaft Speed with All Six Injectors Operating

2. The left vertical axis corresponds to the speed plot (blue line), and the right vertical axis corresponds to the fuel injector signal voltage. Signal levels greater than zero correspond to the fuel injector injecting fuel into the cylinder.

The reason the injector signals are represented with green, magenta and red lines is due to the instrumentation configuration. There are three pairs of injector voltage sensors. The injectors are grouped sequentially (Green: injector one and two), (Magenta: injector three and four), (Red: injector five and six). Keep in mind, the following analysis is using the injector signals for validation purposes. The motivation for this analysis assumes the fuel injector signals would not be accessible as inputs for fielded-real world fault detection algorithm.

The data in Figure 11 shows the instantaneous speed plots over two revolutions of the crankshaft for a baseline (no injector fault) condition and an injector number three fault condition. At low speed and load conditions (30 lbs-ft @ 700 RPM), the speed oscillates in a relatively sinusoidal manner for both the no-fault and fault conditions. The abnormal characteristic of an increased range in RPM variation in the crankshaft speed for the faulted case (red line) observed primarily during cylinder two's firing order indicates the presence of an injector fault. Based on observations across multiple low speed and load setpoint conditions, the characteristic 'spike' in speed oscillation is typically observed one to two cylinders later in the firing order.

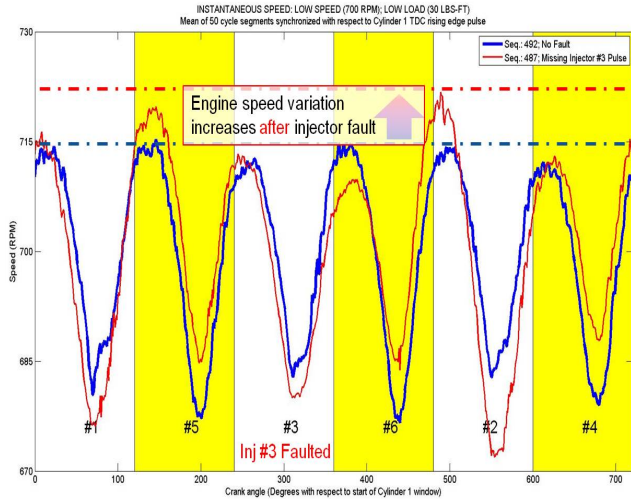


Figure 11. Low Speed-Low Load Speed Profile over Two Revolutions of the Crankshaft

However it was observed during our investigation that the potential to utilize this dynamic characteristic for fault detection purposes over the full range of engine operation is limited. This is due to the fact that the ‘spike’ is not as clearly or consistently evident at higher speed and load conditions. This is discussed and shown in the following figures.

The data in Figure 12 and Figure 13 illustrate two examples of the speed variation observed at higher speed and load conditions (400 lbs-ft @ 1450 RPM and 560 lbs-ft @ 2200 RPM respectively). In Figure 12 the ‘spike’ is not evident. It appears that the relative drop in speed during the crank angle region corresponding to cylinder six firing order may be indicative of the injector three misfire.

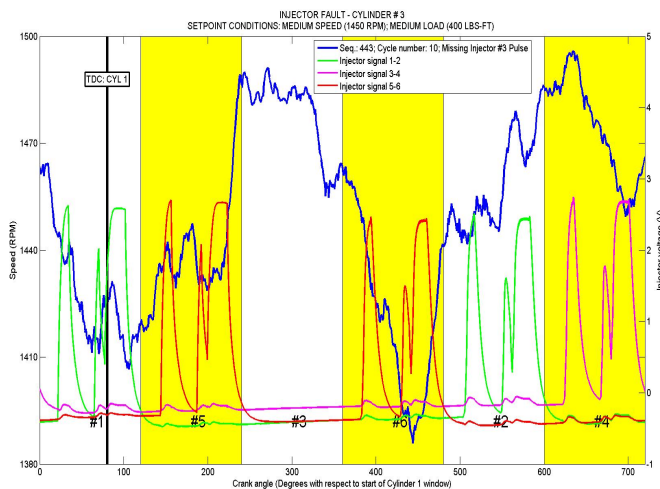


Figure 12. Speed Profile Over Two Revolutions of the Crankshaft (400 lbs-ft @ 1450 RPM)

In Figure 13, there is a ‘spike’ occurring during cylinder two’s firing order. However there is also a ‘spike’ occurring during cylinder one’s firing order. These inconsistent speed variations require additional pre-processing prior to implementing automated fault detection algorithms.

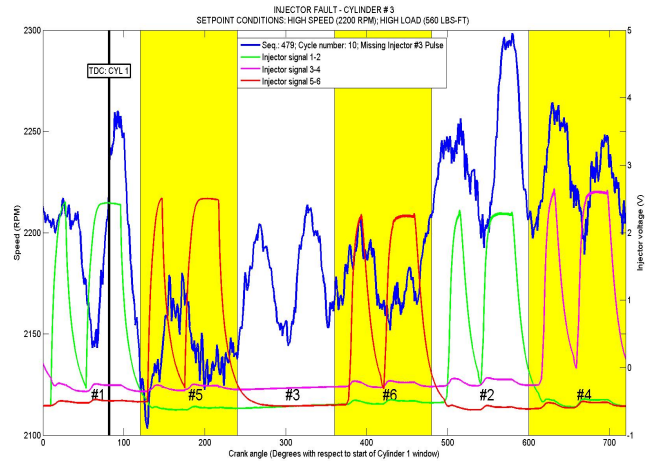


Figure 13. Speed Profile Over Two Revolutions of the Crankshaft (560 lbs-ft@2200 RPM)

In general, the correlation between fuel injector misfire and speed fluctuation is limited when using a visual inspection of the time domain speed plots based on instantaneous speed calculation of the shaft encoder. Distinguishing injector misfires at higher speeds (>700 RPM) from ‘noisy’ engine speed data is not clear-cut and will require more sophisticated processing techniques.

The instantaneous speed based detection approach is capable of detecting failure however it is computationally more complicated to implement. Visual detection of a given injector fault is apparent at different speed and load conditions. The automated detection techniques developed in the course of this investigation are effective at detecting injection related failures at low speeds and loads. At higher speeds, using instantaneous speed to distinguish specific injector faults is more difficult due to the speed fluctuations resulting from torsional vibration dynamics of the engine crankshaft. For these operating conditions, the pattern recognition algorithms would require more processing power than the other time-domain solutions presented in this paper.

For this investigation, this technique was not considered for further development due to the number of other practical methods which were found to be suitable for embedding into controllers. However, instantaneous speed measurements were found to be valuable using an order domain approach. This technique enables detection of faults using the crank encoder at high speeds and loads as well as

the ability to identify the specific cylinder with the injector fault.

The following sections will discuss the timer board data collection approach and the order domain fault detection processing technique in further detail.

## 6. ORDER DOMAIN SPEED ENCODER ANALYSIS

This section will discuss the method for collecting and processing order domain data for injector fault analysis. This investigation is based upon work by Geveci, Osburn, and Franchek (2005).

### 6.1. Data Collection Background

Throughout the test schedule, the engine's instantaneous crankshaft velocity was sampled using a Time Interval Measurement System (TIMS). While common A/D converters sample the amplitude of a sensor's output at discrete time intervals, a TIMS records the number of timer board counts between the output signal's threshold crossings. TIMS measurement systems are used in various areas in machinery health prediction systems as shown in Lebold, Maynard, Reichard (2005), Lebold Maynard (2003) and Maynard, Lebold, Groover, Trethewey (2000).

An AVL 365C encoder system was employed to produce an output signal representative of the crankshaft's instantaneous velocity. The encoder system's periodic output waveform is amplified to rails using threshold values to produce a TTL logic signal having hi/low amplitudes corresponding to the encoder wheel's optical pattern as illustrated in Figure 14.

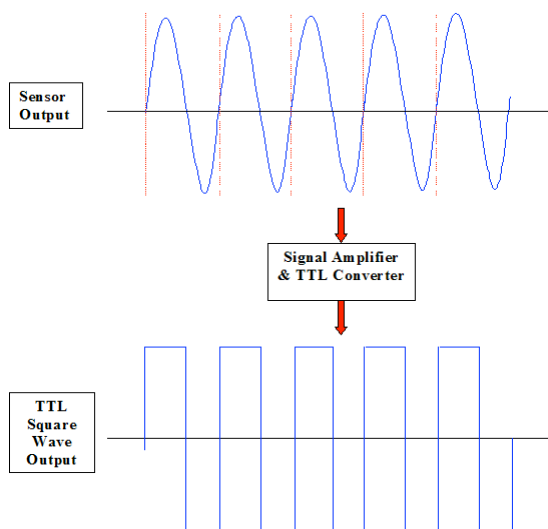


Figure 14. Conversion of sinusoidal sensor output to square wave pulses.

The TTL logic levels are then converted into a binary representation to accurately define the pattern-passing rate. The exact transition time from low-to-high and high-to-low of an optical pattern passing is important for processing. Therefore the fastest available analog-to-digital (A/D) converter would be preferable. However, these devices are very costly and are not necessary since the replication of the signal's temporal characteristics is not necessary. Only accurate capture of the transition times are necessary. A timer/counter board is an ideal candidate for this system because it provides a means of computing the transition time between an incoming logic signal with a high degree of resolution.

The National Instruments PCI-6602 Timer/Counter Board is a cost-effective solution and was used for data collection. With an 80MHz clock reference, pulse-width measurements from the PCI-6602 can be performed at a resolution of 0.0000000125 seconds. The data collection system is designed around the semi-buffered period measurement approach using direct memory access (DMA) for speed. In pulse-width measurement applications, the board counts the number of ticks (clock pulses of the time-base) that occur in every semi-period of the input waveform. It then stores this value in a data buffer. This results in an array of clock counts per semi-period of the encoder hub's pattern as illustrated in Figure 15. Knowing the reference clock frequency used during the measurement, the pulse-width, period or frequency of the input waveform can be computed. For this application, the series of pulse widths will be used to determine the engine's order content.

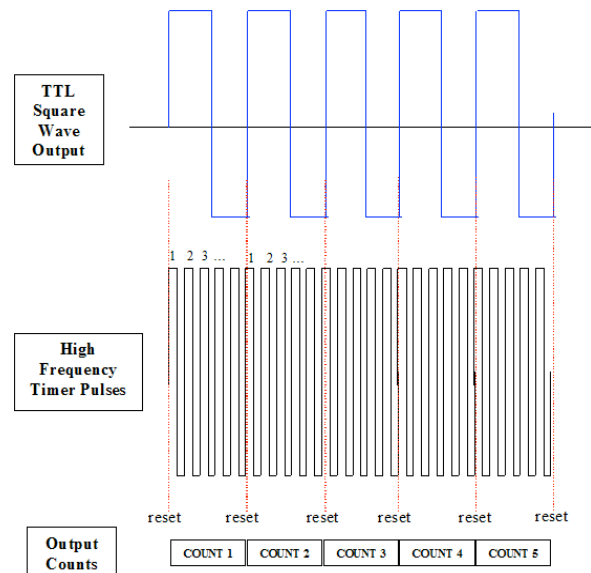


Figure 15. Zero crossing timer reset points.

### 6.2. Order Domain Data Analysis

Datasets from the TIMS provided a semi-period resolution of 720 PPR. Before processing, all datasets were normalized to begin at the TDC corresponding to the beginning of cylinder one's power stroke, hereafter referred to as 'true TDC'. In this way, the relative phase differences between particular cylinder firing events were consistent for all snapshots.

Encoder semi-periods in the datasets were combined to allow for a period-based analysis. Because encoder wheels typically have highs/lows of different widths, a period-based analysis eliminates the variance between semi-periods. Shown below, Figure 16 illustrates the time history of a semi-period signal (blue) and the constructed period based signal (black) that was used in the subsequent analysis. It is important to note that the mean of both signals was removed to illustrate a reduction in high-frequency signal content.

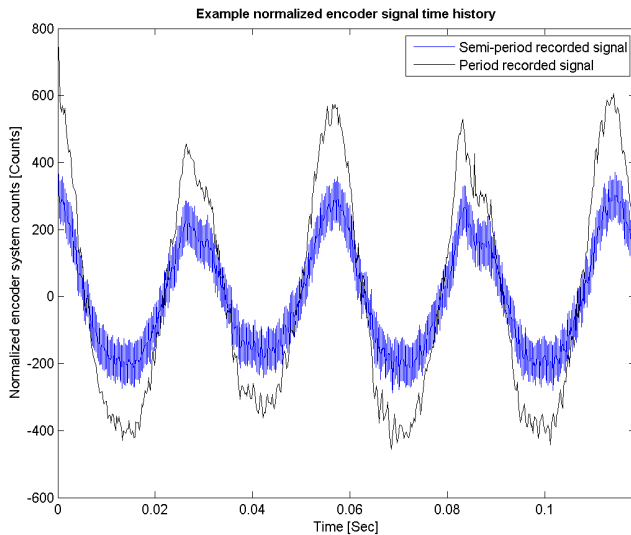


Figure 16. The Semi-Period Signal (Blue) Recorded in Snapshot 492 During Test #25 and the Constructed Period Based Signal (Black)

The period based signals were then demodulated to calculate the amplitude of the crankshaft's torsional vibration in degrees in the rotational domain. To reduce the effects of events uncorrelated to the shaft's rotation in the demodulated signal, synchronous averaging was also applied in the rotational domain. The amplitude and phases of the demodulated, averaged signal were then analyzed in the order domain.

Figure 17 through Figure 19 illustrate order spectra comparing data collected during baseline and injector fault tests at the discrete low, medium and high engine speeds, respectively.

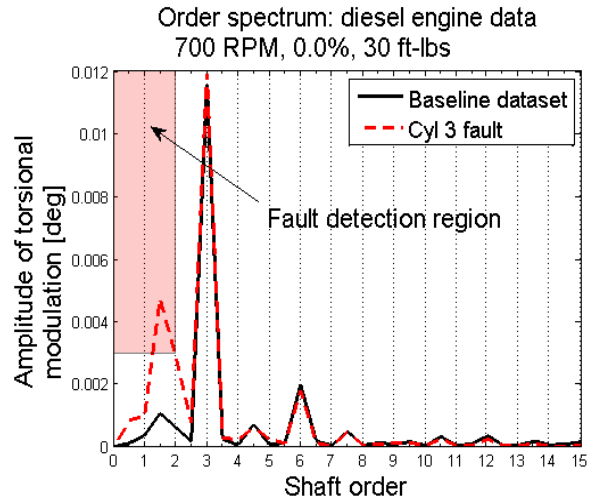


Figure 17. Baseline (Snapshot 492) and Cylinder Three Fault (Snapshot 487) Order Spectra for Low Speed, Low Load Engine Operating Conditions

An order domain analysis of the data collected during the injector fault test schedule suggests that cylinder faults can be identified by monitoring the amplitude of the crankshaft's orders lower than two. When an injector fault was introduced into the test system at low, medium, and high load/speed engine conditions, noticeable modulation of the crankshaft occurred between orders one-half and two.

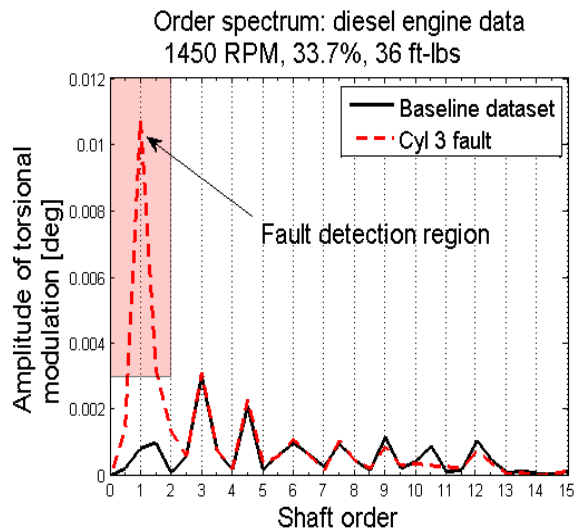


Figure 18. Baseline (Snapshot 446) and Cylinder Three Fault (Snapshot 443) Order Spectra for Medium Speed, Medium Load Engine Operating Conditions

Using these same datasets, an injector fault threshold amplitude for the orders between one-half and two was calculated. When the amplitude of any of the orders

between one-half and two rises above this threshold value and enters the “fault detection region”, a cylinder fault is diagnosed. The injector fault detection region above the empirically determined 0.003 degrees threshold is illustrated in Figure 17 through Figure 19.

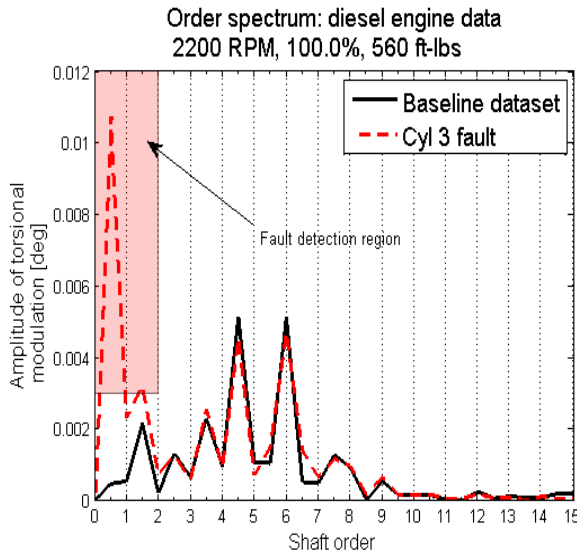


Figure 19. Baseline (Snapshot 485) and Cylinder Three Fault (Snapshot 479) Order Spectra for High Speed, High Load Engine Operating Conditions

The effect of an injector fault on the amplitude of the half order can be seen in Figure 17 through Figure 19.

Figure 19 testing results suggest that the magnitude of the torsional modulation caused by an injector fault is dependent on the engine’s operating conditions. At low speed, low load conditions the injector fault causes little torsional modulation to the shaft’s steady state rotation at the half order. At high speed, high load conditions, however, the torsional modulation at the one-half order is very pronounced.

After an injector fault is diagnosed, it is necessary to identify the cylinder that contains the injector fault. This, too, was accomplished in the order domain. A shaft order can be defined to describe the frequency of a particular synchronous event that occurs at each shaft revolution. From this definition, one can conclude that a shaft’s one half order contains information about events that occur once every two shaft revolutions. For a six cylinder, four-stroke engine this order specifically characterizes the frequency of a crankshaft’s torsional loads that result from an injector fault. These loads are caused by the energy loss due to lack of cylinder combustion, which in turn results in a momentary decrease in crankshaft velocity relative to the no-fault steady state crankshaft velocity profile.

To identify an injector fault, the phase of the half order was analyzed for each dataset. Because all datasets were normalized to begin with the first true TDC of the first cylinder in the engine’s firing sequence, the relative phase of cylinder firing events was maintained in this analysis. After the order spectrum was calculated for each dataset, the phase of the one-half order was extracted. Figure 20 illustrates the phase of each dataset’s half order as injector signals to various cylinders were deliberately attenuated. The phase clustering of the one-half order provides a means to identify a particular faulty injector signal.

An analysis of Figure 20 reveals that the one-half order phases are clustered in groups that are separated by roughly 120°. The physical significance of this separation angle can be traced to the crank rotation angle between cylinder firing events that is 120° for a four-stroke, six cylinder engine.

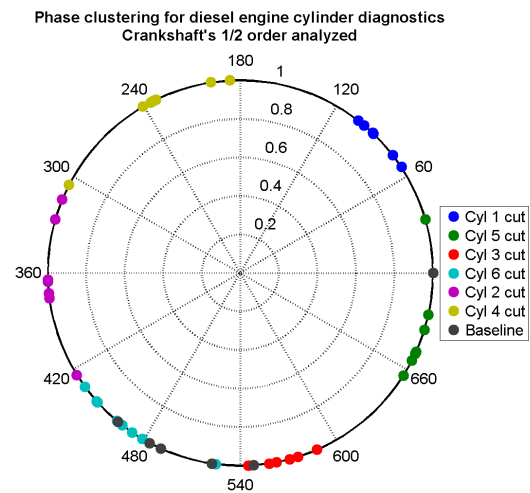


Figure 20. Phase Clustering of One-Half Order

Order domain inspection of data collected by the crankshaft’s existing encoder system provides valuable information that can be used for injector fault diagnostics. By monitoring the amplitude of the crankshaft’s low order content relative to the suggested fault detection window, injector health can be classified. If a faulty injector signal is diagnosed, the phase of the one-half order can be compared to the clustering in Figure 20 to determine the cylinder that is most likely faulted.

## 7. CONCLUSION

This paper described four approaches for detecting fuel injector misfire events; an injector signal based approach, a vibration signal approach, a crankshaft speed based approach, and an order domain approach. Relative to the crankshaft speed based approach, the injector signal and vibration time domain approach both showed the greatest



potential for on-board vehicle implementation. These approaches demonstrated the capability of detecting individual injector misfires on the seven liter diesel engine over the (30 lbs-ft @ 700 RPM) to (560 lbs-ft @ 2100 RPM) operating range when tested on an engine dynamometer. Furthermore, each approach equally demonstrated the ability to correctly identify the specific cylinder location of the faulted injector.

Both approaches require a TDC/one PPR or 360PPR encoder signal for synchronization or the camshaft speed encoder signal, along with the basic hardware and software for algorithm implementation.

The injector signal based approach does require access to the signal taps for the six respective injectors while the vibration based approach requires one accelerometer mounted to the top of the engine. Due to the nature of vibration monitoring, this approach should also work for detecting any combustion related faults but must be first validated on an actual platform to ensure it is immune to externally induced vibrations. Among other factors, the extent to which fuel line clogs or worn injector actuators are relevant faults may dictate the reliability of this approach in a fielded environment.

The order domain approach showed great potential for on-board vehicle implementation. The approach demonstrated the capability of detecting individual injector misfires on the seven liter diesel engine over the (30 lbs-ft @ 700 RPM) to (560 lbs-ft @ 2100 RPM) operating range when tested on an engine dynamometer. Additionally, this approach demonstrated the ability to correctly identify the specific cylinder location of the faulted injector.

#### ACKNOWLEDGMENT

This paper is based upon work supported by the Army RDECOM-TARDEC under Contract Number N00024-02-D-6604, Delivery Order Number 0707. The content of the information does not necessarily reflect the position or policy of the Government, and no official endorsement should be inferred.

#### REFERENCES

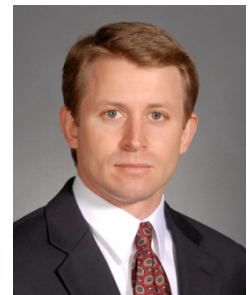
- Banks, J.C., Hines, J., Lebold, M.S., Campbell, R.L., et al. (2001). Failure Modes and Predictive Diagnostics Considerations for Diesel Engines. *Meeting of the Society for Machinery Failure Prevention Technology*, April 2001.
- Geveci, M., Osburn, A. and Franchek, A. (2005). An investigation of crankshaft oscillations for cylinder health diagnostics. *Mechanical Systems and Signal Processing*, vol 19, pp. 1107-1134.
- Lebold M.S., Maynard K.P., et al. (2003). Technology Development For Shaft Crack Detection in Rotating Equipment. *EPRI International Maintenance Conference*, August 2003.
- Lebold, M.S., Maynard K.P., Reichard, K.M., et al., (2005). A Non-Intrusive Technique For On-Line Shaft Crack Detection and Tracking. *Proceedings of IEEE Aerospace Conference*, March 2005.
- Maynard, K.P, Lebold, M.S., Groover, C.L., and Trethewey, M.W., (2000). Application of Double Resampling to Shaft Torsional Vibration Measurement for the Detection of Blade Natural Frequencies. *Society for Machinery Failure Prevention Technology*, March, 2000.
- Murphy, B.J., Lebold, M.S., Reichard, K.M., et al. (2003 ). Diagnostic Fault Detection for Internal Combustion Engines Via Pressure Curve Reconstruction. *Proceedings of IEEE Aerospace Conference*, March 2003.

#### BIOGRAPHIES

**Mitchell S. Lebold** is a Research Engineer with the Applied Research Laboratory at The Pennsylvania State University with more than fifteen years of experience in algorithm development and embedded monitoring systems. He holds an M.S. and B.S. degree in Electrical Engineering from Pennsylvania State University. Mr. Lebold leads several multi-disciplinary research and development projects in the areas of wireless smart sensors, algorithm development, and open standards for condition-based maintenance. He has designed and developed numerous custom test and measurement systems as well as embedded wireless data acquisition systems for machinery health monitoring. His current research involves engine and driveline diagnostics and prognostics algorithm development.



**James S. Pflumm** is a research engineer at the Applied Research Laboratory. He received his B.S.M.E and M.S.M.E in Mechanical Engineering from Penn State University in 2002 and 2005 respectively. His current work at ARL is in support of test and analysis of failure modes for ground vehicles at the sub-system and component level.



**Jeffrey C. Banks** is the Department Head of Complex Systems Engineering & Monitoring and he has 17+ years' experience in applying advanced signal processing techniques, intelligent systems technology, and embedded diagnostics / prognostics tools to condition monitoring applications for the US Navy, US Marine Corps, US Army, NASA and Industry. His education includes a B.S.M.E. from Villanova University and a M.S. in Acoustics from The Pennsylvania State University.

His research engineer duties include developing machinery health management systems and diagnostic and prognostic technology for various DoD assets including the U.S. Marine Corps Expeditionary Fighting Vehicle (EFV), Light Armored Vehicle (LAV), AV-8B Harrier, U.S. Army Heavy Expanded Mobility Tactical Truck (HEMTT), and Heavy Brigade Combat Team M1A2 Abrams, M2/M3 Bradley Fighting Vehicle, M88 Hercules platforms. He has developed two short courses at all NASA facilities in the areas of Condition Based Maintenance (CBM) and Reliability Centered Maintenance (RCM). Additional responsibilities include conducting degrader analyses for a variety of complex systems and platforms including aircraft engines and combat ground vehicles. He has also designed and developed diagnostic instrumentation systems for machinery failure test beds and field data acquisition systems. Mr. Banks has first authored and published more than 50 conference papers and technical reports.

**Jonathan P. Bednar** is a Graduate Research Assistant with the Applied Research Laboratory at The Pennsylvania State University. He earned his B.S. degree in Mechanical Engineering in 2010 from The Pennsylvania State University and will complete his M.S. degree in 2012 focusing in acoustics and vibrations.



His current research includes condition monitoring of Compression Ignition engine crankshafts using torsional features and the optimization of a high-temperature pressure sensing device.

**Dr. Karl M. Reichard** is the head of the ARL Penn State Embedded Hardware/Software Systems and Applications Department and teaches and advises graduate students in The Pennsylvania State University Graduate Program in Acoustics, and the Departments of Mechanical and Electrical Engineering. He has over



25 years of experience in the design and implementation of signal processing, control and embedded diagnostics/prognostics systems. He has developed unattended remote sensing, active control and health

monitoring systems for a variety of manned and unmanned platforms for the Navy, Army, Marine Corp, NASA, and industry. In addition to his work at Penn State ARL, Dr. Reichard has also worked for the US Army at Aberdeen Proving Grounds where he developed data measurement, acquisition, and processing systems to support testing of military systems. He earned B.S., M.S., and Ph.D. degrees in Electrical Engineering from Virginia Tech.

**Kenneth Fischer** has led the Condition Based Maintenance (CBM) Team within the U.S. Army Tank Automotive Research, Development, and Engineering Center (TARDEC), located in Warren, Michigan since 2009. In this capacity, he has overseen the design, development, integration and test of the diagnostic hardware and software components required to implement CBM on military ground vehicles. Mr. Fischer holds a MS Degree in Engineering Management from University of Dayton and a BS in Computer Engineering from Lawrence Technological University and has over 15 years of automotive experience in embedded diagnostic and test system design.

**Joseph Stempnik** is an electric engineer at the Tank Automotive Research Engineering Center (TARDEC). He received his B.S. in Electrical Engineering from Lawrence Technological University. His current work at TARDEC is in support of test and evaluation of engines that support Army ground vehicles and has over 15 years of automotive experience in embedded controls and software system design.

# Constructing an Efficient Self-Tuning Aircraft Engine Model for Control and Health Management Applications

Jeffrey B. Armstrong<sup>1</sup> and Donald L. Simon<sup>2</sup>

<sup>1</sup>*Vantage Partners, LLC, Cleveland, Ohio, 44135, U. S. A.  
jeffrey.b.armstrong@nasa.gov*

<sup>2</sup>*NASA Glenn Research Center, Cleveland, Ohio, 44135, U. S. A.  
donald.l.simon@nasa.gov*

## ABSTRACT

Self-tuning aircraft engine models can be applied for control and health management applications. The self-tuning feature of these models minimizes the mismatch between any given engine and the underlying engineering model describing an engine family. This paper provides details of the construction of a self-tuning engine model centered on a piecewise linear Kalman filter design. Starting from a nonlinear transient aerothermal model, a piecewise linear representation is first extracted. The linearization procedure creates a database of trim vectors and state-space matrices that are subsequently scheduled for interpolation based on engine operating point. A series of steady-state Kalman gains can next be constructed from a reduced-order form of the piecewise linear model. Reduction of the piecewise linear model to an observable dimension with respect to available sensed engine measurements can be achieved using either a subset or an optimal linear combination of “health” parameters, which describe engine performance. The resulting piecewise linear Kalman filter is then implemented for faster-than-real-time processing of sensed engine measurements, generating outputs appropriate for trending engine performance, estimating both measured and unmeasured parameters for control purposes, and performing on-board gas-path fault diagnostics. Computational efficiency is achieved by designing multidimensional interpolation algorithms that exploit the shared scheduling of multiple trim vectors and system matrices. An example application illustrates the accuracy of a self-tuning piecewise linear Kalman filter model when applied to a nonlinear turbofan engine simulation. Additional discussions focus on the issue of transient response accuracy and the advantages of a piecewise linear

Kalman filter in the context of validation and verification. The techniques described provide a framework for constructing efficient self-tuning aircraft engine models from complex nonlinear simulations.

## 1. INTRODUCTION

An emerging technology in the field of aircraft engine controls and health management is the inclusion of real-time on-board self-tuning models for the in-flight estimation of engine performance variations (Luppold, Roman, Gallops, & Kerr, 1989). Self-tuning engine models are comprised of an engine model and an associated tracking filter as shown in Figure 1. Here, the Aircraft Engine block also includes an on-board engine control computer (not shown). Real-time sensor and actuator information available within this control computer is used as inputs to the engine model. The engine model reflects engine aero-thermodynamic performance at both steady-state and transient conditions. Analytical modeling approaches, such as piecewise linear models and nonlinear component level models, are commonly applied. Recently, Volponi (2008) also demonstrated the merits of a hybrid modeling approach combining analytical (physics-based) and empirical (neural network) elements. The tracking filter, typically based on Kalman filter estimation concepts, is designed to automatically adjust tuning parameters within the engine model to enable the model to match the observed performance of the physical engine. The discrepancy between the physical engine parameters and the model variables is referred to as model mismatch. The tuning parameter adjustment is necessary to enable the model to account for the gas turbine engine performance variations caused by deterioration, wear, and fouling that turbomachinery will incur over time with usage.

Jeffrey B. Armstrong et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

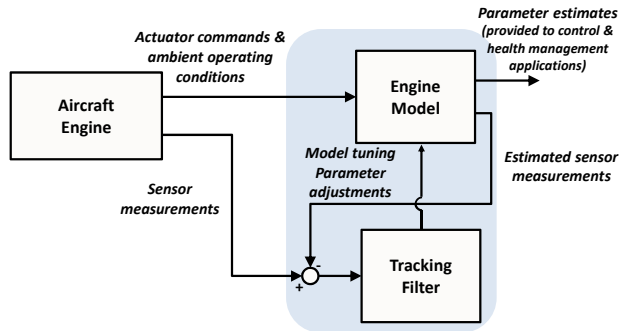


Figure 1. Self-tuning engine model architecture including an aircraft engine, an engine model, and a tracking filter to tune the model

One of the earliest investigations to consider the use of self-tuning engine model technology was under the NASA and Department of Defense led Performance-Seeking Control (PSC) program of the 1980's and 1990's (Shaw, Foxgrover, Berg, Swan, Adibhatla, & Skira, 1986), (Nobbs, Jacobs, & Donahue, 1992). The PSC program demonstrated multiple performance benefits achievable through in-flight propulsion control optimization including reduced fuel-burn, increased thrust, and increased component life (Gilyard & Orme, 1993). Central to the PSC design is the inclusion of a self-tuning engine model that provides the control system with real-time estimates of unmeasured engine performance parameters. Follow-on research efforts have continued to mature and advance aircraft engine model-based control technology (Dwyer, 1990), (Klaus & Kreiner, 2001), (Brunell, Bitmead, & Connolly, 2002).

In addition to control applications, self-tuning engine model technology also holds benefits for propulsion system health management. This includes estimating, trending, and forecasting the level of performance deterioration within the major rotating modules of the engine, and diagnosing faults that impact engine gas path performance (Gallops, Gass, & Kennedy, 1992), (Bushman & Gallops, 1992), (Armstrong & Simon, 2011). It is expected that the application of on-board self-tuning engine model technology within the aircraft engine industry will continue to increase. A future vision put forth by Behbahani, Adibhatla, and Rauche (2009) is to develop an integrated on-board self-tuning model-based engine controller architecture with control, diagnostic, and prognostic functionalities.

This paper is intended to serve as a guide for the development and implementation of self-tuning engine models for turbomachinery diagnostics, prognostics, and health management. It will cover the required information, design considerations, and design steps necessary to construct and implement such models. The focus of this work is specifically on piecewise linear self-tuning engine models with a Kalman filter as the tracking filter.

The remainder of this paper guides the reader through the design and implementation details of a self-tuning engine model. First, the overall design is discussed briefly to introduce the various requirements and components of such a model. Next, the generation of linear state space engine models and the design and setup of the Kalman filter is explained. Then, the implementation is explicated to aid the reader in constructing an efficient adaptation of this design. This is followed by the presentation of example results from the application of the technique to a turbofan engine simulation. Finally, a discussion of some design choices and model advantages is presented.

## 2. MODEL COMPONENTS

The self-tuning engine model outlined in this paper is composed of a number of modular components. Two major components comprise this design: a piecewise linear state-space engine model and an associated piecewise linear steady-state Kalman filter.

The input signals available to a self-tuning engine model are restricted to the following signals, which are also provided to the engine controller: engine actuator commands and sensor measurements from the engine. Additional inputs may consist of airframe sensed measurements, such as free stream conditions, that affect gas-path analyses.

The outputs of the self-tuning engine model will normally consist of three components: sensor estimates, tuning parameters, and unmeasured parameter estimates. In this architecture, the sensor estimates should be nearly identical to the actual sensed measurements since the model is "tuned" to match these measurements. Tuning parameters generated by the model are available as outputs. The parameters selected and applied as model tuning parameters depend on design decisions, but they often act as proxies for engine performance parameters, as is discussed later. Finally, the self-tuning engine model can produce estimates of unmeasured engine parameters. Since the engine model is physics-based, the estimates of unmeasured parameters are expected to approach the actual values. The estimates often include thrust, stall margins, and gas-path pressures and temperatures where physical measurement is impractical.

A piecewise linear state-space model forms the basis for modeling theoretical engine dynamics. This model is comprised of state space matrices and associated trim points, both of which are interpolated based on operating point. The piecewise linear model is created from a nonlinear aerothermal model, but offers some advantages over the more complex physics-based simulation. A piecewise linear model is usually less computationally intensive than its nonlinear equivalent, and the simpler structure allows for straightforward design of tuning solutions, a Kalman filter in this case.

Accompanying the piecewise linear model in this approach is a piecewise steady-state Kalman filter. This implementation uses gain matrices that are pre-computed and held constant as opposed to updating the gain matrices online while processing the input data. Similar to the state-space matrices, the Kalman gain matrices are also interpolated based on operating point. The Kalman filter produces tuning parameters that drive the difference between the actual sensor measurements and their respective estimates to zero.

When employing piecewise solutions, design considerations related to interpolation mechanisms, parameter scaling, degrees of freedom, and processing time are important. For accuracy reasons, it is often advantageous to consider interpolation based on multiple dimensions, as will be described. Efficient algorithms should be employed to exploit multidimensional data alignment and shared scheduling. Additionally, correction techniques applied internally within the model, as discussed later, require some supporting code infrastructure.

### 3. MODEL DATA COMPUTATION

Constructing this self-tuning model requires translating a nonlinear aerothermal model to an equivalent piecewise linear model and piecewise linear Kalman filter. The initial step is to linearize the nonlinear model at multiple operating points to generate a piecewise linear state-space model and an associated set of trim points. Once linearization is complete, the Kalman gain matrices are computed. The resultant trim points, state-space system matrices, and Kalman gains can then be used in the self-tuning engine model implementation.

#### 3.1. Linearization

The nonlinear model of an aircraft engine can be represented by the following equations

$$\begin{aligned} \dot{x} &= f(x, u, h) \\ y &= g(x, u, h) \\ z &= g_z(x, u, h) \end{aligned} \quad (1)$$

where  $x$  and  $u$  represent the vectors of engine state variables and control command inputs, respectively. The vector  $h$  represents health parameters, such as efficiency or flow capacity, reflective of performance deterioration within the major modules of the engine. For given input values, the nonlinear functions  $f$ ,  $g$ , and  $g_z$  generate the vectors of state derivatives  $\dot{x}$ , sensed engine outputs  $y$ , and unmeasured engine outputs  $z$ , respectively. By linearizing the engine model at a given operating point, the following state-space equations are obtained:

$$\begin{aligned} \dot{x} &= A \underbrace{(x - x_{trim})}_{\Delta x} + B \underbrace{(u - u_{trim})}_{\Delta u} + L \underbrace{(h - h_{ref})}_{\Delta h} \\ \dot{x} &= A\Delta x + B\Delta u + L\Delta h \\ \underbrace{y - y_{trim}}_{\Delta y} &= C \underbrace{(x - x_{trim})}_{\Delta x} + D \underbrace{(u - u_{trim})}_{\Delta u} + M \underbrace{(h - h_{ref})}_{\Delta h} \\ \Delta y &= C\Delta x + D\Delta u + M\Delta h \\ \underbrace{z - z_{trim}}_{\Delta z} &= F \underbrace{(x - x_{trim})}_{\Delta x} + G \underbrace{(u - u_{trim})}_{\Delta u} + N \underbrace{(h - h_{ref})}_{\Delta h} \\ \Delta z &= F\Delta x + G\Delta u + N\Delta h \end{aligned} \quad (2)$$

Here,  $A$ ,  $B$ ,  $C$ ,  $D$ ,  $F$ ,  $G$ ,  $L$ ,  $M$ , and  $N$  are the state-space matrices reflecting system dynamics. The trim vectors, denoted by the subscript “trim,” reflect the values of the state variables, commands, and measured and unmeasured outputs when the model is at steady-state (i.e.,  $\dot{x} = 0$ ) at the given operating point. Collectively, the trim vectors define what is referred to as a “trim point.” The vector  $h_{ref}$  represents a reference health condition specified by the system designer. In Equation 2, parameter deviations relative to trim or reference conditions are denoted by the delta symbol ( $\Delta$ ).

The initial step in creating this self-tuning engine model is the computation of linear state-space models from the nonlinear model at multiple operating points. These operating points serve as the interpolation scheduling parameters in the piecewise linear model. Figure 2 shows a notional three-dimensional example of operating point specification using altitude, Mach number, and power setting as the scheduling parameters. The number of operating points and spacing between operating points, which does not have to be uniform, are design decisions left to the end user. In general, a denser grid of operating points will allow the piecewise linear model to more closely approximate the nonlinear model. However, that will increase memory storage requirements required for implementation. While the operating points generally reside within a standard flight envelope, it may be advantageous to select some operating points beyond the standard envelope (assuming the nonlinear engine model is operable and valid at these points). This expanded operating envelope functionality is necessary to enable the piecewise linear model to account for scenarios where the actual aircraft engine operates beyond normal expected operating conditions.

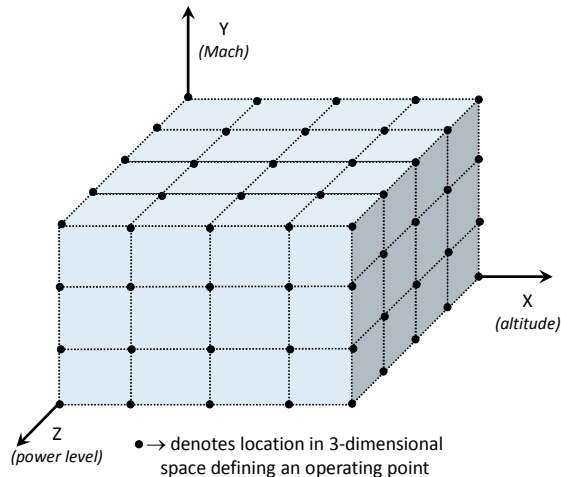


Figure 2. Example of three-dimensional piecewise linear model operating point scheduling

As described above, the piecewise linear model will use two related datasets: the trim points for the piecewise linear model and the state-space system matrices that model engine dynamics. Scheduling of the model and the generation of these datasets are discussed below.

### 3.1.1. Selection of Operating Points and Scheduling

The authors suggest selecting engine power level and flight conditions as scheduling parameters to enhance simulation accuracy. The rotational speed of the engine, specifically the fan speed for a two-spool turbofan, is considered a suitable proxy for power level. Alternatively, engine command parameters, such as power lever angle, may be used. However, since these variables do not necessarily reflect actual engine conditions, their usage may introduce inaccuracies during interpolation of trim points.

When scheduling on flight condition, Mach number and pressure altitude are suggested. While neither value is directly measurable, both can be easily computed from free stream and inlet pressure and temperature measurements. The use of Mach number and altitude will hide the nonlinearities that may be present in pressure and temperature changes over the flight envelope, allowing for more uniform steps in scheduling parameters.

The number of dimensions chosen for scheduling is a tradeoff between computational complexity and accuracy. The added accuracy that higher dimensional scheduling provides comes with both data storage and computational requirement penalties; significantly more mathematical operations are required as interpolation dimensions increase. As is discussed later, parameter correction may be used to minimize altitude effects. One may choose, therefore, to eliminate altitude as a scheduling parameter. However, parameter correction is imperfect, and there may be a resultant loss in accuracy.

Because two discrete data sets are generated (i.e., trim points and state-space system matrices) the interpolation scheduling may be separated if desired to decrease data storage requirements. Trim points require nearly an order of magnitude less storage space per operating point compared to state-space matrices. Furthermore, the dynamics of the system are not expected to change drastically; scheduling of the state-space matrices may not require the same grid density as the steady-state trim points. It may be advantageous to generate a denser or higher dimensional data set of trim points, while keeping the dynamics data set sparser or of a lower scheduling dimension (Brotherton, Volponi, Luppold, & Simon, 2003).

### 3.1.2. Trim Points Calculation

Generation of trim point data can be achieved using a steady-state engine model, as the trim points represent only the engine inputs and outputs without dynamics. After choosing operating conditions at which to compute trim points, the data can be generated in a hierarchical manner. For example, to generate the trim points in the three recommended dimensions discussed above, one would implement the algorithm below:

*for each X in selected altitudes:*

*for each Y in selected Mach numbers:*

*for each Z in selected power levels:*

*Compute engine inputs, sensed outputs, and unmeasured outputs*

*Append trim point data sets*

The trim points generally should be computed around the traditional flight envelope of the engine. Many models should be capable of computing conditions beyond the typical operating limits of the engine. Having data in regions beyond the expected operating envelope will help to protect against unforeseen flight conditions. Furthermore, the algorithm described will inherently generate unrealistic conditions. If, for example, the Mach numbers of interest vary from 0 to 0.8 and the altitude varies from sea level to a high cruise point, the algorithm will attempt to compute a condition of 0.8 Mach at sea level, which is likely an unrealistic condition. If generating the trim points from a physics-based model, this condition can most likely be computed with reasonable accuracy as altitude and Mach number translate into free stream and engine inlet conditions.

In situations where the algorithm cannot compute the desired steady-state condition, the operating point should be logged as a failure. After attempting to compute all conditions using the outlined algorithm, the failed conditions can be interpolated from those that were successfully calculated. It is suggested that the interpolation of failed conditions be performed linearly from nearest

successful computations along the power level axis. Extrapolation can be performed when a failed steady-state condition is not bounded by successful calculations. Extrapolated parameters, however, should not be relied on heavily for accuracy.

If the self-tuning model is to be applied to a variety of actual engines, it may be advantageous to carefully choose an engine performance level at which to compute the trim points. A fleet of engines will exhibit a statistical distribution of degradation of the rotating modules, which is manifested in changes to each module's efficiency and flow capacity (Sallee, 1978). The self-tuning engine model would benefit from being designed at a mean or median degradation condition with regards to the specific fleet to minimize the possible difference in performance variations across all engines.

### 3.1.3. Dynamics Calculation

Each of the matrices previously presented in Equation 2 must be calculated by perturbing their respective driving parameters via the nonlinear model. In this self-tuning implementation, the dynamics are captured by perturbing a given parameter within a modified version of the steady-state model that balances at a point described with non-zero state derivative conditions. When the perturbation is applied, the modified steady-state solver will attempt to calculate a balanced engine state, where the state variables are held constant, and the state derivatives are permitted to assume non-zero values. This condition is the instantaneous dynamic response of the engine after a perturbation is applied. The difference between the perturbed and unperturbed engine model outputs and state derivatives represent the dynamic behavior of the system for each given perturbation.

As stated earlier, it may be advantageous to compute the state-space matrices at different operating conditions than the trim points to address storage space concerns. If we again assume that the recommended three dimensions are used for scheduling state-space matrix interpolation, the algorithm will be:

*for each X in selected altitudes:*

*for each Y in selected Mach numbers:*

*for each Z in selected power levels:*

*Compute state-space matrices*

*Append matrix data sets*

The procedure for computing the matrices via perturbation of the inputs flows as follows:

*Compute steady-state  $x_{trim}$ ,  $y_{trim}$ ,  $z_{trim}$ , and  $u_{trim}$  for current operating point*

*for each state in x:*

*Compute  $x$ ,  $y$ ,  $z$  for  $(x_i + \delta x_i)$ , where  $\delta x_i$  is the perturbation size for the  $i^{th}$  state, as  $x_p$ ,  $y_p$ ,  $z_p$  using "unbalanced" steady-state model*

*Compute state derivatives from the "unbalanced" state*

*Set column  $i$  of  $A$  to be  $\dot{x}_p / \delta x_i$*

*Set column  $i$  of  $C$  and  $F$  to be  $(y_p - y_{trim}) / \delta x_i$ ,  $(z_p - z_{trim}) / \delta x_i$  respectively*

*for each input in u:*

*Compute  $x$ ,  $y$ ,  $z$  for  $(u_i + \delta u_i)$ , where  $\delta u_i$  is the perturbation size for the  $i^{th}$  actuator, as  $x_p$ ,  $y_p$ ,  $z_p$  using "unbalanced" steady-state model*

*Compute state derivatives from the "unbalanced" state*

*Set column  $i$  of  $B$  to be  $\dot{x}_p / \delta u_i$*

*Set column  $i$  of  $D$  and  $G$  to be  $(y_p - y_{trim}) / \delta u_i$ ,  $(z_p - z_{trim}) / \delta u_i$ , respectively*

*for each health parameter in h:*

*Compute  $x$ ,  $y$ ,  $z$  for  $(h_i + \delta h_i)$ , where  $\delta h_i$  is the perturbation size for the  $i^{th}$  health parameter, as  $x_p$ ,  $y_p$ ,  $z_p$  using "unbalanced" steady-state model*

*Compute state derivatives from the "unbalanced" state*

*Set column  $i$  of  $L$  to be  $\dot{x}_p / \delta h_i$*

*Set column  $i$  of  $M$  and  $N$  to be  $(y_p - y_{trim}) / \delta h_i$ ,  $(z_p - z_{trim}) / \delta h_i$  respectively*

The health parameters are set nominally to the desired design point that describes a mean or median engine deterioration level. Note that the procedure above generates state-space systems in continuous form. Conversion to discrete-time state-space equivalents can be performed using an appropriate technique such as zero-order hold after generating the continuous time matrices.

The scale of the applied perturbations depends on the variables to be perturbed. The authors suggest using perturbations at least an order of magnitude less than a particular variable's steady-state value. Some trial and error experimentation is necessary with each parameter to determine an acceptable perturbation scale.

The algorithm outlined suggests that the small changes are performed in a single direction. However, improved accuracy might be gained by applying perturbations in two directions and computing an average effect from these two perturbations. Moreover, some perturbations in a given direction may not be possible if, for example, they exceed the capabilities of a given engine actuator. The implementer should take care to consider these special cases when performing linearization.

Similar to the trim point calculation procedure, one may encounter problematic operating conditions due to the algorithm's simplicity with respect to cycling through desired points. However, unlike the solution for failed convergence of trim points during calculations, interpolation and extrapolation is not recommended when dealing with matrices that cannot be reliably computed. Extrapolation can quickly lead to unrealistic dynamic behavior if great care is not taken, and the system dynamics are not expected to vary drastically across neighboring operating points. Instead, it is suggested that a nearby (in terms of scheduling parameters) successfully computed state-space matrix set should be used for the given failed calculation point.

### 3.2. Kalman Filter Design

The piecewise linear Kalman filter is the core of this self-tuning engine model. For each state-space system of the piecewise linear engine model, a corresponding Kalman gain matrix must also be computed. In this implementation, steady-state Kalman filtering is applied. This means that the Kalman gain matrix corresponding to each state-space system is invariant—it is pre-computed off-line, which helps to reduce computational requirements at runtime that would accompany the online calculation of the Kalman gain.

The system must be observable with respect to the number of available sensed engine measurements to construct this steady-state Kalman filter. The goal of tuning is to eliminate model mismatch due to the unknown performance characteristics of the engine. It is assumed that the model itself is theoretically correct, but the actual engine may exhibit behavior that differs from the theoretical model due to performance degradation, manufacturing variations, or other unknown variables. The health parameters, which theoretically quantify these performance differences, can be selected as engine tuning parameters. Additionally, since these parameters remain relatively constant in the short-term, they are usually measured over the course of a single flight.

The observability issue may prove problematic when dealing with aircraft gas turbine engines. Often times the number of sensors available for use with the self-tuning engine model is less than the number of health parameters present in the model. To overcome this underdetermined estimation problem, two techniques are suggested to

transform the state-space matrices appropriately. If the health parameters are shifted to become states in our model in Equation 2, the system becomes:

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{h} \end{bmatrix} &= \begin{bmatrix} A & L \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta h \end{bmatrix} + B\Delta u \\ \Delta y &= \begin{bmatrix} C & M \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta h \end{bmatrix} + D\Delta u \\ \Delta z &= \begin{bmatrix} F & N \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta h \end{bmatrix} + G\Delta u \end{aligned} \quad (3)$$

Since engine performance deterioration evolves slowly in time, the health parameter states in Equation 3 are modeled without dynamics. Once the health parameters are augmented with the state variables, they can be estimated by applying a Kalman filter as long as the system is observable. However, a necessary condition for observability given the Equation 3 formulation is that there are at least as many measurements as health parameters (España, 1994). To construct a reduced-order state space system of appropriate dimension to enable Kalman filter formulation, consider a transformation matrix,  $V^*$ , that maps the health parameter vector,  $h$ , to a tuning vector of lower dimension,  $q$ , such that:

$$q = V^* h \quad (4)$$

An approximation for  $h$  based on  $q$  can be calculated using the pseudoinverse of  $V^*$ :

$$h = V^{*\dagger} q \quad (5)$$

Then, substituting Equation 5 into Equation 3 produces the following reduced-order state space system:

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{q} \end{bmatrix} &= \begin{bmatrix} A & LV^{*\dagger} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta q \end{bmatrix} + B\Delta u \\ \Delta y &= \begin{bmatrix} C & MV^{*\dagger} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta q \end{bmatrix} + D\Delta u \\ \Delta z &= \begin{bmatrix} F & NV^{*\dagger} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta q \end{bmatrix} + G\Delta u \end{aligned} \quad (6)$$

The choice of the transformation matrix is a design decision to be made prior to constructing the Kalman gains. To allow for piecewise interpolation of the Kalman gain, the value of the transformation matrix must remain constant regardless of operating condition so that the definition of the tuning vector does not change based on operating point (Simon, Armstrong, & Garg, 2011).

The first technique for dimensional reduction of the health parameters is to select a subset of health parameters to use as tuning parameters, effectively assuming the excluded parameters remain constant. In this scenario, the elements



of the transformation matrix will be comprised of ones and zeros appropriately selected to map the selected subset of health parameters properly. Defining this subset of parameters is a design decision. Based on a theoretical error analysis, the optimal subset can be algorithmically selected (Simon, Armstrong, & Garg, 2011). While this technique preserves the definition of the selected health parameters, the excluded health parameters cannot be estimated and “smearing” effects may cause inaccuracies in the estimation of the selected subset of parameters (Simon, Armstrong, & Garg, 2011).

A second technique, referred to as “optimal tuner selection,” can be employed to produce a transformation matrix that is a linear combination of all health parameters (Simon & Garg, 2010). This method involves optimizing the transformation matrix and the resultant definition of the tuning vector to minimize a desired estimation error. The error to be minimized is normally either the theoretical estimation error in a selection of unmeasured outputs ( $z$ ), health parameters ( $h$ ), or a combination of both. The selection of which errors to minimize is tailored to the intended usage of the self-tuning engine model. Because the value of the transformation matrix must not change with operating condition, a global optimization algorithm should be employed across the expected flight envelope (Simon, Armstrong, & Garg, 2011).

Once the dimensions of the tuning vector are reduced (if necessary) to make the estimation problem observable and the process and measurement noise covariance matrices are specified, the Kalman gain matrices are constructed at every operating point where a state-space system exists (Simon, Armstrong, & Garg, 2011). The general algorithm proceeds as follows, again, assuming three-dimensional interpolation:

*for each X in selected altitudes:*

*for each Y in selected Mach numbers:*

*for each Z in selected power levels:*

*Transform or reduce the state-space system per Equation 6*

*Calculate the associated Kalman gain*

*Append transformed state-space and Kalman gain matrix data sets*

Because of the mathematics inherent in computing the Kalman gain, most notably the algebraic Riccati equation, the gain matrix may not be calculable at some operating points (Zarchan & Musoff, 2005). In these cases, it is suggested that the entire state-space system be disposed of and replaced with the nearest (in terms of scheduling points) state-space system where a Kalman gain can be reliably computed. The failure to construct the Kalman gain may imply some mathematical stability issues with the state-space system used as a basis for the computation.

### 3.3. Data Storage

The algorithms presented in the previous section do not address data storage. The issue of storing the data generated is somewhat architecture and platform dependent, but some general guidelines are suggested to improve efficiency when using the resultant data.

For efficient interpolation, the authors found that it was beneficial to align the data in memory such that any given single vector or matrix in a data set containing multiple vectors or matrices exist in a single, congruent memory location. Placing each data set in a continuous memory block allowed for fast pointer arithmetic during interpolation. Access to vectors or matrices in the data set “in place” via pointer arithmetic avoided unnecessary penalties resulting from copying data to temporary storage during interpolation procedures.

The suggested memory layout of data described above should be considered when designing long-term storage of the data sets. If the data is initially generated in the proper format and saved to a permanent storage medium, the process of loading the data at runtime should result in an advantageous memory layout automatically. One suggestion is to store high-dimensional (3+ dimensions) data as concatenated two-dimensional arrays with scheduling as shown in Figure 3 for the three dimensional case. The advantage of this layout will become apparent when the interpolation implementation is discussed.

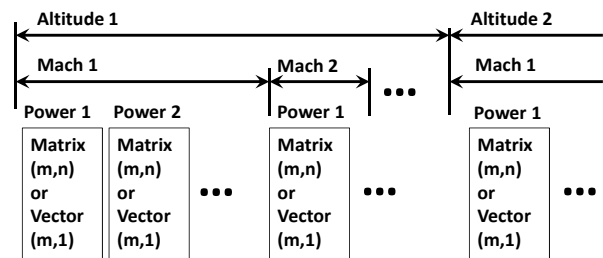


Figure 3. Suggested hierarchical data storage

### 4. MODEL IMPLEMENTATION

The implementation details surrounding the self-tuning engine model are application-specific. This section outlines an implementation that emphasizes efficiency and accuracy. The resultant design is appropriate for online, real-time applications and ground-based data analyses. The discussion will focus on a discrete-time implementation.

The overall self-tuning engine model design is illustrated in the block diagram in Figure 4. The self-tuning engine model requires sensed engine measurements ( $y$ ) and actuator

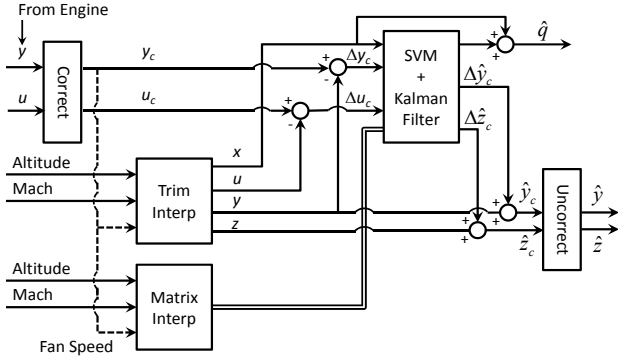


Figure 4. Self-tuning engine model overview

inputs ( $u$ ). The model produces sensor estimates ( $\hat{y}$ ), unmeasured engine parameter estimates ( $\hat{z}$ ), and tuning parameters ( $\hat{q}$ ). The sensor estimates should match the sensed measurements. The tuning parameters will be transformable back into estimates of engine performance parameters (Simon & Garg, 2010).

The individual components of this self-tuning engine model are explained below. The details of the implementation are independent of computing language and hardware platform.

#### 4.1. Parameter Correction

For aircraft engine applications, the use of corrected parameters within the self-tuning model is encouraged to improve accuracy and to reduce the number of operating points included in the piecewise linear design. Parameter correction is used to minimize the effects of atmospheric variations due to temperature and pressure (Volponi, 1998). The inlet total pressure and temperature sensors are employed to normalize parameters with respect to standard day sea level static conditions. In a gas turbine engine, it is likely that the only actuator input requiring correction will be a fuel flow command. The sensor measurements must also be corrected prior to use. The tuning parameters, which are considered to be proxies for the health parameters and are assumed to be independent of altitude and Mach number, will not require correction. The correction of trim values and system matrices is performed during linearization. Within the block shown in Figure 4, all values remain in corrected form, including inputs, to improve accuracy.

#### 4.2. Kalman Filter Implementation

The filter in this self-tuning engine model is a steady-state Kalman filter implementation. The Kalman gain, state-space matrices, and trim vectors are delivered to this module via the interpolation routines, as will be discussed. This implementation uses a discrete-time form of the Kalman filter. A block-diagram in Figure 5 outlines the structure of this Kalman filter. Here,  $k$  represents the discrete time

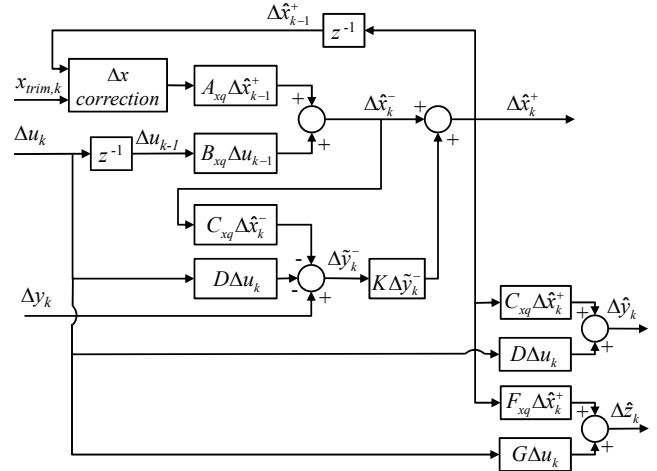


Figure 5. Block diagram of Kalman filter

index and the matrices and vectors have been augmented per Simon and Garg (2010) to include tuning parameters in the state vector,  $\Delta x$ , accompanied by proper state-space matrix modifications. Here, the “+” and “-” superscripts denote Kalman filter *a posteriori* and *a priori* estimates, respectively,  $\Delta \tilde{y}_k^-$  is the residual between the sensed and estimated measurement vector, and  $z^{-1}$  is the unit sample delay.

The implementation of a piecewise linear Kalman filter does pose unique implementation requirements. The Kalman filter, which is a recursive estimator, relies on state estimates calculated at the previous time step. However, on each time step in the piecewise linear implementation, the trim vectors, state-space matrices, and Kalman gain matrix are interpolated, and all are likely to shift from the previous time point. Therefore, the *a posteriori* state estimate calculated the previous time step will reflect a deviation relative to the state trim vector applied during the previous time step. Prior to use on the current time step, the *a posteriori* state estimate must be updated to reflect the change in the trim values, as shown below:

$$\Delta x'_{k-1} = \Delta x_{k-1} + (x_k - x_{k-1}) \quad (7)$$

In the above equation, the expression inside the parenthesis reflects the change in trim values from one time step to the next. Applying this adjustment ensures that deviations from trim are relative to the trim point applied at time step  $k$ , as opposed to the trim point previously applied at time step  $k-1$ . For additional details on the formulation of the Kalman filter, readers are referred to Simon & Garg, 2010.

#### 4.3. Interpolation Technique

Self-tuning engine model computational efficiency is highly dependent on the interpolation technique employed within the model. Therefore, to lessen overall model computational requirements the interpolation procedure

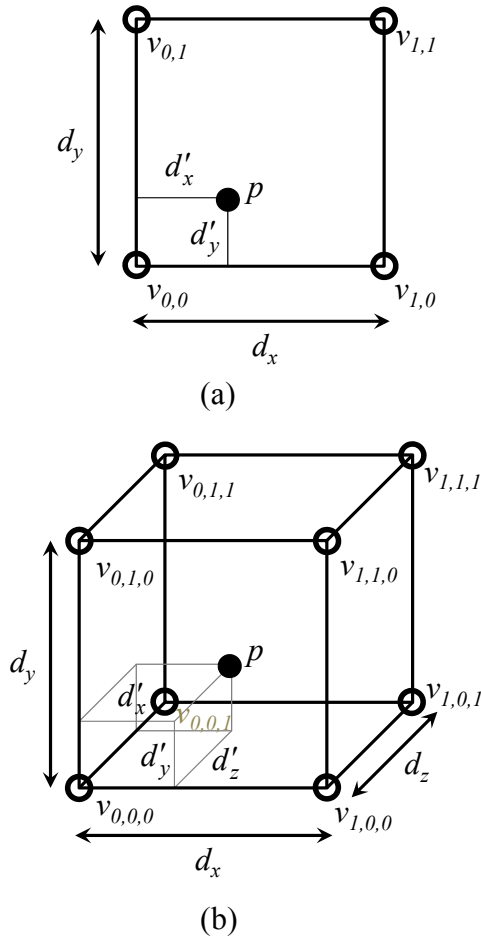


Figure 6. (a) Two- and (b) Three-dimensional interpolation

should employ rapid and efficient techniques. The earlier section, focusing on the storage of the trim vectors, state-space matrices, and Kalman gain matrix data sets, discussed the proper layout of data in memory for efficient interpolation. Once this data is stored in memory, efficient interpolation methods can be used across the multiple dimensions.

When working in multiple dimensions, linear interpolation requires an expansion. Consider the two simple interpolation cases in Figure 6. The point,  $p$ , at which to interpolate can be projected onto each of the interpolation axes, as the figure shows, and the point is bounded by either four or eight schedule points for the two- and three-dimensional cases, respectively. For each axis (i.e., dimension), weights are computed that represent the normalized distances from each bounding point on that axis. For a single axis, assuming  $d$  is the distance between two schedule operating points and  $d'$  is the distance between the projection of  $p$  onto the axis and the schedule point that precedes  $p$ , two weights can be calculated as:

$$\begin{aligned} w_0 &= 1 - (d' / d) \\ w_1 &= d' / d \end{aligned} \quad (8)$$

The weight  $w_0$  represents the contribution of the schedule point  $v_0$  that precedes the projection of  $p$ , while the weight  $w_1$  is the contribution of the schedule point  $v_1$  that follows the projection of  $p$ . The weight pairs are computed for each axis. This procedure will yield two pairs of weights for the two-dimensional interpolation case and three weighting pairs for the three-dimensional case. The interpolated value for the two-dimensional case (bi-linear interpolation) is then defined as:

$$\begin{aligned} x_p &= w_{x,0}w_{y,0}v_{0,0} + w_{x,1}w_{y,0}v_{1,0} + \\ &w_{x,0}w_{y,1}v_{0,1} + w_{x,1}w_{y,1}v_{1,1} \end{aligned} \quad (9)$$

Likewise, the interpolated value for three dimensions (tri-linear interpolation) would be:

$$\begin{aligned} x_p &= w_{x,0}w_{y,0}w_{z,0}v_{0,0,0} + w_{x,1}w_{y,0}w_{z,0}v_{1,0,0} + \\ &w_{x,0}w_{y,1}w_{z,0}v_{0,1,0} + w_{x,1}w_{y,1}w_{z,0}v_{1,1,0} + \\ &w_{x,0}w_{y,0}w_{z,1}v_{0,0,1} + w_{x,1}w_{y,0}w_{z,1}v_{1,0,1} + \\ &w_{x,0}w_{y,1}w_{z,1}v_{0,1,1} + w_{x,1}w_{y,1}w_{z,1}v_{1,1,1} \end{aligned} \quad (10)$$

The calculations described above require determining the bounding points along each axis from the schedules. A simple search for the desired index is usually sufficient, but some efficiency gains can be realized by using a bracketed, binary search with memory of the last successful search between requests. A binary search technique can rapidly search through a sorted array, such as our scheduling axes, with a worst case  $O(\log n)$ \* performance (Knuth, 1997). An additional improvement can be gained by storing the index of the lower bounding point for each axis at each time step. Because the change in operating point on each time step is likely to be relatively small, the previously used index on each axis can be rapidly checked to see if it is still applicable rather than performing a binary search on every time step. The performance on the majority of time steps would remain at  $O(1)$ , and only the applied weights would require recalculation.

Another way to improve efficiency is to limit the number of schedule searches and subsequent weight calculations based on the sharing of scheduling axes. The trim point vectors would all share one set of scheduling axes, while the state-space matrices and Kalman gains would use a less dense set. Therefore, only two passes of weight calculations would be necessary, one for trim vectors and another for matrices. The weights could then be shared between trim point vector interpolations when applying the data set for each parameter

\* Shown in Big O notation indicating that worst case computational time grows proportional to  $\log n$ , where  $n$  is the number of grid points on the scheduling axis.

to either equation 9 or 10. Similarly, the matrix weights could be shared for all the state-space matrix data sets and the Kalman gain data set. This approach decreased the number of schedule searches by a factor of five for this three-dimensional model design.

As explained earlier, the memory layout of the data can lead to dramatic improvements in performance. Matrix interpolation may seem costly, and this implementation is suggesting a minimum of five matrix interpolations per time step, with each involving a considerable number of products to be computed. To minimize the impact of the large number of products necessary, one method may be to exploit “single instruction, multiple data,” or SIMD, instructions that are conveniently available on many modern central processing units, including modern embedded processors (ARM, 2010-2011), (Intel Corporation, 1997-2012), (International Business Machines Corporation, 2006). Rather than focus on processor-specific capabilities, the use of the Basic Linear Algebra Subprograms, or BLAS, library is suggested (Lawson, Hanson, Kincaid, & Krogh, 1979). Some modern interpreted languages will use these procedures internally, and modern optimizing compilers can often detect and use these procedures in a manner transparent to the designer. Examining equations 9 and 10, one may notice that the BLAS routines “\*AXPY,” which multiplies a vector by a scalar (our weights) and adds the product to another vector, can be applied multiple times to calculate the desired interpolated value (BLAS, 2011). Using such routines eliminates the element-by-element multiplication that might be used naively, allowing the SIMD capabilities of the processor to be used.

The continuous memory locations, which had been suggested earlier, allow for further efficiency improvement. If each data set is held in a single memory block, each matrix or vector can be accessed “in-place” rather than copying or extracting the matrix or vector elements to an appropriately sized array prior to weighting each individual point. In a lower level language, pointer arithmetic can be exploited to specify the location in memory of an individual matrix or vector within a data set. By employing these efficiency gains, the computational costs of performing multidimensional interpolation of matrices and vectors are minimized without sacrificing accuracy.

## 5. EXAMPLE RESULTS

For evaluation purposes, this self-tuning engine model is compared against the nonlinear engine model upon which it is based. To illustrate the capabilities of the suggested design, an appropriate self-tuning engine model has been derived from the Commercial Modular Aero-Propulsion System Simulation 40k, or C-MAPSS40k, a nonlinear aerothermal model that simulates a 40,000lbf-class turbofan engine (May, Csank, Lavelle, Litt, & Guo, 2010). The self-tuning engine model was designed to reflect an engine at

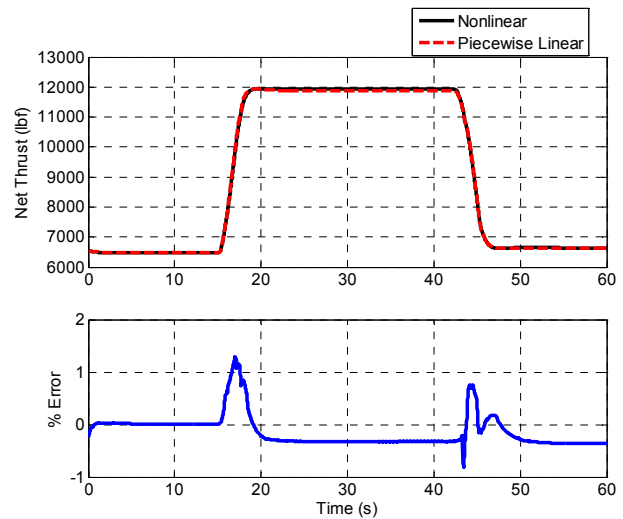


Figure 7. Thrust calculations using nonlinear and piecewise linear models at 25,000 ft and 0.55 Mach

50% of useful life remaining. The piecewise linear state-space model and Kalman filter have been designed as discrete-time models using the same time step size as the nonlinear model serving as the comparison basis.

First, the standalone piecewise linear model (without tuning) is compared with the nonlinear model to determine the accuracy of the linearization and effectiveness of the interpolation algorithms. Figure 7 shows the thrust estimate for the piecewise linear model as compared to the nonlinear C-MAPSS40k model running at the mean degradation design point for a rapid power increase followed by a subsequent decrease to the original power level. The altitude and Mach number conditions tested lie between schedule points, meaning the interpolation algorithm is being exercised in this example. The percent error graph, which examines the point-by-point difference in thrust between the piecewise linear model versus the nonlinear model, shows a noticeable increase in the residuals during transients.

The self-tuning model provides much of its advantage for engines that operate away from the model degradation design point. Under the same transient situation with tuning enabled, the self-tuning model should be able to maintain accuracy when applied to engines that are not represented by the mean performance level. Figure 8 illustrates the accuracy of a self-tuning engine model when estimating the unmeasured combustor exit temperature for an ideal (new) engine and an end-of-life engine. The combustor exit temperature has been chosen for comparison because it experiences significant shift as engine performance degrades. In this example, the nonlinear engine model serves as the “truth” model and its sensed outputs and actuator commands are provided as inputs to the self-tuning engine model. The figure shows the outputs of the nonlinear

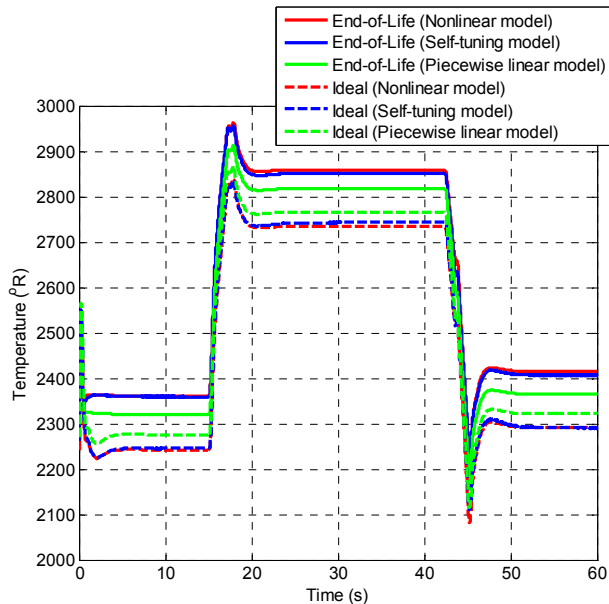


Figure 8. Unmeasurable combustor exit temperature estimates for an off-design engine

engine model (red), the self-tuning engine model (blue), and the piecewise linear model without tuning (green). Here, the estimate of combustor exit temperature produced by the self-tuning engine model exhibits good matching with the nonlinear model at these two extreme performance ranges. Conversely, the piecewise linear model, which does not have self-tuning functionality, is unable to match the nonlinear model as well.

The self-tuning engine model is considerably simpler than the full nonlinear model. While C-MAPSS40k is capable of running faster than real-time on modern consumer-grade computing hardware, the self-tuning engine model exhibits better computational performance due to its efficient interpolation algorithm and the inherent simplicity of its linear design. Compared to the nonlinear model, the self-tuning engine model runs approximately an order of magnitude faster than the nonlinear model on the same computer.

## 6. DISCUSSION

The main advantage of the self-tuning engine model is its ability to eliminate the mismatch between the theoretical model and the actual engine. Because of this feature, on-board implementations of such models may be desirable for a variety of reasons, including control system integration, on-board model-based engine diagnostics, and simple informational purposes. However, this model loses much of its utility if it is unable to accurately estimate parameters during transient operation.

The accuracy during transients is related to a number of design choices made during the generation of the self-tuning

model's data. The schedule density of the piecewise linear model must always be considered during the design phase. Comparison of the piecewise linear model itself against its nonlinear basis model during simple transients represents a "best case" accuracy that can be achieved by the subsequent self-tuning engine model. Often times, adjustments to the perturbations during linearization must be made to better improve dynamic accuracy. Additionally, some highly nonlinear parameters, such as stall margins, may not lend themselves to linearization inherently; these parameters appear to result in large mismatches even when great care is taken to improve accuracy.

Transient behavior of the tuning parameters within the Kalman filter is normally adjusted via modification of the process noise. Scaling of the noise has been shown to accelerate or decelerate the response of tuning parameters to engine transients (Simon, Armstrong, & Garg, 2011). Furthermore, ongoing research suggests that numerical factors related to the globally optimal tuner selection strategy may produce tuner transformation matrices that have undesirable transient properties. The current optimization algorithm does not consider the transient behavior of resultant Kalman filters, only the steady-state errors. Depending on the models involved, this issue may or may not be encountered during implementation of a self-tuning engine model.

The advantage of this self-tuning engine model is the relative simplicity of its design, among other benefits. The structure of the Kalman filter, although this model uses a piecewise version, lends itself to well-known verification and validation, or V&V, procedures (Schumann & Liu, 2007). For online implementations or control system integration, the ability to perform V&V on this self-tuning engine model using accepted processes is advantageous when comparing against alternatives such as a full nonlinear, physics-based online model.

## 7. CONCLUSION

A piecewise linear Kalman filter has been proposed as a self-tuning engine model solution. The well-understood Kalman filter algorithm combined with an efficient implementation make this piecewise solution an attractive candidate for resolving differences between theoretical, physics-based models and actual engine hardware.

The two outputs of the self-tuning engine model, tuning parameters and unmeasured parameter estimates, can be exploited for a variety of purposes. Estimates of these parameters, which could be employed in either ground-based or on-board solutions, could be used for performance trending and assisting in current engine health management programs. Such trending information is also useful for engine diagnostic algorithms by allowing these conceptual algorithms to discern between normal engine degradation and possible faults. The estimated parameters, while useful

for informational purposes alone, introduce the prospect of advanced parameter synthesis and control algorithms, including controlling directly on thrust. Possible efficiency gains may be realized through the accurate estimation of engine-specific operational limits, providing the opportunity to relax generally conservative stall margin, temperature, and pressure limits.

#### ACKNOWLEDGEMENT

The research associated with this work was performed under the Vehicle Systems Safety Technologies project as part of the National Aeronautics and Space Administration's Aviation Safety Program.

#### NOMENCLATURE

$A, B, C,$	
$D, F, G,$	Linear state-space system matrices
$L, M, N$	
$d, d'$	Distance in interpolation calculations
$h$	Health parameter vector
$q$	Engine tuning parameter vector
$U$	Engine control input vector
$V^*$	Transformation matrix mapping health parameters to engine tuning parameters
$X, Y, Z$	Interpolation scheduling axes
$f$	Nonlinear function of engine state derivatives
$g$	Nonlinear function of engine outputs
$u$	Control inputs
$v$	Value placeholder for interpolation
$w$	Weighting for interpolation
$x$	Engine state vector
$y$	Engine sensed measurement vector
$z$	Engine unmeasured parameter vector
$z^{-1}$	Unit sample delay
$\Delta$ prefix	Deviation from trim value
$\delta$ prefix	Perturbation value
<b>Subscripts</b>	
$c$	Corrected value
$k$	Discrete time index
$ref$	Health parameter reference vector
$trim$	Trim vector
<b>Superscripts</b>	
$\dagger$	Pseudoinverse
$-$	<i>a priori</i> Kalman filter estimate
$+$	<i>a posteriori</i> Kalman filter estimate
<b>Diacritical Marks</b>	
$\hat{\phantom{x}}$	Estimated value
$\sim$	Residual between estimated and sensed measurement vector

#### REFERENCES

- ARM. (2010-2011). ARMv6 SIMD Instruction Intrinsics. In *ARM Compiler Toolchain Version 4.1: Compiler Reference* (pp. A-1 - A-67). ARM.
- Armstrong, J. B., & Simon, D. L. (2011). Implementation of an Integrated On-Board Aircraft Engine Diagnostic Architecture. *47th AIAA Joint Propulsion Conference*. San Diego, CA.
- Behbahani, A., Adibhatla, S., & Rauche, C. (2009). Integrated Model-Based Controls and PHM for Improving Turbine Engine Performance, Reliability, and Cost. *45th AIAA Joint Propulsion Conference*. Denver, CO.
- BLAS. (2011). Retrieved April 30, 2012, from <http://www.netlib.org/blas/>
- Brotherton, T., Volponi, A., Luppold, R., & Simon, D. (2003). eSTORM: Enhanced Self Tuning On-board Real-time Engine Model. *Proceedings of the 2003 IEEE Aerospace Conference*. Big Sky, MT.
- Brunell, B., Bitmead, R., & Connolly, A. (2002). Nonlinear Model Predictive Control of an Aircraft Gas Turbine Engine. *Proceedings of the IEEE Conference on Decision and Control*, 4, pp. 4649-4651. Las Vegas, NV.
- Bushman, M. A., & Gallops, G. A. (1992). In-Flight Performance Diagnostic Capability of an Adaptive Engine Model. *28th AIAA Joint Propulsion Conference*. Nashville, TN.
- Dwyer, W. J. (1990). *Adaptive Model-Based Control Applied to a Turbofan Aircraft Engine*. Cambridge, MA: Massachusetts Institute of Technology.
- España, M. D. (1994). Sensor Biases Effect on the Estimation Algorithm for Performance-Seeking Controllers. *ASME Journal of Propulsion and Power*, 10, 527-532.
- Gallops, G. W., Gass, F. D., & Kennedy, M. H. (1992). On-Board Condition Management for Aircraft Gas Turbines. *37th ASME International Gas Turbine and Aeroengine Congress and Exposition*. Cologne, Germany.
- Gilyard, G. B., & Orme, J. S. (1993). *Performance Seeking Control: Program Overview and Future Directions*. NASA.
- Intel Corporation. (1997-2012). *Intel Architecture Instruction Set Extensions Programming Reference*.
- International Business Machines Corporation. (2006). *PowerPC Microprocessor Family: Vector/SIMD Multimedia Extension Technology Programming Environments Manual*. Hopewell Junction, NY.
- Klaus, L., & Kreiner, A. (2001). Model Based Control Concepts for Jet Engines. *ASME Turbo Expo 2001*. New Orleans, LA.
- Knuth, D. (1997). *The Art of Computer Programming, Volume 3: Sorting and Searching*. Reading, Massachusetts: Addison-Wesley.
- Lawson, C. L., Hanson, R. J., Kincaid, D., & Krogh, F. T. (1979). *Basic Linear Algebra Subprograms for*

- FORTTRAN Usage. *ACM Transactions on Mathematical Software*, 308-323.
- Luppold, R. H., Roman, J. R., Gallops, G. W., & Kerr, L. J. (1989). Estimating In-Flight Engine Performance Variations Using Kalman Filter Concepts. *25th AIAA Joint Propulsion Conference*. Monterey, CA.
- May, R. D., Csank, J., Lavelle, T. M., Litt, J. S., & Guo, T. H. (2010). A High-Fidelity Simulation of a Generic Commercial Aircraft Engine and Controller. *46th AIAA Joint Propulsion Conference*. Nashville, TN.
- Nobbs, S. G., Jacobs, S. W., & Donahue, D. J. (1992). Development of the Full-Envelope Performance Seeking Control Algorithm. *28th AIAA Joint Propulsion Conference*. Nashville, TN.
- Sallee, G. (1978). *Performance Deterioration Based on Existing (Historical) Data – JT9D Jet Engine Diagnostics Program*.
- Schumann, J., & Liu, Y. (2007). Tools and Methods for the Verification and Validation of Adaptive Aircraft Control Systems. *2007 IEEE Aerospace Conference*. Big Sky, MT.
- Shaw, P., Foxgrover, J., Berg, D. F., Swan, J., Adibhatla, S., & Skira, C. A. (1986). A Design Approach to Performance Seeking Control. *22nd AIAA Joint Propulsion Conference*. Huntsville, AL.
- Simon, D. L., & Garg, S. (2010, March). Optimal Tuner Selection for Kalman Filter-Based Aircraft Engine Performance Estimation. *Journal of Engineering for Gas Turbines and Power*, 132.
- Simon, D. L., Armstrong, J. B., & Garg, S. (2011). Application of an Optimal Tuner Selection Approach for On-Board Self-Tuning Engine Models. *Proceedings of the ASME Turbo Expo 2011*.
- Volponi, A. (2008). *Enhanced Self Tuning On-Board Real-Time Model (eSTORM) for Aircraft Engine Performance Health Tracking*. National Aeronautics and Space Administration.

- Volponi, A. J. (1998). Gas Turbine Parameter Corrections. *ASME International Gas Turbine and Aeroengine Congress and Exposition*. Stockholm, Sweden.
- Zarchan, P., & Musoff, H. (2005). *Fundamentals of Kalman Filtering: A Practical Approach*. AIAA.

## BIOGRAPHIES



**Jeffrey B. Armstrong** is a Controls Engineer working as a contractor at the NASA Glenn Research Center in Cleveland, Ohio. He holds bachelor (2000) and master of science (2002) degrees in aerospace and mechanical engineering from Case Western Reserve University and has years of experience designing and implementing numerical simulations. In the past he has worked in microgravity research, rocket trajectory validation and verification, and enterprise medical software fields. Currently Jeffrey participates in air-breathing propulsion diagnostics research.



**Donald L. Simon** is a Controls Engineer at the NASA Glenn Research Center. He holds a B.S. degree from Youngstown State University (1987) and a M.S. degree from Cleveland State University (1990), both in electrical engineering. During his career he has focused on the development of advanced control and health management technologies for current and future aerospace propulsion systems. Mr. Simon's specific research interests are in aircraft gas turbine engine performance diagnostics and performance estimation. He currently leads the propulsion gas path health management research effort ongoing under the NASA Aviation Safety Program, Vehicle Systems Safety Technologies Project.

# Maintenance Planning with Prognostics for Systems Located In Various Places

F. Camci<sup>1</sup>, M. Sevkli<sup>2</sup>, M. Karakas<sup>2</sup>, I. K. Jennions<sup>1</sup>

<sup>1</sup>*IVHM Centre, Cranfield University, UK*

*f.camci@cranfield.ac.uk*

*i.jennions@cranfield.ac.uk*

<sup>2</sup>*School of Engineering, Fatih University, TR*

*msevkli@fatih.edu.tr*

*mkarakas@fatih.edu.tr*

## ABSTRACT

Predictive maintenance has been attracting researchers and industry in recent years, since maintenance and repair of assets is one of the most contributing factors of operating & support cost. Predictive maintenance proposes to maintain the assets only when necessary aiming to reduce the unnecessary repair and maintenance by monitoring the health of the assets. The expected time of the failure is estimated by analyzing the monitored signals and remaining useful life of the asset before failure is used to plan, get prepared and perform the maintenance. When one team is responsible for maintenance of systems that are located in various places, the travel time between these systems should also be incorporated in the maintenance planning. Off shore wind farms and railway switches are two examples of these systems. This paper presents formulation of the problem that incorporates travel times between systems and prognostics information obtained from each system.

## 1. INTRODUCTION

Predictive maintenance has been attracting researchers and industry in recent years, since maintenance and repair of assets is one of the most contributing factors of operating & support cost (Camci & Chinnam, 2010). Predictive maintenance, also called Condition Based Maintenance (CBM), proposes to maintain the assets only when necessary aiming to reduce the unnecessary repair and maintenance by monitoring the health of the assets. The term asset represents any system that is monitored for

predictive maintenance.

In CBM, health of the asset is observed real time by analyzing signals collected from sensors embedded on the asset. Two main aspects of CBM are diagnostics and prognostics (Jardine et. al. 2006). Diagnostics is the process of detection of an existing incipient failure, whereas prognostics is the process of forecasting the time of the failure and identification of remaining useful life (RUL) of the asset before failure (Camci & Chinnam 2010a). Maintenance is scheduled immediately for an asset when a failure is diagnosed. On the other hand, when a failure is prognosed, the identified remaining useful life can be used for planning, preparation, and performing the maintenance. The asset is expected to perform its functionality maybe with less efficiency within the identified RUL.

In predictive maintenance research, RUL information may be given in different formats such as the real time (e.g., three months to failure), operational working period (e.g., 3000 miles of driving before failure), efficiency decrease (%95 efficiency), or probability of failure (0.20 probability to fail) (Camci & Chinnam 2010a). In most maintenance scheduling methods with RUL information, a threshold value is set to RUL (Barbera et. al. 1996, Marseguerra et. al. 2002, Sloan & Shanthikumar 2002, Yam et. al., 2001, Berenguer et. al. 2003). In these methods, maintenance is performed when the RUL reaches the threshold. Even if the threshold is optimized perfectly for an asset, the cost of travelling for maintenance will be a critical issue that cannot be incorporated easily in the static threshold optimization. An optimization model without threshold setting has been discussed in (Camci F. 2009), (Camci F. 2009a). However, the optimization model presented in these studies assumes that all the components or systems are located in the same place and do not incorporate travel time between systems.

---

F. Camci et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.



Consider an off shore wind farm with many wind turbines located in the middle of the sea. Assume that one of the wind turbines is scheduled for maintenance next week, since the RUL is expected to reach to the threshold next week. A team of maintenance operators will be sent to the sea for maintenance. Consider that two wind turbines are located 10 miles and 20 miles away from the wind turbine scheduled for maintenance. The RUL of these wind turbines are identified to be two and three weeks, respectively. Maintenance scheduling based on only RUL thresholds will not always give the best maintenance scheduling. Thus, cost of travelling should also be incorporated within the maintenance scheduling. This paper presents the problem of incorporating the maintenance scheduling with RUL information for geographically distributed assets. Section 2 presents the problem modeling, and section 3 demonstrates the problem with small number of systems. Section 4 concludes the paper.

## 2. PROBLEM DEFINITION

Consider multiple systems that are located in various places and are within the responsibility of one maintenance team. The health of each asset is observed using condition monitoring techniques and forecasted by analyzing the signals obtained using prognostics techniques. Forecasted health (RUL information) is assumed in the format of probability of failure.

The problem aims to identify the most cost effective maintenance routing. The cost function to be minimized involves three major terms: expected failure cost, maintenance cost, and travel cost. All three terms are highly correlated with the time of the maintenance of each asset. Early maintenance reduces the failure risk more than late maintenance. The order of the locations to be visited for maintenance affects the time of maintenance in each location. For example, if maintenance is scheduled for an asset that requires long travelling time, then all other assets waiting for maintenance will be negatively affected from this decision. Fig. 1 illustrates the effect of early, late, and no maintenance on failure probability of a system for a given time frame.

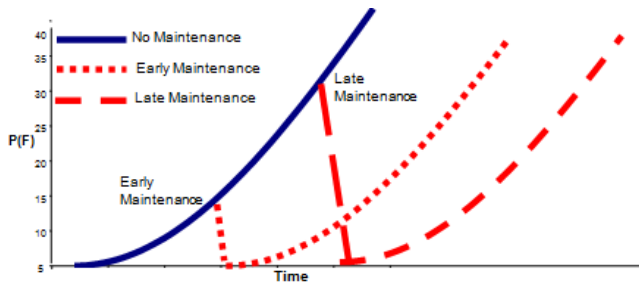


Fig.1 Illustration of reducing risk by maintenance

The problem aims to minimize the total cost that consists of failure, maintenance, and travel cost by finding the best

maintenance schedule of n geographically distributed assets. The objective function is the sum of each asset cost.

In the simple case of the problem, it is assumed that every location is visited only once and the tour ends at the initial location. Thus, the maintenance cost will be constant assuming that one and only one maintenance will be performed at each location. Thus, only failure cost will be considered in the formulation, since total maintenance cost will be constant. Mathematically, the problem can be expressed as follows: find a permutation  $(\pi_1 \pi_2 \dots \pi_n)$  of integers from 1 to n that minimizes the total cost of travelling, expected failure and maintenance cost of assets given a distance matrix  $D = d_{i,j}$ , where  $d_{i,j}$  is the distance between locations i and j. This is the first objective function to be explored in this paper and shown in (1). Note that the distance information is actually considered as time-to-travel information. If it is given as real distance, it should be converted to time-to-travel, since the time information affects the expected failure cost, not the distance.

$$\sum_{i=1}^n f_{\pi_i} + \sum_{k=0}^i d_{\pi_k, \pi_{k+1}} \quad (1)$$

$\pi_i$ : Asset number scheduled for maintenance with  $i^{th}$  order (starting point when  $i=0$ )

$f_{\pi_i}$ : Failure cost function for system located at  $\pi_i$

$Tm_{\pi_i}$ : Scheduled maintenance time for location  $\pi_i$

$\sum_{k=0}^i d_{\pi_k, \pi_{k+1}}$ : Time to travel to reach from  $\pi_k$  to  $\pi_{k+1}$

Failure cost is the product of fixed failure cost and the cumulative failure probability ( $CP_{i,T}$ ) of the asset as shown in (2). Effects of maintenance times should be considered in calculation of cumulative failure probability ( $CP_{i,T}$ ) within the given period (T). The given period is divided into periods as before and after maintenance since only one maintenance in the given period is assumed.  $CP_{i,T}$  is calculated as the union of failure probabilities in before and after maintenance periods ( $P_{i,b}$ ,  $P_{i,a}$  failure probabilities before maintenance, after maintenance respectively) since failure may occur either before or after maintenance in the given period. The formulation of  $CP_{i,T}$  calculation is given in (3).

$$FR_i = CP_{i,T} \times F_i \quad (2)$$

$$CP_{i,T} = P_{i,b} + P_{i,a} - P_{i,b} \quad (3)$$

$P_{i,b}$ : Cumulative failure probability starting time 0 till just before the maintenance

$P_{i,a}$ : Cumulative failure probability starting from maintenance time till end of given period

Forecasted failure probabilities can be obtained in two ways: prognostics-based and reliability-based. In prognostics-based calculation, failure probabilities ( $Pfo$ )

represent the specific failure probability of the asset under observation with CBM. This failure probability ( $Pfo$ ) is used until the maintenance time of the system. Many prognostics methods to predict failure probabilities have been presented in the literature (Lu H. et. al.2001, Xu et. al., 2008, Xu et. al., 2009, Heng et. al. 2009, Sritavastava & Das 2009). After the maintenance, failure probabilities obtained from reliability-based calculation is used. Reliability based failure probability ( $Pfw$ ) is obtained from the reliability analysis of past similar assets. Reliability based failure probability ( $Pfw$ ) is not specific to the asset under observation. We do not use prognostics-based failure probability after maintenance because we do not know how the asset under observation will behave after maintenance (maintenance has not happened yet.)

Probability of failure in each section bases on the maintenance time and corresponds the failure probability value at the time the maintenance will be performed as shown in (4) and (5). Note that ( $Pfw$ ) and ( $Pfo$ ) are received as input in the problem.

$$P_{i,b} = Pfo_{tm} \tag{4}$$

$$P_{i,a} = Pfw_{T-tm} \tag{5}$$

The second objective function involves failure and travel costs as shown in (6). Travel cost simply is calculated as product of traveling cost one unit distance and total travel to reach location  $\pi_i$ .

$$\sum_{i=1}^n f_{\pi_i} \sum_{k=0}^i d_{\pi_k, \pi_{k+1}} + g \sum_{i=0}^n d_{\pi_i, \pi_{i+1}} \tag{6}$$

$$g \sum_{i=0}^n d_{\pi_i, \pi_{i+1}} : \text{Cost of travelling from } \pi_0 \text{ to } \pi_n$$

The third objective function involves maintenance cost as well as failure and travel cost as shown in (7). This is important when systems at various locations may be maintained in different number of times in the given period and they are different with different maintenance costs ( $M_{\pi_i}$ ).

$$\sum_{i=1}^n f_{\pi_i} \sum_{k=0}^i d_{\pi_k, \pi_{k+1}} + g \sum_{i=0}^n d_{\pi_i, \pi_{i+1}} + \sum_{i=0}^n M_{\pi_i} \tag{7}$$

### 3. DEMONSTRATION OF THE PROBLEM

In this section, the problem is demonstrated with a simple case study. In this case study, there are four locations to perform maintenance, and a starting point  $\pi_0$ . Distances between locations are shown in Table 1.

Table 1 Distance matrix

Locations	0	1	2	3	4
0	0	4	11	6	2
1	4	0	8	10	6

2	11	8	0	13	10
3	6	10	13	0	4
4	2	6	10	4	0

Fixed failure and maintenance costs for the systems at these four customers are indicated in Table 2.

Table 2 General Parameters of first example

Locations	1	2	3	4
Failure Cost (\$)	3500	3200	300	5100
Maintenance Cost (\$)	450	230	400	150

Table 3: Results of the case study

Visiting Order				Time	Total Cost		
$\pi_1$	$\pi_2$	$\pi_3$	$\pi_4$		OF1 in Eq. 1	OF2 in Eq. 6	OF3 in Eq. 7
4	3	2	1	31	2809.85	<b>5909.85</b>	<b>7139.85</b>
4	3	1	2	35	2863.45	6363.45	7593.45
4	2	3	1	39	2990.04	6890.04	8120.04
4	2	1	3	36	2878.36	6478.36	7708.36
4	1	2	3	35	2869.59	6369.59	7599.59
4	1	3	2	42	3049.07	7249.07	8479.07
3	4	2	1	32	<b>2785.23</b>	5985.23	7215.23
3	4	1	2	35	2811.33	6311.33	7541.33
3	2	4	1	39	2945.77	6845.77	8075.77
3	2	1	4	35	2901.21	6401.21	7631.21
3	1	2	4	36	2910.32	6510.32	7740.32
3	1	4	2	43	2961.12	7261.12	8491.12
2	3	4	1	38	2926.14	6726.14	7956.14
2	3	1	4	42	3080.28	7280.28	8510.28
2	4	3	1	39	2917.07	6817.07	8047.07
2	4	1	3	43	2951.86	7251.86	8481.86
2	1	4	3	35	2819.3	6319.3	7549.3
2	1	3	4	35	2884.5	6384.5	7614.5
1	3	2	4	39	3021.5	6921.5	8151.5
1	3	4	2	39	2915.29	6815.29	8045.29
1	2	3	4	31	2823.15	5923.15	7153.15
1	2	4	3	32	2786.4	5986.4	7216.4
1	4	2	3	39	2921.64	6821.64	8051.64
1	4	3	2	38	2910.35	6710.35	7940.35

To simplify the problem, it is assumed that each location will be visited at least and at most once. This limits the total number of possible visit alternatives to 24. In Table 3, total costs for all these alternatives are shown. First four columns in Table 3 show the visiting order of locations. Fifth column is the total travelled distance. Sixth column gives the total cost obtained from the first objective function in (1). The

seventh column displays the second objective function given in (6). The last column is the third objective function in (7).

Changes in the visit order changes the time of the visit for each location. Thus, the failure probability for the system at each location for a different visit order becomes different leading to different failure costs, which cause variations in the values of the first objective function. The best result for the first objective function is obtained from the visit order (3-4-2-1) given in the 7<sup>th</sup> row with \$2785.23.

The calculation of the second objective function involves the cost of travel in addition to failure cost. It is assumed that traveling one unit distance-time costs 100\$. The best routing is given in the first permutation (i.e., 4- 3- 2- 1) with minimum cost of \$5909.85. It is shown in this example that the travel cost affects the decision of the maintenance order.

The third objective function includes the maintenance costs as well as failure and travel cost. This example was simplified with the assumption of visiting each location at least and at most once. Thus, the maintenance cost for each permutation will be the same. Even though maintenance times are different, systems at all locations will be maintained once. Thus, the ranking of results of the second and third objective function are the same. If this assumption is removed, then some of the locations may be visited multiple times or some are not visited at all. Table 4 gives an example of costs for a given visit order without any visit limits.

Table 4: Result example without visit limit

Visiting Order				Time	Total Cost			Main't Cost
$\pi_1$	$\pi_2$	$\pi_3$	$\pi_4$		OF1 in Eq. 1	OF2 in Eq. 6	OF3 in Eq. 7	
4	3	2	1	31	2809.85	5909.85	7139.85	1230
4	1	2	4	28	2016.56	4816.56	5646.56	830

As seen from the table, not visiting location three reduces the failure cost because the failure consequence of the system in location three is very low compared to the system in location four (\$300 in location 3, \$5100 in location 4). Thus, replacing maintenance in location three with location four reduces the total failure cost. In addition, maintenance cost in location four is lower compared to location three. Total travel time is also low if location three is not visited at all. Note that this example is selected to demonstrate the nature of the problem, not fully demonstrate a real system.

#### 4. CONCLUSION

The cost reduction pressure on industry has increased research on condition based maintenance. Maintenance schedule now bases on detection and forecasting of failures. This paper presents a problem formulation to be used in maintenance planning that incorporates prognostics output with distances between systems located in various places. Formulation and numerical example are presented in the paper. As future work, it is recommended to develop

heuristic or meta-heuristics algorithms for the given problem formulation with large number of locations.

#### REFERENCES

Barbera F., Schneider H., & Kelle P., (1996) A condition based maintenance model with exponential failures and fixed inspection interval, *The Journal of the Operational Research Society*, 47(8), 1037–1045

Berenguer C., Grall A., Diulle L., & Roussignol M., (2003) Maintenance policy for a continuously monitored deteriorating system, *Probability in the Engineering and Informational Sciences*, 17(2), 235- 250

Camci F. (2009a), Comparison of Genetic and Binary PSO Algorithms on System Maintenance Scheduling Using Prognostics Information, *Engineering Optimization*, 41(2), 119-136

Camci F., (2009) System Maintenance Scheduling With Prognostics Information Using Genetic Algorithm, *IEEE Transaction on Reliability*, 58(3)

Camci F., Chinnam R. B., (2010) *Process Monitoring, Diagnostics and Prognostics in Machining Processes*, LAP Lambert Academic Publishing, 978-3838335667

Camci F., Chinnam R. B., (2010a), Health-State Estimation and Prognostics in Machining Processes", *IEEE Transactions on Automation Science and Engineering*, 7(3), 581-597

Heng A., Tan A. C.C., Mathew J., Montgomery N., Banjevic D., (2009) Intelligent Condition-based Prediction of Machinery Reliability, *Mechanical Systems and Signal Processing*, 23, 1600-1614

Jardine A.K.S., Lin D., and Banjevic D., (2006) A review on machinery diagnostics and prognostics implementing condition-based maintenance, *Mechanical Systems and Signal Processing*, 20(7), 1483-1510

Lu H., Kolarik W. J., Lu S. S., (2001) Real-Time Performance Reliability Prediction, *IEEE Transactions on Reliability*, 50(4), 353-357

Marseguerra M., Zio E., & Podofillini L., (2002) Condition-based maintenance optimization by means of genetic algorithm and monte carlo simulation, *Reliability Engineering and System Safety*, 77(2), 151–166

Sloan T. W., & Shanthikumar J. G., (2002) Using in-line component condition and yield information for maintenance scheduling and dispatching in semiconductor wafer fabs, *IIE Transactions*, 34(2), 191–209

Sritavastava A. N., Das S., (2009) Detection and Prognostics on Low-Dimensional Systems, *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, 39(1), 44-54

Xu Z., Ji Y., Zhou D., (2008) Real-time Reliability Prediction for a Dynamic System Based on the Hidden Degradation Process Identification, *IEEE Transactions on Reliability*, 57(2), 230-242

- Xu Z., Ji Y., Zhou D., (2009) A New Real-Time Reliability Prediction Method for Dynamic Systems Based on Online Fault Prediction, *IEEE Transactions on Reliability*, 58(3), 523-537
- Yam R. C. M., Tse P.W., Li L. & Tu P. (2001), Intelligent predictive decision support system for condition-based maintenance, *International Journal of Advanced Manufacturing Technology*, 17(5), 383-391

short course in IVHM and the world's first IVHM MSc, begun in 2011.

Ian is on the editorial Board for the International Journal of Condition Monitoring, a Director of the PHM Society, contributing member of the SAE IVHM Steering Group and HM-1 IVHM committee, a Fellow of IMechE, RAeS and ASME. He is the editor of the recent SAE book: IVHM – Perspectives on an Emerging Field.

#### ACKNOWLEDGEMENT

This research was supported by the The Scientific and Technological Research Council of Turkey (TUBITAK) under Project 108M275.

#### BIOGRAPHIES

**Fatih Camci** has been a faculty member at the IVHM Centre at Cranfield University, UK since 2010. He works on development of diagnostics, prognostics and maintenance planning technologies for electro-mechanical systems. Previously, Dr. Camci worked as an Assistant Professor at Fatih University in Turkey and as senior project engineer at Impact Technologies, in Rochester NY before joining Cranfield University. Dr. Camci received his BSc and MSc degrees in Computer Engineering at Istanbul and Fatih University in Turkey. He received his PhD in Industrial Engineering from Wayne State University, USA.

**Mehmet Sevkli** is an Associate Professor in Industrial Engineering Department, Fatih University, Istanbul, Turkey. He received his B.S. in Industrial Engineering from Sakarya University, his M.Sc. in Industrial Engineering from Fatih University and his Ph.D. in Industrial Engineering, from Istanbul Technical University. His research interests include Meta-heuristics algorithms applied to the combinatorial optimization problem and Multi Criteria Decision Making Methods.

**Musa Karakas** is an MSc student at Computer Engineering Department at Fatih University, Istanbul Turkey. He worked part-time in a research project aiming to develop maintenance planning systems for railway turnouts.

**Ian Jennions** Ian's career spans over 30 years, working mostly for a variety of gas turbine companies. He has a Mechanical Engineering degree and a PhD in CFD both from Imperial College, London. He has worked for Rolls-Royce (twice), General Electric and Alstom in a number of technical roles, gaining experience in aerodynamics, heat transfer, fluid systems, mechanical design, combustion, services and IVHM. He moved to Cranfield in July 2008 as Professor and Director of the newly formed IVHM Centre. The Centre is funded by a number of industrial companies, including Boeing, BAe Systems, Rolls-Royce, Thales, Meggitt, MOD and Alstom Transport. He has led the development and growth of the Centre, in research and education, over the last three years. The Centre offers a

# Bayesian Framework Approach for Prognostic Studies in Electrolytic Capacitor under Thermal Overstress Conditions

Chetan S. Kulkarni<sup>1</sup>, José R. Celaya<sup>2</sup>, Kai Goebel<sup>3</sup>, and Gautam Biswas<sup>4</sup>

<sup>1,4</sup> *Vanderbilt University, Nashville, TN, 37235, USA*  
*chetan.kulkarni@vanderbilt.edu*  
*gautam.biswas@vanderbilt.edu*

<sup>2</sup> *SGT Inc. NASA Ames Research Center, Moffett Field, CA, 94035, USA*  
*jose.r.celaya@nasa.gov*

<sup>3</sup> *NASA Ames Research Center, Moffett Field, CA, 94035, USA*  
*kai.goebel@nasa.gov*

## ABSTRACT

Electrolytic capacitors are used in several applications ranging from power supplies for safety critical avionics equipment to power drivers for electro-mechanical actuators. Past experiences show that capacitors tend to degrade and fail faster when subjected to high electrical or thermal stress conditions during operations. This makes them good candidates for prognostics and health management. Model-based prognostics captures system knowledge in the form of physics-based models of components in order to obtain accurate predictions of end of life based on their current state of health and their anticipated future use and operational conditions. The focus of this paper is on deriving first principles degradation models for thermal stress conditions and implementing Bayesian framework for making remaining useful life predictions. Data collected from simultaneous experiments are used to validate the models. Our overall goal is to derive accurate models of capacitor degradation, and use them to remaining useful life in DC-DC converters.

## 1. INTRODUCTION

Avionics systems play a critical role in many aspects of aircraft flight control. As the system complexity and flight criticality of functions performed by these systems increases, the related consequences of in-flight malfunctions are bound to increase. This drives the need for Integrated Vehicle Health Management (IVHM) technologies for flight-critical avionics. Studying and analyzing the performance degradation of embedded electronics in the aircraft domain is abso-

lutely necessary to increase aircraft reliability, assure in-flight performance, and reduce maintenance costs, (J. R. Celaya, Wysocki, Vashchenko, Saha, & Goebel, 2010; Ferrell, 1999). In addition to this, an understanding of the behavior of deteriorated components is needed as well as the capability to anticipate failures and predict the remaining useful life (RUL) of the electronic systems.

An avionics system module consists of hardware (power supply, Global positioning system (GPS) receiver, Inertial measurement unit (IMU)) and software (GPS software, INAV - integrated navigation solution) components (Kulkarni, Biswas, Bharadwaj, & Kim, 2010). Switched-mode power supplies are widely used in DC-DC converters because of their high efficiency and compact size. Buck-boost DC-DC converter steps voltage levels up/down based on the application requirements, in which electrolytic capacitors and metal oxide semiconductor field effect transistor (MOSFET) switches are known to have the highest degradation and failure rates among all of the components (Goodman, 2005). Degraded capacitors affect the performance and efficiency of the DC-DC converters in a significant manner and also impose a risk on instantiating cascading failures on other connected subsystems.

In this paper we develop an effective prognostics and health management (PHM) methodology that applies to electronic systems and components. In particular, we develop a methodology to enable early detection of failure precursors in capacitor elements associated with DC-DC power supplies. Our approach combines physics-based modeling supported by empirical experimental analysis for improving the degradation models, and then implementing bayesian framework to predict remaining useful life of electrolytic capacitors. Our hy-

Chetan S. Kulkarni et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

pothesis is that early detection will lead to better prediction and end of life estimates by tracking and modeling the degradation process.

The structure of the paper is as follows. Section II discusses the prognostics methodology implemented for this work. Section III presents introduction to electrolytic capacitors and its basic structure, operation and degradation mechanisms. Section IV discusses capacitor first principle models in detail. Section V describes the thermal stress aging experiments conducted for this work. Section VI and VII presents the prognostic framework methodology and RUL results respectively. The paper ends with discussion and conclusion in section VIII.

## 2. PROGNOSTIC METHODOLOGY

Prognostics is the process of predicting health condition and remaining useful life based on current and previous state, current and future operating conditions. Prognostics and health management (PHM) methods combine sensing, data collection, interpretation of environmental, operational, and performance related parameters to indicate systems health as well as anticipate damage propagation due to degradation. PHM methodologies can be implemented through the use of various techniques that study parameter variations, which indicate changes in performance degradation based on usage duration and conditions.

Prognostics is an essential technology for improving system safety, reliability, and availability. Prognostics deals with determining the health state of components, and projecting its health evolution into the future to make end of life (EOL) and remaining useful life (RUL) estimations. Model-based prognostics approaches perform these tasks with the help of first principles based physics models that capture knowledge about the system, its components, and their failures (Daigle & Goebel, 2011; Saha & Goebel, 2009).

We adopt an approach wherein we develop detailed physics-based models of components and systems that include descriptions of how fault parameters evolve in time. The implemented prognostics architecture is as shown in Figure 1. We have implemented this approach on empirical degradation models in our earlier work (J. Celaya et al., 2011a; Kulkarni et al., 2012). Experimental studies are being conducted and first principles based degradation models are being derived using the descriptions mentioned in (Kulkarni, Celaya, et al., 2011; J. Celaya et al., 2011a; Fife, 2006; MIL-C-62F, 2008). Identifying the failure precursors and developing accurate models of degradation/ failure has been the most difficult problem of our research goal. These models depend on known as well as unknown and possibly time-varying wear parameters. Early detection and analysis may lead to better prediction and end of life estimates of the capacitor by tracking and modeling the degradation process. Faults and degra-

dations appear as parameter value changes in the model, and this provides the mechanisms for tracking system behavior under degraded conditions (J. Celaya et al., 2011a, 2011b). The derived state space models are then implemented in a Bayesian framework for prognostics.

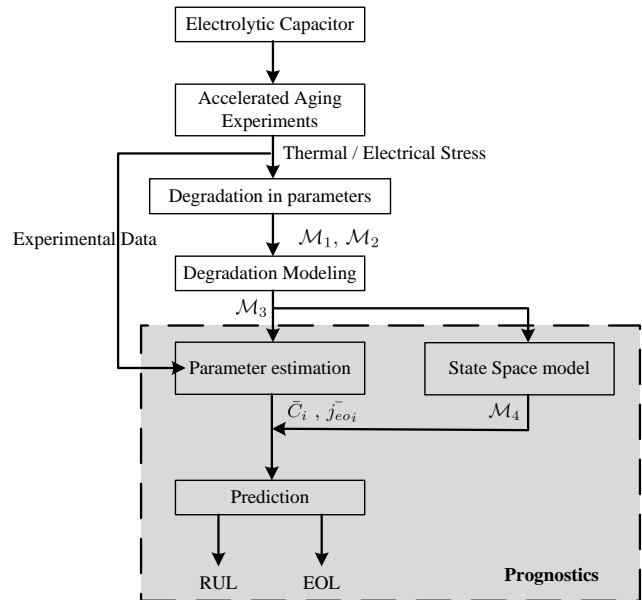


Figure 1. Prognostics Methodology

In the next section we discuss in brief the basics of electrolytic capacitors, their detailed structure and different mechanisms under which the devices degrade.

## 3. ELECTROLYTIC CAPACITORS

An aluminum electrolytic capacitor, as illustrated in Figure 2, consists of a cathode aluminum foil, electrolytic paper, electrolyte, and an aluminum oxide layer on the anode foil surface, which acts as the dielectric. When in contact with the electrolyte, the oxide layer possesses an excellent forward direction insulation property (Bengt, 1995). Figure 3 shows a

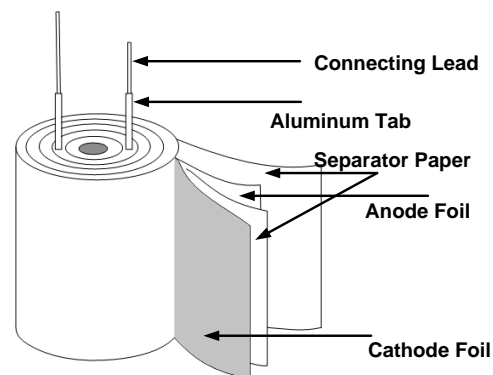


Figure 2. Illustration of an Electrolytic Capacitor

detailed view of the cross section of an electrolytic capacitor structure. To get higher capacitance values for the same surface area of the anode and cathode foils, the foil is etched by a chemical process. Together with magnified effective surface area attained by etching the foil, a high capacitance value is obtained in a small volume (Fife, 2006). Since the oxide layer has rectifying properties, a capacitor has polarity. If both the anode and cathode foils have an oxide layer, the capacitors would be bipolar. In this work, we analyze non-solid aluminum electrolytic capacitors in which the electrolytic paper is impregnated with liquid electrolyte. After etching, the plates are anodized by coating them with a thin aluminum oxide layer on the surface of the foil. This layer of aluminum oxide acts as the dielectric (insulator) and serves to block the flow of direct current between the two capacitor plates (Fife, 2006).

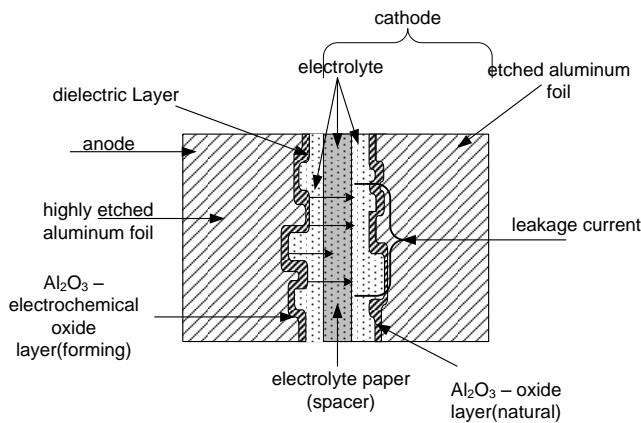


Figure 3. Capacitor Detail Structure

Electrolytic capacitor performance is strongly affected by its operating conditions, such as voltage, current, frequency, and ambient temperatures. Degradation in the capacitor manifests an increase in the equivalent series resistance ( $ESR$ ) and decrease in capacitance ( $C$ ), due to deterioration of electrolyte quality, decreases in electrolyte volume due to evaporation, weakening of the oxide layer, over operating time (Bengt, 1995; Fife, 2006). The  $ESR$  of a capacitor is the sum of the resistance due to aluminum oxide, electrolyte, spacer, and electrodes (foil, tabbing, leads, and ohmic contacts) (Bengt, 1995) and capacitance is the ability of a capacitor to store charge in an electric field. The health of a capacitor is often measured by the values of these two parameters.

There are certain industry standard thresholds for these parameter values, if the measurements exceed these thresholds then the component is considered unhealthy, i.e., the component has reached its end of life, and should be immediately replaced before further operations (Lahyani, Venet, Grellet, & Viverge, 1998). Failures in a capacitor can be one of two types: (1) catastrophic failures, where there is complete loss

of functionality due to a short or open circuit, and (2) degradation failures, where there is gradual deterioration of capacitor due to accumulated damages.

The fishbone diagram in Figure 6 summarizes the most common set of failure modes for electrolytic capacitors that have been discussed in (J. Celaya et al., 2011a; Kulkarni et al., 2012). This diagram identifies the relationship between root causes and failure modes observed in electrolytic capacitors. These root causes can occur individually or combined manner in a capacitor depending upon the conditions of operation. Our focus in this work is on the thermal stressors that govern the capacitor degradation, specifically, we study high temperature scenarios and their effects on the electrolytic capacitor degradation.

### 3.1. Equivalent Electrical Circuits

A simplified electrical lumped parameter model of impedance,  $\mathcal{M}_1$ , defined for an electrolytic capacitor is shown in Figure 4. The  $ESR$  dissipates some of the stored energy in the capacitor. An ideal capacitor would offer no resistance to the flow of current at its leads.

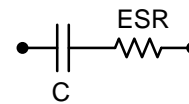


Figure 4. Lumped Parameter Model ( $\mathcal{M}_1$ )

It has been observed that under thermal overstress storage conditions (Bengt, 1995; J. Celaya et al., 2011a), the capacitance ( $C$ ) and  $ESR$  value depends of the electrolyte resistance  $R_E$ . A more detailed lumped parameter model derived for an electrolytic capacitor under thermal overstress condition,  $\mathcal{M}_2$  can be modified from  $\mathcal{M}_1$ , as shown in Figure 5.  $R_1$  is the combined series and parallel resistances in the model.  $R_E$  is the electrolyte resistance. The combined resistance of  $R_1$  and  $R_E$  is the  $ESR$  of the capacitor.  $C$  is the total capacitance of the capacitor as discussed earlier.

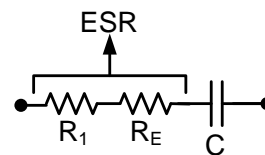


Figure 5. Lumped Parameter Model ( $\mathcal{M}_2$ )

## 4. PHYSICS BASED MODELING

In this section we discuss about deriving the first principles based degradation models for capacitors under thermal overstress conditions. Under thermal overstress conditions since the device was subjected to only high temperature with no charge applied we observe degradation only due to electrolyte

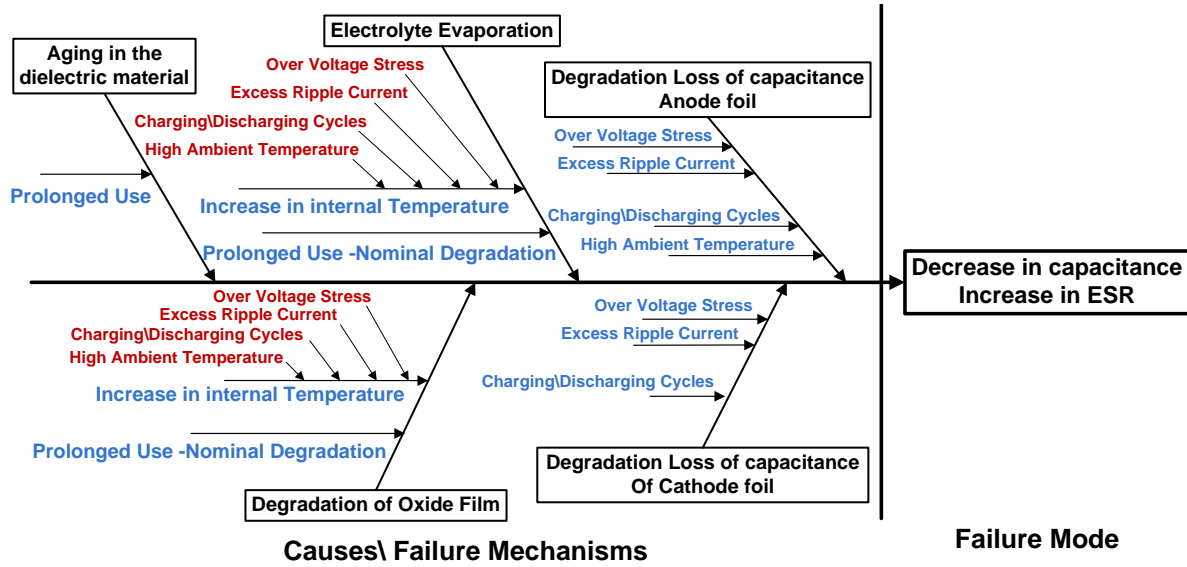


Figure 6. Fishbone diagram of failure mechanisms in aluminum electrolytic capacitor

evaporation. The degradation models are derived based on the underlying physics of operation and measurements from experimental data.

#### 4.1. Structural Model

For deriving the physics based models of an electrolytic capacitor it is also necessary to know about the structural and manufacturing details, since health estimations are done based on the type of electrolyte, volume of electrolyte, oxide layer thickness etc. The models defined use this information for making effective degradation/failure predictions. A detail structural study of the electrolytic capacitor under test is discussed in this section.

During modeling it is not possible to know the exact amount of electrolyte present in a capacitor. But using information from the structural details as shown in Figure 7, we can approximately calculate the amount of electrolyte present. A very highly porous separator paper is used which soaks all the electrolyte. The paper separator is sandwiched between anode and cathode, each having a thickness  $d_S$ ,  $d_A$  and  $d_C$  respectively ( $d_S \approx d_A \approx d_C$ ). Based on the type and configuration, the electrolyte volume will vary which can be updated in the model parameters.

The equation for calculating the approximate electrolyte volume is derived from calculating the total capacitor capsule volume,  $V_c$  given by :

$$V_c = \pi r_c^2 h_c \quad (1)$$

where :

$r_c$  = radius of capacitor capsule

$h_c$  = height of capacitor capsule

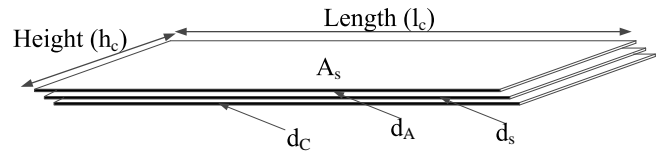


Figure 7. Capacitor open structure

The approximate electrolyte volume,  $V_e$  based on all the other known structural details of the capacitor can expressed as:

$$V_e = \pi r_c^2 h_c - A_s (d_A + d_C) \quad (2)$$

#### 4.2. Capacitance Degradation Model

Exposure to high temperatures,  $T_{applied} > T_{rated}$  results in accelerated aging of capacitors (Kulkarni, Celaya, et al., 2011; J. Celaya et al., 2011a; 60068-1, 1988). Higher ambient storage temperature accelerates the rate of electrolyte evaporation leading to degradation of the capacitance (Kulkarni, Celaya, et al., 2011; Bengt, 1995). The depletion in electrolyte volume,  $V_e$ , (Kulkarni, Biswas, et al., 2011; Rusdi et al., 2005) is given by :

$$V_e(t) = V_o - (A_s j_{eo} w_e) \times t \quad (3)$$

where:

$V_o$  = initial electrolyte volume

$j_{eo}$  = evaporation rate ( $\text{mg min}^{-1} \text{area}^{-1}$ )

$w_e$  = volume of ethyl glycol molecule

$t$  = time in hours.

The total lumped capacitance from first principles of electro-



magnetism is given by :

$$C = (2\epsilon_R\epsilon_0 A_s)/d_C \quad (4)$$

where:

$\epsilon_R$  = relative dielectric constant

$\epsilon_0$  = permittivity of free space

Thus from Eq. (3) and Eq. (4) we can derive the first principles capacitance degradation model,  $\mathcal{M}_3$  given by :

$$\mathcal{M}_3 : C(t) = \left( \frac{2\epsilon_R\epsilon_0}{d_C} \right) \left( \frac{V_0 - V_e(t)}{j_{eo} t w_e} \right) \quad (5)$$

As discussed earlier, increase in the core temperature evaporates the electrolyte thus decreasing the electrolyte volume leading to degradation in capacitance. The resultant decrease in the capacitance can be computed using model,  $\mathcal{M}_3$  wherein the decrease in electrolyte volume, ( $V_e$ ) leads to decrease in capacitance, ( $C$ ).

### 4.3. Dynamic Models

The electrolyte volume,  $V_e$  can also be calculated and can also be expressed as  $V_e = A_s \cdot d_C$ . Hence the oxide surface area,  $A_s$  can be expressed in terms of electrolyte volume,  $V_e$  and oxide thickness,  $d_C$  as:

$$A_s = \frac{V_e}{d_C}. \quad (6)$$

From Eq. (4) and Eq. (6), dynamic capacitor degradation model can be updated as :

$$C_k = \left( \frac{2\epsilon_R\epsilon_0}{d_C} \right) \left( \frac{V_{e(k)}}{d_C} \right) \quad (7)$$

From Eq. (3) the first order discrete approximation for change in electrolyte volume can be expressed as:

$$\begin{aligned} \frac{dV_e}{dt} &= -(w_e A_s j_{eo}), \\ V_{e(k+1)} &= V_{e(k)} + \frac{dV_e}{dt} \Delta t, \\ V_{e(k+1)} &= V_{e(k)} - (w_e A_s j_{eo}) \Delta t. \end{aligned} \quad (8)$$

From Eq. (7) we have,

$$\begin{aligned} V_{e(k)} &= \frac{C_k}{2\epsilon_R\epsilon_0} d_C^2, \\ V_{e(k)} &= (C_k) \alpha. \end{aligned} \quad (9)$$

where:

$$\alpha = \frac{d_C^2}{2\epsilon_R\epsilon_0}$$

From Eq. (9) we can express Eq. (8) as :

$$\begin{aligned} C_{k+1} \alpha &= C_k \alpha + \frac{dC}{dt} \Delta t, \\ C_{k+1} \alpha &= C_k \alpha - (w_e A_s j_{eo}) \Delta t, \text{ hence} \\ C_{k+1} &= C_k - \frac{(w_e A_s j_{eo})}{\alpha} \Delta t. \end{aligned} \quad (10)$$

The complete discrete time dynamic model for capacitance degradation can be summarized as :

$$\mathcal{M}_4 : C_{k+1} = C_k - \frac{(2\epsilon_R\epsilon_0 w_e A_s j_{eo})}{d_C^2} \Delta t \quad (11)$$

Model  $\mathcal{M}_4$ , in Eq. (11) is implemented in a Bayesian tracking framework. In this work we are implementing a unscented Kalman filter (UKF) since the degradation in capacitance (state) due to decrease in electrolyte is considered to be a dynamic linear model and the evaporation rate ( $j_{eo}$ ) parameter, assumed to be varying and estimated online. Next we discuss the implementation of the Bayesian framework methodology for prognostics (Saha & Goebel, 2009; Daigle et al., 2012; Daigle & Goebel, 2011).

## 5. UNSCENTED KALMAN FILTER

Estimation in nonlinear system is very important because many practical systems involve nonlinearities in their operation through one form or another. Estimation of the state accurately of such non-linear system is very important to diagnostics and be further implemented to prognostic applications which is extremely difficult. The Extended Kalman Filter (EKF) which applies the KF to nonlinear system by linearizing all nonlinear models, has become a most widely used method for estimation of nonlinear system. Although the EKF maintains is computationally efficient recursive update form of the KF for non-linear systems, it suffers a number of serious limitations (Julier & Uhlmann, 1997, 2004).

1. Only reliable if the error propagation to the future states can be approximated by a linear function.
2. Linearization can be applied only if the Jacobian matrix exists i.e., cannot be implemented if the system has discontinuities.
3. Calculating Jacobian matrices can be a very difficult and error-prone process.

The Unscented Kalman Filter (UKF) was proposed by (Julier & Uhlmann, 1997, 2004) to overcome these problems in nonlinear systems. The unscented Kalman filter, instead of approximating the nonlinearity, approximates the state distribution (Julier & Uhlmann, 1997, 2004). This procedure maintains the nonlinear functions exactly, eliminating the need to calculate Jacobian's, and thereby providing an easier implementation framework. In this section we will look in the

detail framework of UKF which has been implemented for prognostics in this work.

A nonlinear system, described by the difference equation and the observation model with additive noise is given as :

$$\begin{aligned} x_k &= \mathbf{F}[x_{k-1}, u_{k-1}, k] + w_{k-1} \\ z_k &= \mathbf{H}[x_k, k] + v_k \end{aligned} \quad (12)$$

where  $x(k)$  is the  $n$ -dimensional state of the system at time step  $k$ ,  $u(k)$  is the input vector,  $w(k)$  is the process noise,  $Q$ ,  $z(k)$  is the observation vector and  $v_k$  is the measurement noise,  $R$ .

To solve the problem of predicting the future state or observation based on the Unscented transform (UT), UT takes a random variable  $x$ , with mean  $\bar{x}$  and covariance  $\mathbf{P}_{xx}$ , which is related to a second random variable  $y$  by a nonlinear function  $y = f(x)$ . A small set of points (*sigma points*) with mean  $\bar{y}$  and covariance  $\mathbf{P}_{xx}$  are selected (Julier & Uhlmann, 1997), which are deterministically selected and weighted to exhibit properties to match the mean and covariance of the original distribution. A non-linear transformation is applied to each point to get the transformed points, statistics of the transformed points is then calculated to estimate the mean and covariance of the transformed points. The sigma point weights do not directly represent probabilities of the sigma points, and hence do not have to lie in the interval  $[0, 1]$ . The weights  $W_i$  can be positive or negative, but need to obey the following condition to provide an unbiased estimate.

$$\sum_i W_i = 1 \quad (13)$$

Each sigma point is instantiated through the function( $f$ ) to obtain new set of sigma points  $\mathcal{Y}$ .

$$\mathcal{Y}_i = f(\mathcal{X}_i) \quad (14)$$

The mean of the transformed points is given by:

$$\bar{y} = \sum_i W_i \mathcal{Y}_i \quad (15)$$

The covariance of the transformed points is given by:

$$P_{yy} = \sum_i W_i (x_o - \mu_0)(x_o - \mu_0)^T \quad (16)$$

The basic idea of the unscented transform is that it is easier to approximate a probability distribution  $x$  than it is to approximate an arbitrary nonlinear function  $f$  or transformation (Julier & Uhlmann, 2004). This basic principle is implemented in the unscented Kalman filter where the unscented transform is exploited for nonlinear state estimation (Julier &

Uhlmann, 1997, 2004). At each step, the unscented transform is applied to the state estimate and is used for a single step prediction. In contrast, here, we apply the transform to the state parameter distribution at given single time point  $k_P$ , and use this for multi-step predictions to EOL. There are several methods which exists for selecting the sigma points out of which we implement the symmetric unscented transform for the prognosis problem (Daigle et al., 2012; Julier & Uhlmann, 2004). Detailed results will be presented in Section 6 for capacitor degradation problem.

In the symmetric unscented transform,  $2n_x + 1$  sigma points are selected symmetrically about the mean as follows (Julier & Uhlmann, 2004):

$$\begin{aligned} \mathcal{X}_0 &= \bar{x} \\ W_0 &= \frac{k}{n+k} \\ \mathcal{X}_i &= \bar{x} + \sqrt{((n+k)P_{xx})_i}, \\ W_i &= \frac{k}{2(n+k)} \\ \mathcal{X}_{i+n} &= \bar{x} - \sqrt{((n+k)P_{xx})_i} \\ W_{i+n} &= \frac{k}{2(n+k)} \end{aligned} \quad (17)$$

where  $\sqrt{((n+k)P_{xx})_i}$  refers to the  $i^{th}$  column of the matrix square root of  $(n+k)P_{xx}$ , computed using the Cholesky decomposition since it is numerically efficient and stable. Parameter  $k$  is used to tune the higher moments of distribution and suggested to have a smaller values as possible to bring the sigma points closer together. If  $x$  is assumed Gaussian, then selecting  $n+k=3$  is recommended (Julier and Uhlmann, 1997).

## 6. THERMAL OVERSTRESS EXPERIMENT

In this setup we emulated conditions similar to high temperature storage conditions (Kulkarni, Celaya, et al., 2011; Kulkarni, Biswas, et al., 2011), where capacitors were placed in a controlled chamber and the temperature raised above their rated specification (60068-1, 1988). Pristine capacitors were taken from the same lot rated for 10V and maximum storage temperature rating of 85°C.

Experiments were conducted with 2200  $\mu\text{F}$  capacitors with TOS temperature at 105°C and humidity factor at 3.4%. The chamber temperature was gradually increased in steps of 25°C till the pre-determined temperature limit was reached. The capacitors were allowed to settle at a set temperature for 15 min and then the next step increase was applied. This process was continued till the required temperature limit was attained. To decrease possibility of shocks due to sudden de-

crease in the temperature the above procedure was followed. At the end of specific time interval the temperature was lowered in steps of 25°C till the required room temperature was reached.

Before being characterized the capacitors were kept at room temperature for 15 min. The ESR value is the real impedance measured through the terminal software of the instrument. Similarly the capacitance value is computed from the imaginary impedance using Electrochemical Impedance Spectroscopy (EIS). Characterization of all the capacitors was done for measuring the impedance values using an SP-150 Biologic impedance measurement instrument (Biologic, 2010). Figure 8 shows the plots decrease in capacitance due to accelerated aging for all the 15 capacitors under test at different aging times.

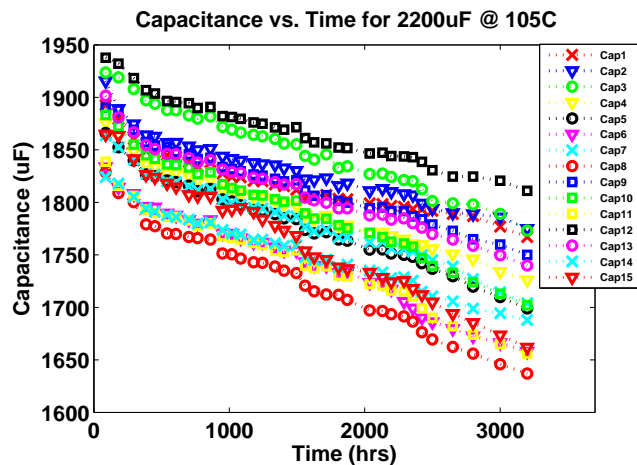


Figure 8. Capacitance Plot for all the devices under TOS

In the thermal overstress experiments, the capacitors we characterized periodically and after 3400 hours of operation it was observed that the average capacitance ( $C$ ) value decreased by more than 9-11% while decrease in  $ESR$  value was observed around 20 - 22%. From literature (60068-1, 1988) under thermal overstress conditons higher capaitance degradation is observed and minor degradation in  $ESR$  which correlated with the data collected. The failure thresholds under storage conditions for capacitance ( $C$ ) is 10% while that for  $ESR$  is around 280- 300% of the pristene condition values (60384-4-1, 2007; Kulkarni, Celaya, et al., 2011). Hence the capacitance degradation data was used as a precursor to failure parameter to estimate the current health condition of the device.

### 7. PREDICTION OF REMAINING USEFUL LIFE RESULTS

State estimation and RUL estimation results are discussed for capacitor Cap # 5 out of a batch of 15 available capacitors under test. Figure 9 shows the result of filter tracking for degradation in capacitance upto 3200 hours of aging time.

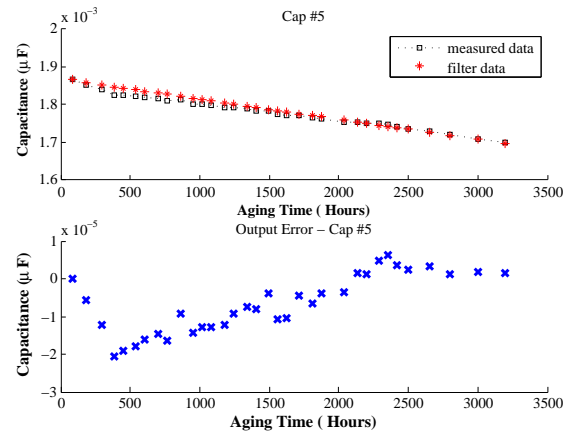


Figure 9. (a) Tracking filter output against measurement data, (b) Output error for Cap # 5

Figure 10 presents results from the remaining useful life prediction algorithm at different aging times  $t_p = 87, 607, 1495, 2131, 2800$  (hrs), at which the capacitors are characterized and their capacitance ( $C$ ) value is calculated. The failure threshold is considered to be 10% decrease in capacitance value, which in this case is at 3200 hours of aging time. End of life (EOL) is defined as the time at which the forecasted capacitance value trajectory crosses the EOL threshold. Therefore, RUL is EOL minus aging times  $t_p = 87.5, 607, 1495, 2131, 2800$  (hrs).

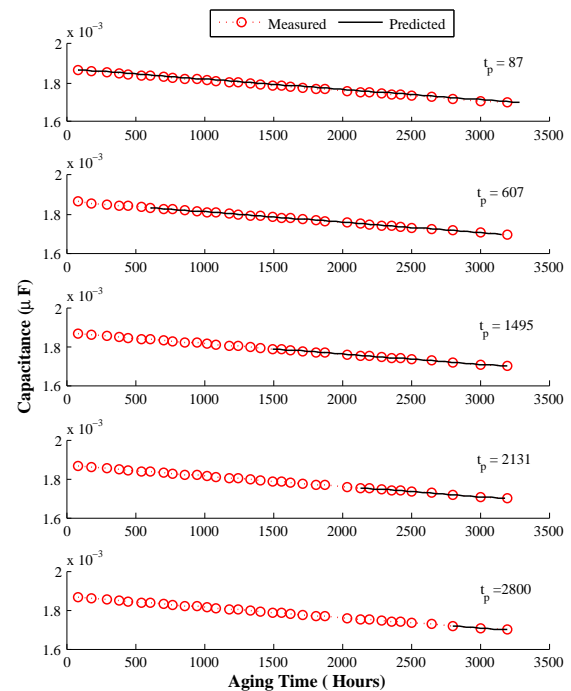


Figure 10. Capacitance decrease prediction at different Aging Time for Cap # 5

An  $\alpha$ - $\lambda$  prognostics performance metric (Saxena et al., 2009, 2008) is presented in Figure 11 for test case of Cap #5. The central dashed line represents ground truth and the shaded region is corresponding to a 30% ( $\alpha = 0.3$ ) error bound in the RUL prediction. Performance metric identifies whether the algorithm performs within desired error margins (specified by the parameter  $\alpha$ ) of the actual RUL at any given time instant (specified by the parameter  $\lambda$ ) (Saxena et al., 2009) and is based on relative accuracy (RA) metric in Eq. (18).

$$RA = 100 \left( 1 - \frac{RUL^* - RUL'}{RUL^*} \right) \quad (18)$$

Table 1 shows the performance summary based on the RA of all the capacitors under thermal stress performance. These metrics allows for an assessment of the percentage accuracy relative to the ground-truth value. RA values of 100 represent perfect accuracy. The RA is presented for all the test cases for different prediction times. The last column of Table 1 represents the median RA of all the test cases for a particular prediction time. It must be noted that if the prediction error magnitude grows beyond 100% RA gives a negative value. We do not consider such cases since these cases would not have qualify the tests for calculating RA (Saxena et al., 2009), these are indicated by NA in Table 1.

From the  $\alpha$ - $\lambda$  metric plot in Figure 11 it can be observed that the relative accuracy is not as good at the end but the accuracy is good enough under acceptable limits. This is due to the non-linearity observed in the data at the end of the aging time and the limitation of the model due to not including the oxide layer breakdown. The residuals show an increased error with aging time, since the breakdown in the oxide layer observed due to stress is not considered for this model which starts to dominate in the later stages of aging of the device. This breakdown is exponential in nature and as we can observe a dip in the capacitance values from the linear path in the later stages.

## 8. CONCLUSION AND DISCUSSION

This paper presents a first principles based degradation electrolytic capacitor model and an parameter estimation algorithm to validate the derived model, based on the experimental data. The majors contributions of the work presented in this paper are:

1. Development of the first principles degradation model based on accelerated life test aging data which includes decrease in capacitance as a function of time and evaporation rate linked to temperature conditions;
2. Implementation of a Bayesian based health state tracking and remaining useful life prediction (RUL) algorithm based on the UKF filtering framework;
3. Prediction of remaining useful life for capacitors based

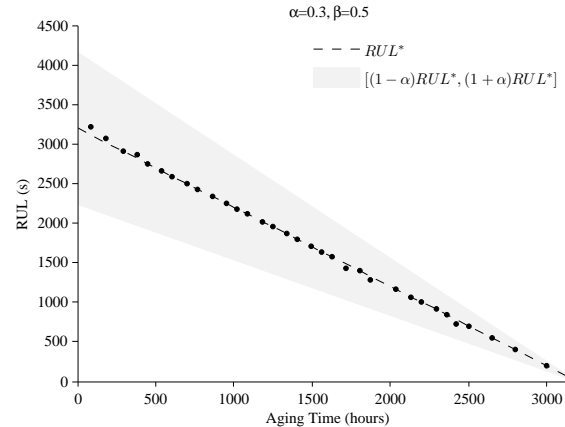


Figure 11. Performance based on Alpha-Lambda metric for Cap#5

first principles degradation model,  $\mathcal{M}_3$ ;

The degradation model,  $\mathcal{M}_3$  based on the first principles gives an indication of how a specific device degrades based on its structure, material properties, operating conditions, etc. The results presented here are based on accelerated aging experimental data and on the accelerated life timescale. In our earlier work we studied the degradation models based on the observed data, and the work discussed here is a next step to generalize the model. Though as discussed in section 4, as a first step a dynamic linear model has been implemented for degradation model. This degradation model for decrease in capacitance,  $C$  and varying evaporation rate  $j_{eo}$  needs to be updated and include the model of break-down in the oxide layer which is exponential in nature and dominates in the later stages of aging. Further research will focus on development of functional mappings that will translate the accelerated life timescale into real usage conditions timescale, where the degradation process dynamics will be slower, and subjected to varying stress conditions.

The performance of the proposed first principles degradation model,  $\mathcal{M}_3$  is acceptable for the current study based on the quality of the model fit to the experimental data and the RUL prediction performance from  $\alpha$ - $\lambda$  metric plot. Additional experiments are currently underway to increase the number of test samples. This will greatly enhance the quality of the model, and guide the exploration of additional degradation-models, where the loading conditions and the environmental conditions are also accounted for towards degradation dynamics.

## ACKNOWLEDGEMENT

Funding for this work was provided by PDM and VVFCs elements of the NASA/ARMD/Avsafe System-wide Safety and Assurance Technologies (SSAT) Project. Authors would also like to acknowledge Dr. Matthew J. Daigle and the members

of Prognostics Center of Excellence (PCoE) at NASA Ames Research Center for engaging in valuable and insightful discussions.

#### NOMENCLATURE

$\epsilon_R$	relative dielectric constant
$\epsilon_O$	permittivity of free space
$V_o$	initial electrolyte volume
$j_{eo}$	evaporation rate ( $\text{mg min}^{-1} \text{ area}^{-1}$ )
$\rho_E$	electrolyte resistivity
$A_s$	effective oxide surface area
$w_e$	volume of ethyl glycol molecule
$d_A$	thickness of anode strip,
$d_C$	thickness of cathode strip
$d_s$	thickness of paper spacer
$\mathcal{M}_1$	electrical lumped parameter model
$\mathcal{M}_2$	updated lumped parameter model
$\mathcal{M}_3$	capacitance degradation model
$\mathcal{M}_4$	capacitance discrete time model

#### REFERENCES

- 60068-1, I. (1988). Environmental testing, part 1: General and guidance. *IEC Standards*.
- 60384-4-1, I. (2007). Fixed capacitors for use in electronic equipment. *IEC Standards*.
- Bengt, A. (1995). Electrolytic capacitors theory and applications. *RIFA Electrolytic Capacitors*.
- Biologic. (2010). Application note 14-zfit and equivalent electrical circuits [Computer software manual].
- Celaya, J., Kulkarni, C., Biswas, G., & Goebel, K. (2011a). A model-based prognostics methodology for electrolytic capacitors based on electrical overstress accelerated aging. *Proceedings of Annual Conference of the PHM Society, September 25-29, Montreal, Canada*.
- Celaya, J., Kulkarni, C., Biswas, G., & Goebel, K. (2011b). Towards prognostic of electrolytic capacitors. *American Institute of Aeronautics and Astronautics, AIAA Infotech@Aerospace 2011, March 2011, St. Louis, Missouri*.
- Celaya, J. R., Wysocki, P., Vashchenko, V., Saha, S., & Goebel, K. (2010). Accelerated aging system for prognostics of power semiconductor devices. In *Ieee autotestcon, 2010* (p. 1-6). Orlando, FL.
- Daigle, M., & Goebel, K. (2011, March). Multiple damage progression paths in model-based prognostics. *Proceedings of the 2011 IEEE Aerospace Conference*.
- Daigle, M., Saha, B., & Goebel, K. (2012, March). A comparison of filter-based approaches for model-based prognostics. In *Proceedings of the 2012 IEEE aerospace conference*.
- Ferrell, B. L. (1999). JSF prognostics and health management. *IEEE Aerospace Conference*, 471.
- Fife, J. (2006, August). *Wet electrolytic capacitors* (Patent No: 7,099 No. 1). Myrtle Beach, SC: AVX Corporation.
- Goodman, D. (2005, March). Practical application of phm/prognostics to cots power converters. *Aerospace Conference, 2005 IEEE*, 3573-3578.
- Julier, S. J., & Uhlmann, J. K. (1997). A new extension of the kalman filter to nonlinear systems. *Proceedings of the 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls*, 182 - 193.
- Julier, S. J., & Uhlmann, J. K. (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3), 401 - 422.
- Kulkarni, C., Biswas, G., Bharadwaj, R., & Kim, K. (2010, April). Effects of degradation in dc-dc converters on avionics systems: A model based approach. *Proceedings of the Machinery Failure Prevention Technology Conference (MFPT)*, 8-13.
- Kulkarni, C., Biswas, G., Celaya, J., & Goebel, K. (2011). A case study for capacitor prognostics under accelerated degradation. *IEEE 2011 Workshop on Accelerated Stress Testing & Reliability (ASTR), September 28-30, San Francisco, CA*.
- Kulkarni, C., Celaya, J., Biswas, G., & Goebel, K. (2011). Prognostic modeling and experimental techniques for electrolytic capacitor health monitoring. *The 8th International Workshop on Structural Health Monitoring 2011 (IWSHM), September 13-15, Stanford University, Stanford, CA*.
- Kulkarni, C., Celaya, J., Biswas, G., & Goebel, K. (2012). Prognostic and experimental techniques for electrolytic capacitor health monitoring. *The Annual Reliability and Maintainability Symposium (RAMS), January 23-36, Reno, Nevada*.
- Lahyani, A., Venet, P., Grellet, G., & Viverge, P. (1998, Nov). Failure prediction of electrolytic capacitors during operation of a switchmode power supply. *IEEE Transactions on Power Electronics*, 13, 1199-1207.
- MIL-C-62F. (2008). General specification for capacitors. *Fixed Electrolytic*.
- Rusdi, M., Moroi, Y., Nakahara, H., & Shibata, O. (2005). Evaporation from water ethylene glycol liquid mixture. *Langmuir - American Chemical Society*, 21 (16), 7308 - 7310.
- Saha, B., & Goebel, K. (2009, September). Modeling li-ion battery capacity depletion in a particle filtering framework. In *Proceedings of the annual conference of the prognostics and health management society 2009*.
- Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., & Schwabacher, M. (2008). Metrics for evaluating performance of prognostic techniques. In *International conference on prognostics and health management 2008*.
- Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K.

(2009, September). On applying the prognostic performance metrics. In *Annual conference of the prognostics and health management society*.

**Chetan S. Kulkarni** is a Research Assistant at ISIS, Vanderbilt University. He received the M.S. degree in EECS from Vanderbilt University, Nashville, TN, in 2009, where he is currently a Ph.D candidate and received a B. E. degree in Electronics and Electrical Engineering from University of Pune, India in 2002.

**José R. Celaya** is a research scientist with SGT Inc. at the Prognostics Center of Excellence, NASA Ames Research Center. He received a Ph.D. degree in Decision Sciences and Engineering Systems in 2008, a M. E. degree in Operations Research and Statistics in 2008, a M. S. degree in Electrical Engineering in 2003, all from Rensselaer Polytechnic Institute, Troy New York; and a B. S. in Cybernetics Engineering in 2001 from CETYS University, México.

**Kai Goebel** received the degree of Diplom-Ingenieur from the Technische Universitt Mnchen, Germany in 1990. He received the M.S. and Ph.D. from the University of California at Berkeley in 1993 and 1996, respectively. Dr. Goebel is a senior scientist at NASA Ames Research Center where he leads the Diagnostics and Prognostics groups in the Intelligent Systems division. In addition, he directs the Prognostics Center of Excellence and he is the technical lead for Prognostics and Decision Making of NASAs System-wide Safety and Assurance Technologies Program. He worked at General Electrics Corporate Research Center in Niskayuna, NY from 1997 to 2006 as a senior research scientist. He has carried out applied research in the areas of artificial intelligence, soft computing, and information fusion. His research interest lies in advancing these techniques for real time monitoring, diagnostics, and prognostics. He holds 15 patents and has published more than 200 papers in the area of systems health management.

**Gautam Biswas** received the Ph.D. degree in computer science from Michigan State University, East Lansing. He is a Professor of Computer Science and Computer Engineering in the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN.

Prediction Time	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
181.67	98.34	95.03	100.00	98.34	98.34	98.34	98.34	93.37	98.34	96.69	91.72	91.72	98.34	98.34	96.69
295.38	98.28	94.84	98.28	100.00	100.00	98.28	98.28	93.11	98.28	96.56	93.11	91.39	98.28	100.00	96.56
384.47	98.22	94.67	100.00	100.00	98.22	98.22	98.22	92.90	98.22	96.45	91.12	91.12	98.22	98.22	96.45
450.93	98.18	94.54	98.18	100.00	100.00	100.00	98.18	92.72	98.18	96.36	92.72	90.91	98.18	100.00	98.18
540.77	98.12	94.36	100.00	100.00	100.00	98.12	98.12	92.48	98.12	96.24	92.48	90.60	98.12	100.00	96.24
607.07	98.07	94.22	98.07	100.00	100.00	100.00	98.07	92.29	98.07	96.14	92.29	90.36	98.07	100.00	98.07
701.62	98.00	94.00	98.00	100.00	100.00	100.00	98.00	91.99	98.00	96.00	91.99	89.99	98.00	100.00	98.00
766.83	97.95	91.78	97.95	100.00	100.00	100.00	100.00	93.84	97.95	93.84	91.78	89.73	97.95	100.00	97.95
860.43	97.86	93.59	97.86	100.00	100.00	100.00	97.86	93.59	97.86	95.73	91.45	89.31	97.86	100.00	97.86
950.07	97.78	93.33	97.78	100.00	100.00	100.00	97.78	93.33	97.78	95.56	91.11	88.89	97.78	100.00	97.78
1019	100.00	90.83	97.71	100.00	100.00	100.00	100.00	93.12	97.71	93.12	93.12	88.54	95.41	100.00	97.71
1084.47	97.64	92.91	100.00	100.00	100.00	100.00	97.64	92.91	97.64	95.27	90.55	90.55	97.64	100.00	97.64
1179.5	97.53	92.58	100.00	100.00	100.00	100.00	97.53	92.58	97.53	95.05	90.10	90.10	97.53	100.00	97.53
1244.82	97.44	92.33	100.00	100.00	100.00	100.00	97.44	92.33	97.44	94.89	92.33	89.77	97.44	100.00	97.44
1338.18	97.31	91.94	100.00	100.00	100.00	100.00	97.31	94.63	97.31	94.63	91.94	89.26	97.31	100.00	97.31
1404.48	97.22	91.65	100.00	100.00	100.00	97.22	100.00	94.43	97.22	94.43	91.65	88.86	97.22	100.00	97.22
1495.4	97.07	91.20	100.00	100.00	100.00	97.07	97.07	94.13	97.07	94.13	91.20	88.27	97.07	100.00	97.07
1560.48	96.95	90.85	100.00	100.00	100.00	96.95	100.00	96.95	96.95	90.85	93.90	87.80	96.95	96.95	100.00
1626.53	96.82	93.64	96.82	100.00	100.00	96.82	96.82	96.82	100.00	93.64	90.47	90.47	96.82	100.00	96.82
1716.57	96.63	89.89	100.00	100.00	96.63	93.26	100.00	100.00	96.63	89.89	93.26	86.52	96.63	96.63	100.00
1807.02	96.41	89.23	96.41	100.00	100.00	92.82	100.00	100.00	96.41	92.82	96.41	89.23	96.41	96.41	100.00
1871.62	96.24	88.71	96.24	100.00	96.24	92.47	100.00	96.24	96.24	88.71	96.24	88.71	96.24	96.24	96.24
1936.88	91.40	91.40	91.40	100.00	100.00	91.40	95.70	95.70	100.00	91.40	95.70	91.40	100.00	95.70	100.00
2013.35	90.64	95.32	90.64	95.32	100.00	90.64	95.32	95.32	95.32	90.64	95.32	90.64	100.00	95.32	95.32
2196.1	90.04	95.02	90.04	100.00	100.00	90.04	95.02	90.04	95.02	90.04	100.00	95.02	100.00	95.02	95.02
2290.12	83.51	94.50	83.51	94.50	100.00	89.01	94.50	83.51	94.50	89.01	100.00	94.50	100.00	94.50	94.50
2355.97	82.23	94.08	82.23	94.08	100.00	82.23	94.08	82.23	94.08	88.15	100.00	100.00	100.00	94.08	88.15
2421.92	80.72	100.00	80.72	100.00	93.57	74.30	100.00	74.30	93.57	87.15	93.57	100.00	100.00	87.15	87.15
2500	71.43	100.00	71.43	92.86	100.00	71.43	92.86	64.29	85.71	85.71	92.86	92.86	92.86	85.71	85.71
2650	54.55	90.91	54.55	90.91	100.00	63.64	90.91	54.55	72.73	90.91	81.82	81.82	90.91	81.82	72.73
2800	37.50	75.00	37.50	75.00	100.00	37.50	87.50	25.00	62.50	87.50	75.00	62.50	75.00	75.00	62.50
3000	NA	25.00	NA	50.00	100.00	0.00	75.00	NA	0.00	75.00	25.00	NA	50.00	50.00	25.00
$\overline{RA}_b$	87.13	90.78	87.99	96.60	99.39	89.29	94.45	85.73	92.14	92.50	90.32	87.15	95.49	94.97	92.38

Figure 1. Summary of RUL forecasting results

$\overline{RA}_b$  is the mean relative accuracy of all capacitors at each prediction time ( $t_p$ )

$\overline{RA}_i$  is the mean relative accuracy of each capacitor at all prediction times

# Real-Time Corrosion Monitoring of Aircraft Structures with Prognostic Applications

Douglas Brown<sup>1</sup>, Duane Darr<sup>2</sup>, Jefferey Morse<sup>3</sup>, and Bernard Laskowski<sup>4</sup>

<sup>1,2,3,4</sup> *Analom, Inc., 562 E. Weddell Dr. Suite 4, Sunnyvale, CA 94089-2108, USA*

*Doug.Brown@analom.com*

*Duane.Darr@analom.com*

*Jeff.Morse@analom.com*

*Bernard.Laskowski@analom.com*

## ABSTRACT

This paper presents the theory and experimental validation of a Structural Health Management (SHM) system for monitoring corrosion. Corrosion measurements are acquired using a micro-sized Linear Polarization Resistance ( $\mu$ LPR) sensor. The  $\mu$ LPR sensor is based on conventional macro-sized Linear Polarization Resistance (LPR) sensors with the additional benefit of a reduced form factor making it a viable and economical candidate for remote corrosion monitoring of high value structures, such as buildings, bridges, or aircraft.

A series of experiments were conducted to evaluate the  $\mu$ LPR sensor for AA 7075-T6, a common alloy used in aircraft structures. Twelve corrosion coupons were placed alongside twenty-four  $\mu$ LPR sensors in a series of accelerated tests. LPR measurements were sampled once per minute and converted to a corrosion rate using the algorithms presented in this paper. At the end of the experiment, pit-depth due to corrosion was computed from each  $\mu$ LPR sensor and compared with the control coupons.

The paper concludes with a feasibility study for the  $\mu$ LPR sensor in prognostic applications. Simultaneous evaluation of twenty-four  $\mu$ LPR sensors provided a stochastic data set appropriate for prognostics. RUL estimates were computed a-posteriori for three separate failure thresholds. The results demonstrate the effectiveness of the sensor as an efficient and practical approach to measuring pit-depth for aircraft structures, such as AA 7075-T6, and provide feasibility for its use in prognostic applications.

## 1. INTRODUCTION

Recent studies have exposed the generally poor state of our nation's critical infrastructure systems that has resulted from

Douglas Brown et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

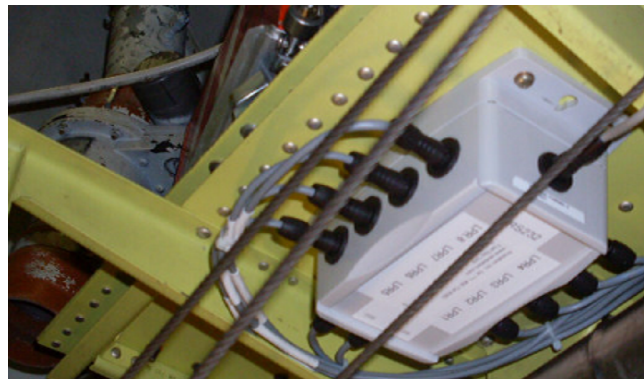


Figure 1. Embedded SHM system installed in the rear fuel-bay bulkhead of a commercial aircraft.

wear and tear under excessive operational loads and environmental conditions. SHM (Structural Health Monitoring) Systems aim at reducing the cost of maintaining high value structures by moving from SBM (Scheduled Based Maintenance) to CBM (Condition Based Maintenance) schemes (Huston, 2010). These systems must be low-cost, simple to install with a user interface designed to be easy to operate. To reduce the cost and complexity of such a system a generic interface node that uses low-powered wireless communications has been developed. This node can communicate with a myriad of common sensors used in SHM. In this manner a structure such as a bridge, aircraft or ship can be fitted with sensors in any desired or designated location and format without the need for communications and power lines that are inherently expensive and complex to route. Data from these nodes is transmitted to a central communications Personal Computer (PC) for data analysis. An example of this is provided in Figure 1 showing an embedded SHM system installed in the rear fuel-bay bulkhead of a commercial aircraft.

Corrosion sensors can be distinguished by the following categories, *direct* or *indirect* and *intrusive* or *non-intrusive*. Direct corrosion monitoring measures a response signal, such



as a current or voltage, as direct result of corrosion. Examples of common direct corrosion monitoring techniques are: corrosion coupons, Electrical Resistance (ER), Electro-Impedance Spectroscopy (EIS) and Linear Polarization Resistance (LPR) techniques. Whereas, indirect corrosion monitoring techniques measure an outcome of the corrosion process. Two of the most common indirect techniques are ultrasonic testing and radiography. An intrusive measurement requires access to the structure. Corrosion coupons, ER, EIS and LPR probes are intrusive since they have to access the structure. Non-intrusive techniques include ultrasonic testing and radiography.

Each of these methods have advantages and disadvantages. Corrosion coupons provide the most reliable physical evidence possible. Unfortunately, coupons usually require significant time in terms of labor and they provide time averaged data that can not be utilized for real time or on-line corrosion monitoring (Harris, Mishon, & Hebbroon, 2006). ER probes provide a basic measurement of metal loss, but unlike coupons, the value of metal loss can be measured at any time, as frequently as required, while the probe is in-situ and permanently exposed to the structure. The disadvantage is ER probes require calibration with material properties of the structure to be monitored. The advantage of the LPR technique is that the measurement of corrosion rate is made instantaneously. This is a more powerful tool than either coupons or ER where the fundamental measurement is metal loss and where some period of exposure is required to determine corrosion rate. The disadvantage to the LPR technique is that it can only be successfully performed in relatively clean aqueous electrolytic environments (*Introduction to Corrosion Monitoring*, 2012). EIS is a very powerful technique that provides both kinetic (corrosion rate) and mechanistic information. The main disadvantages associated with the use of EIS are that the instrumentation is sophisticated and sometimes difficult to use in the field due to the length of time required for each frequency sweep. Additionally, interpretation of the data can be difficult (Buchheit, Hinkebein, Maestas, & Montes, 1998). Finally, ultrasonic testing and radiography can be used to detect and measure (depth) of corrosion through non-destructive and non-intrusive means (Twomey, 1997). The disadvantage with the ultrasonic testing and radiography equipment is the same with corrosion coupons, both require significant time in terms of labor and can not be utilized for real time or on-line corrosion monitoring.

The  $\mu$ LPR presented in this paper improves on existing LPR technology by reducing the form-factor of the sensor to the size of a United States postage stamp, shown in Figure 2, to enable real-time sensing of remote and hard-to-access areas where conventional sensors cannot due to limitations in size and form-factor. Further improvements are realized by narrowing the separation distance between electrodes, which

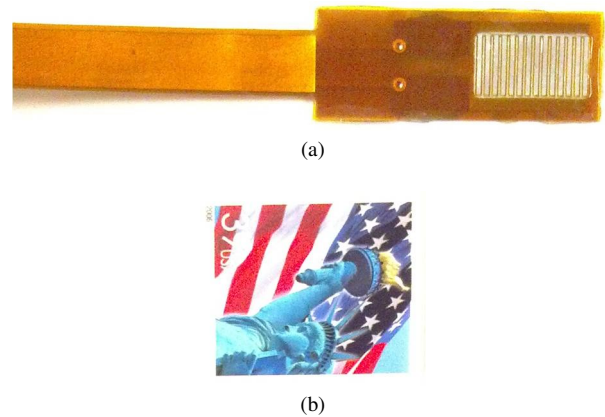


Figure 2. One-to-one scaled comparison of (a) the  $\mu$ LPR sensor attached to a flexcable and (b) a United States postage stamp.

minimizing the effects due to solution resistance. This enables the  $\mu$ LPR to operate outside a controlled aqueous environment, such as an electro-chemical cell, in a broad range of fielded applications (eg. civil engineering, aerospace, petrochemical).

The remainder of the paper is organized by the following. Section 2 describes the general theory governing LPR. Section 3 presents the  $\mu$ LPR discussing the benefits of miniaturizing the sensor from a macro-scaled LPR. Section 4 outlines the experimental setup and procedure used to validate the  $\mu$ LPR sensor. Section 5 presents the experimental measurements with the accompanying analysis which demonstrates the effectiveness of the  $\mu$ LPR sensor. Section 6 evaluates the feasibility for using the  $\mu$ LPR sensor in prognostic applications. Finally, the paper is concluded in Section 7 with a summary of the findings and future work.

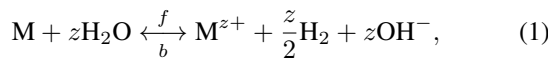
## 2. LPR THEORY

The corrosion of metals takes place when the metal dissolves due to oxidation and reduction (electrochemical) reactions at the interface of metal and the (aqueous) electrolyte solution. Atmospheric water vapor is an example of an electrolyte that corrodes exposed metal surface and wet concrete is another example of an electrolyte that can cause corrosion of reinforced rods in bridges. Corrosion usually proceeds through a combination of electrochemical reactions; (1) anodic (oxidation) reactions involving dissolution of metals in the electrolyte and release of electrons, and (2) cathodic (reduction) reactions involving gain of electrons by the electrolyte species like atmospheric oxygen  $O_2$ , moisture  $H_2O$ , or  $H^+$  ions in an acid (Bockris, Reddy, & Gambola-Aldeco, 2000). The flow of electrons from the anodic reaction sites to the cathodic reaction sites constitutes corrosion current and is used to estimate the corrosion rate. When the two reactions are in equilibrium at the equilibrium corrosion poten-

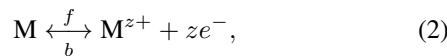
tial,  $E_{corr}$ , the net current on the metal surface is zero without an external source of current. The anodic reactions proceed more rapidly at more positive potentials and the cathodic reactions proceed more rapidly at more negative potentials. Since the corrosion current from the unstable anodic and cathodic sites is too small to measure, an external activation potential is applied across the metal surface and the current is measured for electrochemical calculations. The resulting  $E_a$  vs.  $I_a$  curve is called the polarization curve. Under external activation potential, the anodic and cathodic currents increase exponentially and so when  $\log_{10} I_a$  is plotted against  $E_a$  (a Tafel Plot), the linear regions on the anodic and cathodic curves correspond to regions where either the anodic or cathodic reactions dominate and represent the rate of the electrochemical process. The extrapolation of the Tafel linear regions to the corrosion potential gives the corrosion current,  $I_{corr}$ , which is then used to calculate the rate of corrosion (Burstein, 2005).

### 2.1. Anodic and Cathodic Reactions

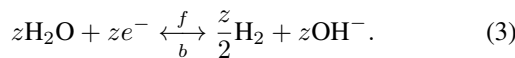
Electrochemical technique of Linear Polarization Resistance (LPR) is used to study corrosion processes since the corrosion reactions are electrochemical reactions occurring on the metal surface. Modern corrosion studies are based on the concept of mixed potential theory postulated by Wagner and Traud, which states that the net corrosion reaction is the result of two or more partial electrochemical reactions that proceed independently of each other (Wagner & Traud, 1938). For the case of metallic corrosion in presence of an aqueous medium, the corrosion process can be written as,



where  $z$  is the number of electrons lost per atom of the metal. This reaction is the result of an anodic (oxidation) reaction,



and a cathodic (reduction) reaction,



It is assumed that the anodic and cathodic reactions occur at a number of sites on a metal surface and that these sites change in a dynamic statistical distribution with respect to location and time. Thus, during corrosion of a metal surface, metal ions are formed at anodic sites with the loss of electrons and these electrons are then consumed by water molecules to form hydrogen molecules. The interaction between the anodic and cathodic sites as described on the basis of mixed potential theory is represented by well-known relationships using current (reaction rate) and potential (driving force). For the above pair of electrochemical reactions (anodic (2) and cathodic (3)), the relationship between the applied current  $I_a$

and potential  $E_a$  follows the Butler-Volmer equation,

$$I_a = I_{corr} \left\{ \exp \left[ \frac{2.303 (E_a - E_{corr})}{\beta_a} \right] - \dots \exp \left[ -\frac{2.303 (E_a - E_{corr})}{\beta_c} \right] \right\}, \quad (4)$$

where  $\beta_a$  and  $\beta_c$  are the anodic and cathodic Tafel parameters given by the slopes of the polarization curves  $\partial E_a / \partial \log_{10} I_a$  in the anodic and cathodic Tafel regimes, respectively and  $E_{corr}$  is the corrosion potential (Bockris et al., 2000).

### 2.2. Electrode Configuration

An electrode is a (semi-)conductive solid that interfaces with an electrolytic solution. The most common electrode configuration is the three-electrode configuration. The common designations are: working, reference and counter electrodes. The working electrode is the designation for the electrode being studied. In corrosion experiments, this is the material that is corroding. The counter electrode is the electrode that completes the current path. All electrochemistry experiments contain a working-counter pair. In most experiments the counter electrode is simply the current source/sink comprised of inert materials like graphite or platinum. Finally, the reference electrode serves as an experimental reference point, specifically for potential (sense) measurements. The reference electrode is positioned so that it measures a point very close to the working electrode.

The three-electrode setup has a distinct experimental advantage over a two electrode setup: only one half of the cell is measured. That is, potential changes of the working electrode are measured independently of changes that may occur at the counter electrode. This configuration also reduces the effect of measuring potential drops across the solution resistance when measuring between the working and counter electrodes.

### 2.3. Polarization Resistance

The corrosion current,  $I_{corr}$ , cannot be measured directly. However, a-priori knowledge of  $\beta_a$  and  $\beta_c$  along with a small signal analysis technique, known as polarization resistance, can be used to indirectly compute  $I_{corr}$ . The polarization resistance technique, also referred to as "linear polarization", is an experimental electrochemical technique that estimates the small signal changes in  $I_a$  when  $E_a$  is perturbed by  $E_{corr} \pm 10$  mV (G102, 1994). The slope of the resulting curve over this range is the polarization resistance,

$$R_p \triangleq \left. \frac{\partial E_a}{\partial I_a} \right|_{|E_a - E_{corr}| \leq 10 \text{ mV}}. \quad (5)$$

Note, the applied current,  $I_a$ , is the total applied current and is not multiplied by the electrode area so  $R_p$  as defined in (5)

has units of  $\Omega$ . Provided that  $|E_a - E_{corr}|/\beta_a \leq 0.1$  and  $|E_a - E_{corr}|/\beta_c \leq 0.1$ , the first order Taylor series expansion  $\exp(x) \approx 1 + x$  can be applied to (4) and (5) to arrive at,

$$R_p = \frac{1}{2.303I_{corr}} \left( \frac{\beta_a\beta_c}{\beta_a + \beta_c} \right). \quad (6)$$

Finally, this expression can be re-written for  $I_{corr}$  to arrive at the Stern-Geary equation,

$$I_{corr} = \frac{B}{R_p}, \quad (7)$$

where  $B = \frac{1}{2.303} [\beta_a\beta_c/(\beta_a + \beta_c)]$  is a constant of proportionality.

## 2.4. Pit-Depth

The pit-depth due to corrosion is calculated by computing the pitting current density,  $i_{pit}$ ,

$$i_{pit}(t) = \frac{i_{corr} - i_{pv}}{N_{pit}}, \quad (8)$$

where  $i_{corr} = I_{corr}/A_{sen}$  is the corrosion current density,  $i_{pv}$  is the passive current density,  $N_{pit}$  is the pit density for the alloy (derived empirically) and  $A_{sen}$  is the effective surface area of the LPR sensor. One critical assumption is the pH is in the range of 6-8. If this cannot be assumed, then a measurement of pH is required and  $i_{passive}$  is needed over the range of pH values. Next, Faraday's law is used to relate the total pitting charge with respect to molar mass loss. Let the equivalent weight ( $EW$ ) represent the weight of the metal that reacts with 1 C of charge, thus contributing to the corrosion and overall loss of material in the anodic (oxidation) reaction given in (2). The total pitting charge,  $Q_{corr}$ , and molar mass loss,  $M$ , can be related to the following,

$$Q_{pit}(t) = zF \cdot M(t), \quad (9)$$

where  $F = 9.650 \times 10^4$  C/mol is Faraday's constant, and  $z$  is the number of electrons lost per atom in the metal in the reduction-oxidation reaction. The  $EW$  is calculated from the known Atomic Weight ( $AW$ ) of the metal,

$$EW = \frac{AW}{z}. \quad (10)$$

Next, the number of moles of the metal reacting can be converted to an equivalent mass loss,  $m_{loss}$ ,

$$m_{loss}(t) = M(t) \cdot AW. \quad (11)$$

Combining (9) through (11), the mass loss  $m_{loss}$  is related to  $Q_{pit}$  by,

$$m_{loss}(t) = \frac{EW \cdot Q_{pit}(t)}{F}. \quad (12)$$

With the mass loss calculated and knowing the density  $\rho$ , the pit-depth modeled using a semi-spherical volume with a depth (or radius)  $d$  is expressed as,

$$d(t) = \left( \frac{3m_{loss}(t)}{2\pi\rho} \right)^{1/3}. \quad (13)$$

Now, note that  $Q_{pit}$  can be found by integrating  $i_{pit}$  over the total time,

$$Q_{pit}(t) = \int_0^t i_{pit}(\tau) d\tau, \quad (14)$$

Substituting (12) and (14) into (13) gives,

$$d(t) = \sqrt[3]{\frac{3EW}{2\pi\rho F} \int_0^t i_{pit}(\tau) d\tau}. \quad (15)$$

Next, by substituting (7) and (8) into (15), the expression for  $d$  can be rewritten as,

$$d(t) = \sqrt[3]{\frac{3EW}{2\pi\rho N_{pit}F} \int_0^t \left( \frac{B}{A_{sen}R_p(\tau)} - i_{pv} \right) d\tau}. \quad (16)$$

In practice,  $R_p$  is not measured continuously, rather, periodic measurements are taken every  $T_s$  seconds. If its assumed over this interval the  $R_p$  values changes linearly then the mean value theorem for integrals can be applied to arrive at an alternative expression for  $d$ ,

$$d(t) = \sqrt[3]{\frac{3T_sEW}{2\pi\rho N_{pit}F} \sum_{k=0}^{N-1} \left( \frac{B}{A_{sen}R_p(kT_s)} - i_{pv} \right)}. \quad (17)$$

## 2.5. Standard Measurements

### 2.5.1. Polarization Resistance

ASTM standard G59 outlines procedures for measuring polarization resistance. Potentiodynamic, potential step, and current-step methods can be used to compute  $R_p$  (G59, 1994). The potentiodynamic sweep method is the most common method for measuring  $R_p$ . A potentiodynamic sweep is conducted by applying  $E_a$  between  $E_{corr} \pm 10$  mV at a slow scan rate, typically 0.125 mV/s. A linear fit of the resulting  $E_a$  vs.  $I_a$  curve is used to compute  $R_p$ .

### 2.5.2. Tafel Coefficients

ASTM standard G59 outlines the procedure for measuring the Tafel slopes,  $\beta_a$  and  $\beta_c$  (G59, 1994). First,  $E_{corr}$  is measured from the open circuit potential. Next,  $E_a$  is initialized to  $E_{corr} - 250$  mV. Then, a potentiodynamic sweep is conducted by increasing  $E_a$  from  $E_{corr} - 250$  mV to  $E_{corr} + 250$  mV at a slow scan rate, typically 0.125 mV/s. Next, a Tafel curve is plotted for  $E_a$  vs  $\log_{10} I_a$  as illustrated by the example in Figure 3. Values for  $\beta_a$  and  $\beta_c$  are estimated from the slopes of the linear extrapolated anodic and cathodic currents, which are identified in Figure 3.

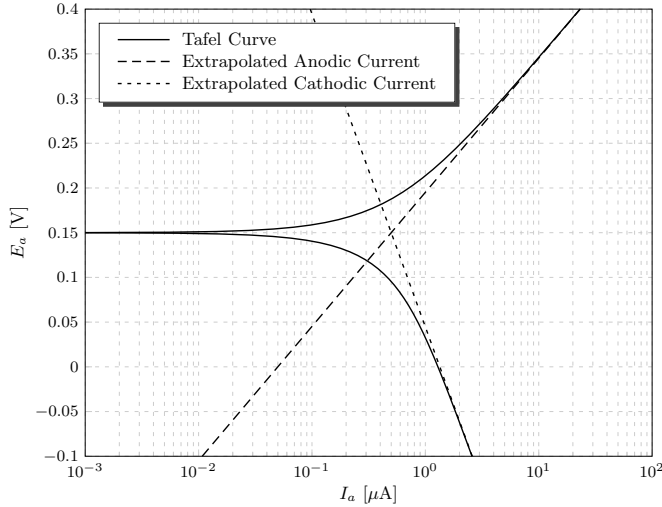


Figure 3. Illustration of a typical Tafel curve identifying the extrapolated anodic and cathodic currents.

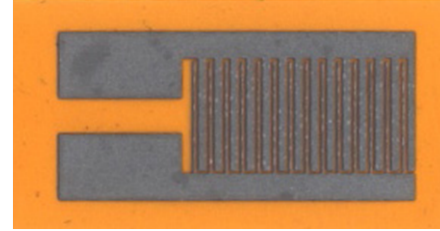
### 3. $\mu$ LPR CORROSION SENSOR

In this section, a micro-LPR ( $\mu$ LPR) is presented which uses the potential step-sweep method to compute polarization resistance. The  $\mu$ LPR works on the same principle as the macro-sized LPR sensors and is designed to corrode at the same rate as the structure on which it is placed. Although LPR theory is well established and accepted as a viable corrosion monitoring technique, conventional macro-sized LPR sensor systems are expensive and highly intrusive. The  $\mu$ LPR is a micro-scaled LPR sensor inspired from the macro-sized version discussed in the previous section. Scaling the LPR sensor into a micro-sized package provides several advantages which include,

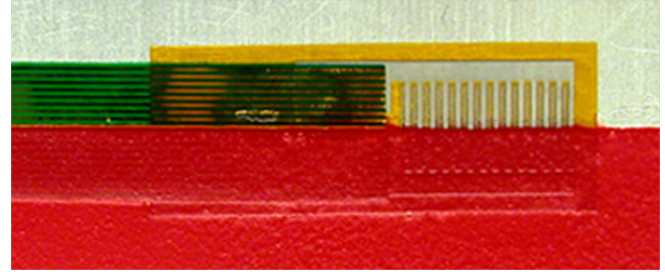
- Miniature form factor
- Two-pair electrode configuration
- Faster LPR measurements

#### 3.1. Form Factor

Expertise in semiconductor manufacturing is used to micro-machine the  $\mu$ LPR. Using photolithography it is possible to manufacture the  $\mu$ LPR sensor from a variety of standard engineering construction materials varying from steels for buildings and bridges through to novel alloys for airframes. The micro sensor is made up of two micro machined electrodes that are interdigitated at  $150\ \mu\text{m}$  spacing. The  $\mu$ LPR sensor is made from shim stock of the source/sample material that is pressure and thermally bonded to Kapton tape. The shim is prepared using photolithographic techniques and Electro Chemical Etching (ECM). It is further machined on the Kapton to produce a highly ductile and mechanically robust micro sensor that is very sensitive to corrosion. Images of the  $\mu$ LPR shown bare and a fitted sensor underneath a coating are shown in Figure 4.



(a)



(b)

Figure 4. Thin film  $\mu$ LPR sensor (a) exposed and (b) quasi-exposed with the lower-half underneath a coating.

#### 3.2. Electrode Configuration

The  $\mu$ LPR differs from conventional macro-sized LPR sensors in two major ways. First, the  $\mu$ LPR is a two electrode device. The reference electrode is eliminated as the separation distance between the working and counter electrodes, typically  $150\ \mu\text{m}$ , minimizes any voltage drop due to the solution resistance,  $R_s$ . Second, both electrodes are composed of the same working metal. This is uncommon in most electrochemical cells where the counter electrode is made of an inert material. The benefit is the electrodes provide a more direct measurement of corrosion than techniques which use electrodes made of different metals (eg. gold).

#### 3.3. LPR Measurements

Potential step-sweeps are performed by applying a series of 30 steps over a range of  $\pm 10\ \text{mV}$  spanning a period of 2.6 s. This allows eight  $\mu$ LPR sensors to be measured in less than 30 s. However, the effective scan-rate of  $7.7\ \text{mV/s}$  generates an additional current,  $I_{dl}$ , due to rapid charging and discharging of the capacitance, referred to as the double-layer capacitance  $C_{dl}$ , at the electrode-electrolyte interface,

$$I_{dl} = C_{dl} \frac{dE_a}{dt}. \quad (18)$$

Let the resulting polarization resistance that is computed when  $I_{dl}$  is non-zero be represented by  $\hat{R}_p$ . It can be shown that  $\hat{R}_p$  is related to  $R_p$  by the following,

$$\hat{R}_p^{-1} = R_p^{-1} + Y_{dl}, \quad (19)$$

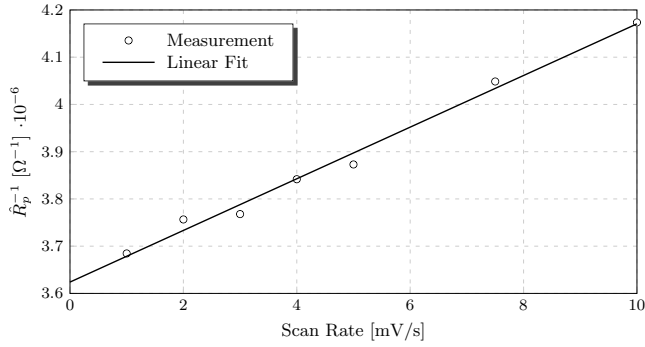


Figure 5. Plot of inverse polarization resistance vs. scan-rate for a  $\mu$ LPR sensor made from AA 7075-T6 submersed in tap water.

such that  $Y_{dl}$  is defined by the admittance,

$$Y_{dl} = \left( \frac{C_{dl}}{20 \text{ mV}} \right) \frac{dE_a}{dt} \quad (20)$$

where  $dE_a/dt$  is the scan rate. An example of this relationship is provided in Figure 5. In this example  $C_{dl}/20 \text{ mV}$  and  $\hat{R}_p^{-1}$  correspond to the slope and y-intercept; these values were computed as  $5.466 \times 10^{-8} \Omega^{-1} \cdot \text{s/mV}$  and  $3.624 \times 10^{-6} \Omega$ , respectively. For a scan rate of  $dE_a/dt = 7.7 \text{ mV/s}$ ,  $Y_{dl}$  is computed as  $4.209 \times 10^{-7} \Omega^{-1}$ . Finally, for a given solution,  $\hat{R}_p$  can be compensated by,

$$R_p = \frac{\hat{R}_p}{1 - Y_{dl} \hat{R}_p} \quad \text{for} \quad Y_{dl} \hat{R}_p < 1. \quad (21)$$

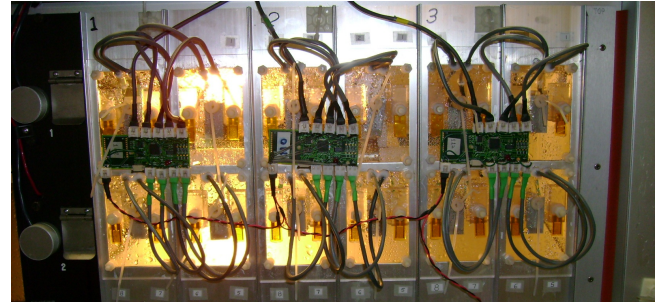
## 4. EXPERIMENT

### 4.1. Setup

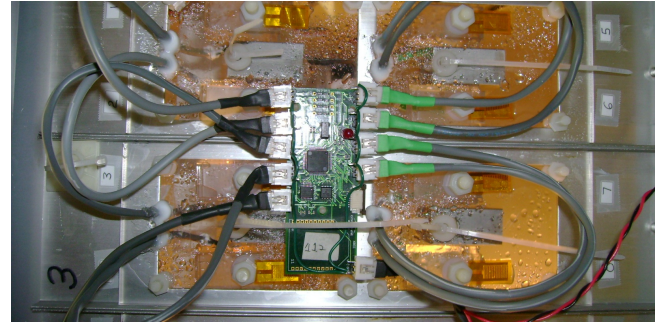
The experiment consisted of twenty-four (24)  $\mu$ LPR sensors and twelve (12) control coupons. The coupons and  $\mu$ LPR sensors were made from AA 7075-T6. Each coupon was placed next to a pair of  $\mu$ LPR sensors. Each sensor was held in place using a non-reactive polycarbonate clamp with a nylon fitting. All the sensors and coupons were mounted on an acrylic plexiglass base with the embedded hardware placed on the opposite side of the frame, shown in Figure 6. An electronic precision balance (Tree HRB-203) with a calibrated range of  $0 - 200 \text{ g}$  ( $\pm 0.001 \text{ g}$ ) was used to weigh the coupons before and after the experiment. Finally, a weathering chamber (Q-Lab QUV/spray) promoted corrosion on the coupons and  $\mu$ LPR sensors by applying a controlled stream of tap water for 10 seconds every five minutes.

### 4.2. Procedure

First, the surface of each coupon was cleaned using sandblasting. Then, each coupon was weighed using the analytical balance. The entire panel of coupons and  $\mu$ LPR sensors were



(a)



(b)

Figure 6. Experimental setup showing (a) all 24  $\mu$ LPR sensors, 12 coupons and three AN101 instrumentation boards and (b) a close-up view of one of the panels used in the experiment.

placed in the weathering chamber for accelerated testing. The experiment ran for approximately 60 days. During the experiment, a set of coupons were periodically removed from the weathering chamber. Throughout the experiment, the SHM embedded hardware was logging  $\hat{R}_p$  from each  $\mu$ LPR sensor. The sample rate was set at one sample per minute. Once accelerated testing was finished, the coupons were removed and the LPR data was downloaded and archived for analysis. The corrosion byproducts were removed from each coupon by applying micro-bead blasting to the coupon surface. Finally, the cleaned coupons were weighed using the analytical scale to compute the relative corrosion depth during the experiment. Figure 7 shows images of coupons before and after micro-bead blasting. Also shown are images comparing two sets of three coupons after 15 days and 57 days of corrosion proceeding micro-bead blasting.

## 5. RESULTS

### 5.1. Coupon Corrosion

The corrosion byproducts were carefully removed using micro-bead blasting. The pitting depth,  $d$ , of each coupon was calculated using the formula,

$$d = 3 \sqrt[3]{\frac{3m_{loss}}{2\pi\rho N_{pit}A_{exp}}}, \quad (22)$$

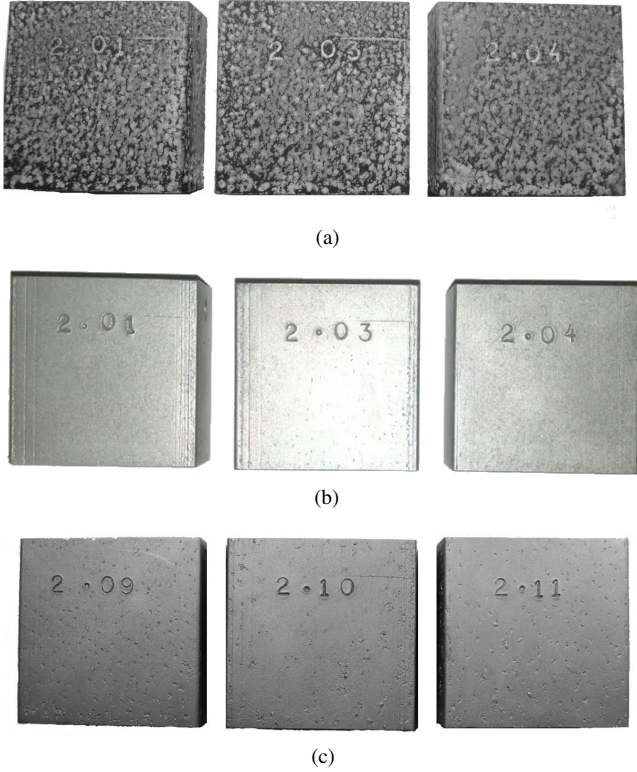


Figure 7. Image of the three AA 7075-T6 coupons (ID 2.01, 2.03 and 2.04) after approximately 15 days of corrosion testing showing (a) the condition of the coupons before cleaning and (b) after cleaning using micro-bead blasting. Also shown for comparison are (c) three AA 7075-T6 coupons (ID 2.09, 2.10 and 2.11) after 57 days and cleaning using micro-bead blasting.

where values for the mass loss  $m_{loss}$ , exposed surface area  $A_{exp}$ , resulting pit-depth,  $d$ , and total time of exposure of each coupon is provided in Table 1 in the Appendix. Values for the pitting density and  $\rho$  were set at  $N_{pit} = 11 \text{ cm}^{-2}$  and  $\rho = 2.810 \text{ g/cm}^3$ , respectively. The pitting density was computed by counting the average number of pits over the surface for coupons 2.06 and 2.08. The measurement uncertainty in the pit-depth due to uncertainty in the mass loss,  $\Delta m_{loss}$  and pit density,  $\Delta N_{pit}$ , is approximately,

$$\Delta d \approx \frac{d}{3} \left( \frac{\Delta m_{loss}}{m_{loss}} + \frac{\Delta N_{pit}}{N_{pit}} \right), \quad (23)$$

where  $\Delta m_{loss} = \pm 0.001 \text{ g}$  is the minimum resolution of the scale and  $\Delta N_{pit} = \pm 3 \text{ cm}^{-2}$  was the standard deviation of the measured pit density over  $1 \text{ cm}^2$  sample areas for coupons 2.06 and 2.08.

## 5.2. $\mu$ LPR Corrosion

The linear polarization resistance measurements were used to compute corrosion pit-depth for each  $\mu$ LPR sensor. The computed pit-depth for each of the 24  $\mu$ LPR sensors over a period of approximately 60 days is provided in Figure 2(a).

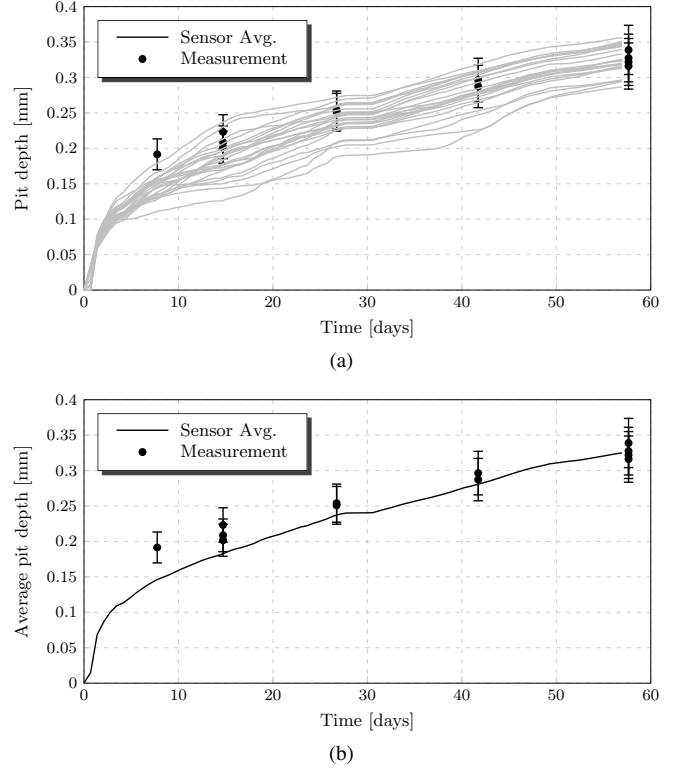


Figure 8. Comparison of the measured and computed pit-depth over a period of approximately 60 days for (a) each  $\mu$ LPR sensor and (b) the average of all  $\mu$ LPR sensors.

Superimposed in the plot are the pit-depth measurements of the corrosion coupons. The average computed pit-depth for each of the 24  $\mu$ LPR sensors over the same period is shown in Figure 2(b).

Next, the corrosion coupons were used to evaluate the performance of the  $\mu$ LPR sensor. Figure 9(a) compares the measured pit-depth with the pit-depth computed from each of the 24  $\mu$ LPR sensors during the removal of each corrosion coupon. A solid line is used to identify an ideal one-to-one relationship between the measured and computed pit-depth. Figure 9(b) shows the distribution of the computed residuals from Figure 9(a). The residuals follow a Gaussian distribution with a computed mean and standard deviation of  $0.0153 \text{ mm}$  and  $0.0272 \text{ mm}$ , accordingly. Figure 9(c) compares the measured pit-depth with the pit-depth computed from the average of the 24  $\mu$ LPR sensors during the removal of each corrosion coupon. Also provided in the plot are the error bars corresponding to one standard deviation in the corrosion measurements.

## 6. APPLICATIONS IN PROGNOSTICS

The data generated from this experiment can be used to demonstrate the feasibility of the  $\mu$ LPR sensor for use in prognostics. According to (Vachtsevanos, Lewis, Roemer, Hess, & Wu, 2006), prognostics is the ability to predict ac-

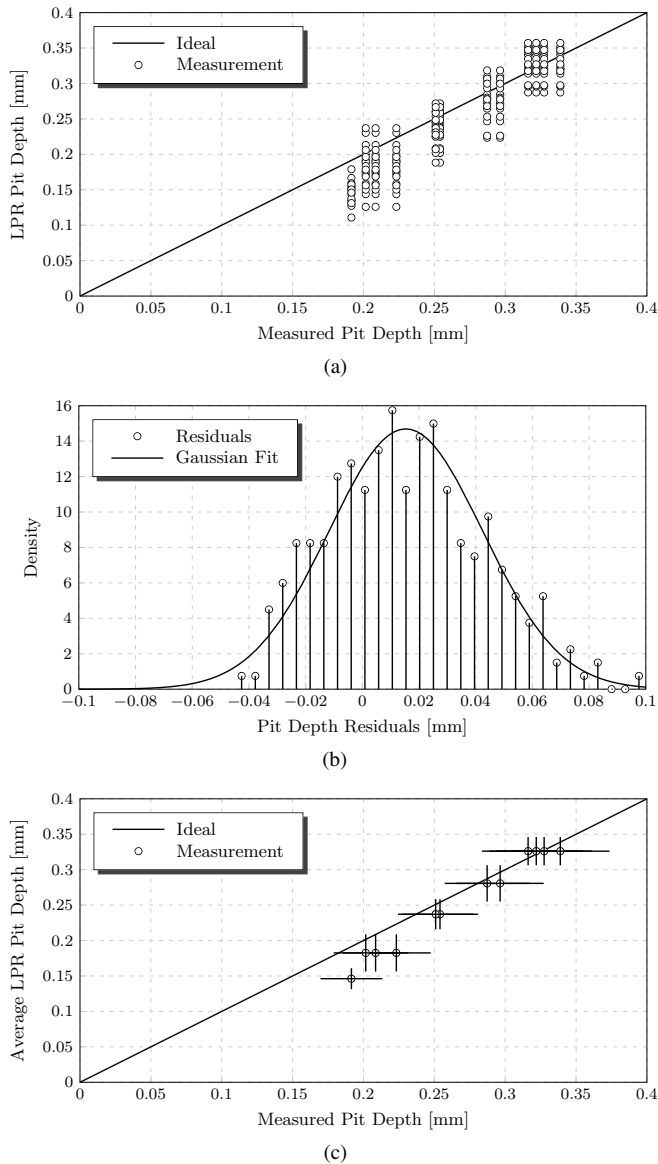


Figure 9. Direct comparison of the measured versus computed pit-depth for (a) each  $\mu$ LPR sensor measured at each coupon removal, (b) the corresponding distribution of the computed residuals and (c) average  $\mu$ LPR sensor measurement at each coupon removal.

curately and precisely the Remaining Useful Life (RUL) of a failing component or subsystem. In this application the failing component is the metallic structure being monitored, more specifically AA 7075-T6, by the  $\mu$ LPR sensor. The physical quantity to be predicted, otherwise referred to as the fault dimension, is the pit-depth of the metallic structure.

Typically the fault dimension is not a quantity that is directly measured. Rather, the fault dimension is commonly computed from a mapping of one or more indirect measures, or features. In this application, the features are temperature and polarization resistance. In this particular experiment, temperature remained at (or near) room temperature. As a re-

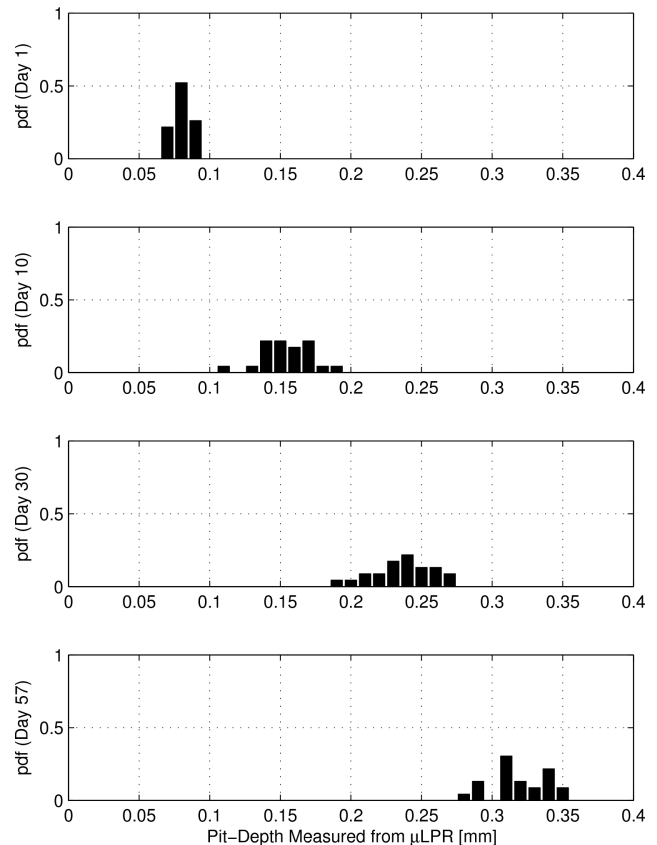


Figure 10. Histograms of the fault dimension (pit depth) after 1, 10, 30 and 57 days into the experiment.

sult, the primary feature is polarization resistance. The fault dimension was computed using the mathematical model derived earlier in Section 2.4 as a mapping function.

Of course, the fault dimension does not stay constant with time. The fault grows with many factors including the current fault dimension and the operating environment. In this example, the operating environment is the environmental chamber with the 5 minute spray cycle operating at room temperature. Figure 10 shows the distribution of the fault dimension for 24  $\mu$ LPR sensors at four different time intervals correspond to the end of day 1, 10, 30 and 57. As time progresses, the distribution of the fault dimension changes. In practice, a fault-growth model is used to project the fault dimension over a future time interval. The fault-growth model usually includes multiple factors. In this example, the fault-growth, or corrosion rate, is dependent on factors that include the solution, temperature, pH and time of wetness. Please note: in the scope of this paper no fault-growth model was studied. All of the data analysis for prognosis feasibility was performed a-posteriori using the  $\mu$ LPR data only.

Knowledge of the fault dimension distribution at any moment in time is important in prognostics. The RUL is defined as a projection of the fault dimension onto the time domain for a fixed value, referred to as the failure threshold. The fail-

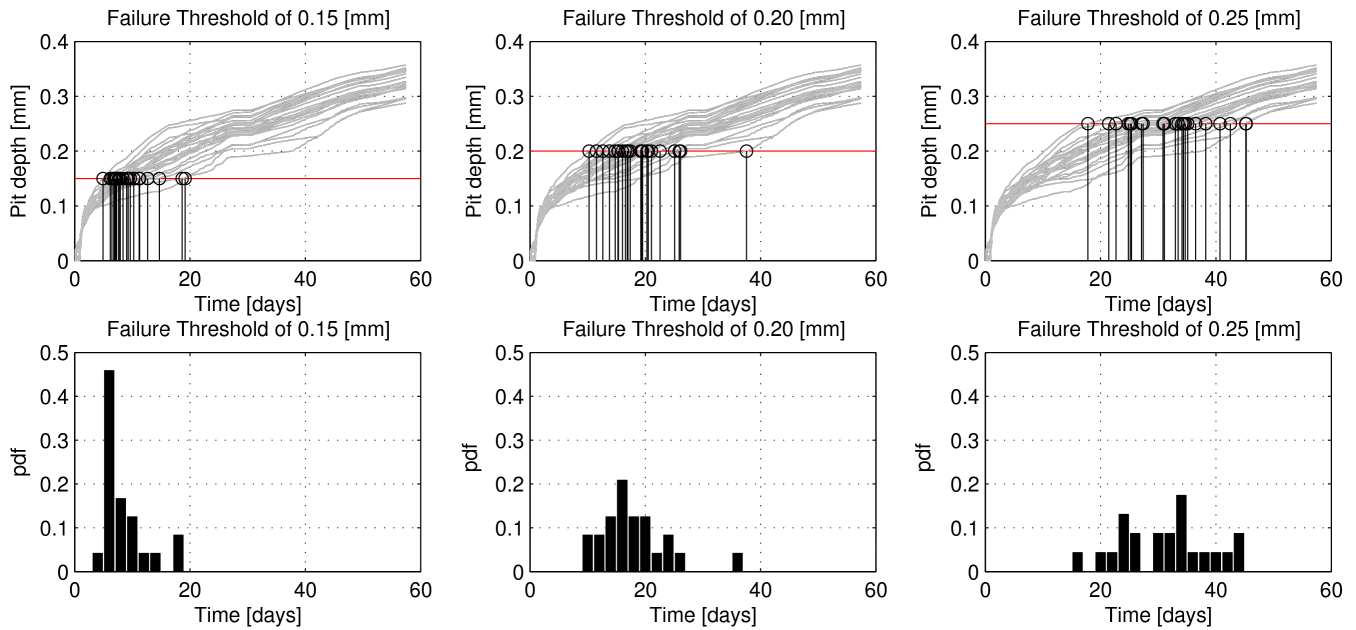


Figure 11. Prognosis feasibility demonstrated by (top) plots of (top) fault-dimension (pit-depth) vs. time identifying the time when fault-dimension computed from each of the 24  $\mu$ LPR sensors first exceeds the failure threshold of (from left-to-right) 0.150 mm, 0.200 mm and 0.250 mm, respectively. Also shown (bottom) are histograms of the RUL corresponding to each respective failure threshold.

ure threshold value is usually determined a-priori by an engineer with experience in the particular area of application. In this example, three failure thresholds are selected for the fault dimension: 0.150 mm, 0.200 mm and 0.250 mm. The upper plots in Figure 11 show the fault-dimension measured from each  $\mu$ LPR sensor over a period of approximately 60 days. Starting from left to right, the failure threshold values of 0.150 mm, 0.200 mm and 0.250 mm are identified with a red horizontal line. The intersection of the failure threshold and each of the 24 fault-dimension curves are identified with a superimposed stem plot. The lower plots in Figure 11 show the probability density function (pdf) of each respective stem plot on the time-axis. Each pdf represents the RUL starting from the beginning of the experiment at time zero. Note, as the failure threshold is raised, both the mean value and variance of the RUL grows. This demonstrates the uncertainty in the RUL prediction increases over an increasing time interval, which is to be expected.

## 7. SUMMARY

A micro-sized LPR ( $\mu$ LPR) sensor was presented for corrosion monitoring in Structural Health Management (SHM) applications. An experimental test was performed to compare corrosion measurements from twenty-four  $\mu$ LPR sensors with twelve corrosion coupons. Both the corrosion coupons and sensors were constructed from the same material, AA 7075-T6. According to the results, the pit-depth computed from the  $\mu$ LPR sensors agreed the pit-depth measured from the corrosion coupons to within a statistical con-

fidence of 95%. The results indicate multiple  $\mu$ LPR sensors can be used to provide an accurate measurement of corrosion. The paper concluded with a feasibility study for the  $\mu$ LPR sensor in prognostic applications.

Future work includes a combination of prognostic model development, extensive laboratory testing and field testing. Before the  $\mu$ LPR sensor can be used in prognostic applications, a prognostic algorithm, more specifically a fault-growth model, must be developed. Additional laboratory testing is required to: evaluate the  $\mu$ LPR sensors for different alloys (eg. AA 2024-T3); perform experiments using standardized protocols (eg. SAE J2334 & ASTM G85); perform a blind study evaluating  $\mu$ LPR-based prognostic algorithms; and certify the AN101 SHM system for electromagnetic interference (EMI), electromagnetic conductance (EMC) and environmental conditions following MIL-STD-461F and MIL-STD-810G specifications. Field testing is necessary to evaluate the performance of the sensor and SHM system in a real-world environment. The system has been certified for initial flight testing on a C-130 legacy aircraft. Follow-on testing includes performing a smaller-scale validation experiment for the  $\mu$ LPR in-flight.

## ACKNOWLEDGMENT

All funding and development of the sensors and systems in the project has been part of the US government's SBIR programs. In particular: 1) In preparing the initial system design and development, funding was provided by the US Air Force under SBIR Phase II contract # F33615-01-C-5612 monitored



by Dr. James Mazza, 2) Funding for the development and experimental set-up was provided by the US Navy under SBIR Phase II contract # N68335-06-C-0317 monitored by Dr. Paul Kulowitch, and 3) for further improvements and scheduled field installations by the US Air Force under SBIR Phase II contract # FA8501-11-C-0012 monitored by Mr. Feraidoon Zahiri.

#### NOMENCLATURE

$\beta_a$	Anodic Tafel slope	V/dec
$\beta_c$	Cathodic Tafel slope	V/dec
$\rho$	Density	g/mm <sup>3</sup>
$d$	Corrosion depth	cm
$k$	LPR sample index	–
$i_{corr}$	Corrosion current density	A/cm <sup>2</sup>
$i_{pit}$	Pitting current density	A/cm <sup>2</sup>
$i_{pv}$	Passive current density	A/cm <sup>2</sup>
$m_{loss}$	Mass loss due to corrosion	g
$z$	Number of electrons lost per atom	–
$\Delta d$	Corrosion depth uncertainty	cm
$\Delta m_{loss}$	Mass loss uncertainty	g
$\Delta N_{pit}$	Pit density uncertainty	cm <sup>-2</sup>
$A_{exp}$	Exposed coupon area	cm <sup>2</sup>
$A_{sen}$	Effective sensor area	cm <sup>2</sup>
$AW$	Atomic Weight	g/mol
$B$	Proportionality constant	V/dec
$C_{dl}$	Double-layer capacitance	F
$E_a$	Applied potential	V
$E_{corr}$	Corrosion voltage	V
$EW$	Equivalent weight	g/mol
$F$	Faraday's constant	C/mol
$I_a$	Applied current	A
$I_{corr}$	Corrosion current	A
$I_{dl}$	Scanning current from $C_{dl}$	A
$M$	Number of moles reacting	mol
$N$	Total number of $\mu$ LPR samples	–
$N_{pit}$	Pit density	cm <sup>-2</sup>
$Q_{corr}$	Charge from oxidation reaction	C
$R_p$	Polarization resistance	$\Omega$
$\hat{R}_p$	Measured polarization resistance	$\Omega$
$R_s$	Solution resistance	$\Omega$
$T_s$	Sampling period	s
$Y_{dl}$	Scanning admittance from $C_{dl}$	s

#### REFERENCES

Bockris, J. O., Reddy, A. K. N., & Gambola-Aldeco, M. (2000). *Modern Electrochemistry 2A. Fundamentals of Electrode Processes* (2nd ed.). New York: Kluwer Academic/Plenum Publishers.

Buchheit, R. G., Hinkebein, T., Maestas, L., & Montes, L. (1998, March 22-27). Corrosion Monitoring of Concrete-Lined Brine Service Pipelines Using AC and

DC Electrochemical Methods. In *CORROSION 98*. San Diego, Ca.

Burstein, G. T. (2005, December). A Century of Tafel's Equation: 1905-2005. *Corrosion Science*, 47(12), 2858-2870.

G102, A. S. (1994). Standard Practice for Calculation of Corrosion Rates and Related Information from Electrochemical Measurements. *Annual Book of ASTM Standards*, 03.02.

G59, A. S. (1994). Standard Practice for Conducting Potentiodynamic Polarization Resistance Measurements. *Annual Book of ASTM Standards*, 03.02.

Harris, S. J., Mishon, M., & Hebborn, M. (2006, October). Corrosion Sensors to Reduce Aircraft Maintenance. In *RTO AVT-144 Workshop on Enhanced Aircraft Platform Availability Through Advanced Maintenance Concepts and Technologies*. Vilnius, Lithuania.

Huston, D. (2010). *Structural Sensing, Health Monitoring, and Performance Evaluation* (B. Jones & W. B. S. J. Jnr., Eds.). Taylor and Francis.

*Introduction to Corrosion Monitoring*. (2012, August 20). Online. Available from <http://www.alspi.com/introduction.htm>

Twomey, M. (1997). Inspection Techniques for Detecting Corrosion Under Insulation. *Material Evaluation*, 55(2), 129-133.

Vachtsevanos, G., Lewis, F., Roemer, M., Hess, A., & Wu, B. (2006). *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*. Hoboken, NJ, USA: John Wiley and Sons.

Wagner, C., & Traud, W. (1938). *Elektrochem*, 44, 391.

#### BIOGRAPHIES

**Douglas W. Brown** is the senior systems engineer at Analatom with eight years of experience developing and maturing PHM and fault-tolerant control systems in avionics applications. He received the B.S. degree in electrical engineering from the Rochester Institute of Technology in 2006 and the M.S/Ph.D. degrees in electrical engineering from the Georgia Institute of Technology in 2008 and 2011, respectively. Dr. Brown is a recipient of the National Defense Science and Engineering Graduate Fellowship and has received several best-paper awards in his work in prognostics and fault-tolerant control.

**Duane Darr** is the senior embedded hardware engineer at Analatom with over 30 years of experience in the software and firmware engineering fields. He completed his undergraduate work in physics, and graduate work in electrical engineering and computer science at Santa Clara University. Mr. Darr's previous work at Epson Imaging Technology Center, San Jose, California, as Senior Software Engineer,

Data Technology Corporation, San Jose, California as Senior Firmware Engineer, and Qume Inc., San Jose California, as Member of Engineering Staff/Senior Firmware Engineer, focused on generation and refinement of software and firmware solutions for imaging core technologies as well as digital servo controller research, development, and commercialization.

**Jefferey Morse** is the director of advanced technology at Analatom since 2007. Prior to this, he was a senior scientist in the Center for Micro and Nano Technology at Lawrence Livermore National Laboratory. He received the B.S. and M.S. degrees in electrical engineering from the University of Massachusetts Amherst in 1983 and 1985, respectively, and a Ph.D. in electrical engineering from Stanford University in 1992. Dr. Morse has over 40 publications, including 12 journal papers, and 15 patents in the areas of advanced materi-

als, nanofabrication, sensors and energy conversion technologies. He has managed numerous projects in various multidisciplinary technical areas, including electrochemical sensors and power sources, vacuum devices, and microfluidic systems.

**Bernard Laskowski** is the president and senior research scientist at Analatom since 1981. He received the Licentiaat and Ph.D. degrees in Physics from the University of Brussels in 1969 and 1974, respectively. Dr. Laskowski has published over 30 papers in international refereed journals in the fields of micro physics and micro chemistry. As president of Analatom, Dr. Laskowski managed 93 university, government, and private industry contracts, receiving a U.S. Small Business Administration Administrator's Award for Excellence.

Table 1. Experimental measurements of coupon corrosion.

Coupon ID	Time Exposed [min]	Area [cm <sup>2</sup> ]	Initial Mass [g]	Final Mass [g]	Mass Loss [g]	pit-depth [mm]
Control	0	$5.801 \times 10^1$	$7.6870 \times 10^1$	$7.6869 \times 10^1$	$1 \times 10^{-3}$	N/A
2.01	$2.1198 \times 10^4$	$5.805 \times 10^1$	$7.7253 \times 10^1$	$7.7215 \times 10^1$	$3.8 \times 10^{-2}$	$2.232 \times 10^{-1}$
2.02	$1.1160 \times 10^4$	$5.798 \times 10^1$	$7.6842 \times 10^1$	$7.6818 \times 10^1$	$2.4 \times 10^{-2}$	$1.915 \times 10^{-1}$
2.03	$2.1198 \times 10^4$	$5.799 \times 10^1$	$7.6927 \times 10^1$	$7.6896 \times 10^1$	$3.1 \times 10^{-2}$	$2.068 \times 10^{-1}$
2.04	$2.1198 \times 10^4$	$5.805 \times 10^1$	$7.6897 \times 10^1$	$7.6869 \times 10^1$	$2.8 \times 10^{-2}$	$2.016 \times 10^{-1}$
2.05	$6.0090 \times 10^4$	$5.801 \times 10^1$	$7.6967 \times 10^1$	$7.6886 \times 10^1$	$8.1 \times 10^{-2}$	$2.873 \times 10^{-1}$
2.06	$3.8510 \times 10^4$	$5.798 \times 10^1$	$7.6884 \times 10^1$	$7.6828 \times 10^1$	$5.6 \times 10^{-2}$	$2.540 \times 10^{-1}$
2.07	$6.0090 \times 10^4$	$5.800 \times 10^1$	$7.6945 \times 10^1$	$7.6856 \times 10^1$	$8.9 \times 10^{-2}$	$2.964 \times 10^{-1}$
2.08	$3.8510 \times 10^4$	$5.803 \times 10^1$	$7.6921 \times 10^1$	$7.6810 \times 10^1$	$5.4 \times 10^{-2}$	$2.509 \times 10^{-1}$
2.09	$8.3010 \times 10^4$	$5.802 \times 10^1$	$7.7005 \times 10^1$	$7.6885 \times 10^1$	$1.20 \times 10^{-1}$	$3.275 \times 10^{-1}$
2.10	$8.3010 \times 10^4$	$5.802 \times 10^1$	$7.7165 \times 10^1$	$7.7057 \times 10^1$	$1.08 \times 10^{-1}$	$3.162 \times 10^{-1}$
2.11	$8.3010 \times 10^4$	$5.802 \times 10^1$	$7.6865 \times 10^1$	$7.6751 \times 10^1$	$1.14 \times 10^{-1}$	$3.219 \times 10^{-1}$
2.12	$8.3010 \times 10^4$	$5.802 \times 10^1$	$7.6902 \times 10^1$	$7.6769 \times 10^1$	$1.33 \times 10^{-1}$	$3.389 \times 10^{-1}$

# Markov Modeling of Component Fault Growth Over A Derived Domain of Feasible Output Control Effort Modifications

Brian Bole<sup>1</sup>, Kai Goebel<sup>2</sup>, George Vachtsevanos<sup>3</sup>

<sup>1,3</sup> *Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332, USA.*

*bbole3@gatech.edu*

*gfv@ece.gatech.edu*

<sup>2</sup> *NASA Ames Research Center, Moffett Field, CA 94035, USA.*

*kai.goebel@nasa.gov*

## ABSTRACT

This paper introduces a novel Markov process formulation of stochastic fault growth modeling, in order to facilitate the development and analysis of prognostics-based control adaptation. A metric representing the relative deviation between the nominal output of a system and the net output that is actually enacted by an implemented prognostics-based control routine, will be used to define the action space of the formulated Markov process. The state space of the Markov process will be defined in terms of an abstracted metric representing the relative health remaining in each of the system's components. The proposed formulation of component fault dynamics will conveniently relate feasible system output performance modifications to predictions of future component health deterioration.

## 1. INTRODUCTION

Continuous improvements in fault diagnostic and fault growth prognostic technologies have spawned a prolific growth in the application domain for these technologies, and a growing research interest in the development of techniques for using available diagnostic and prognostic information to improve system control and maintenance. Presently, a clear analytical process may be applied to implement and evaluate the effectiveness of tools for fault classification and fault growth prediction; however, the same cannot generally be said for the follow-on task of making intelligent control decisions based on available diagnostic and prognostic information. In general, the application of an analytical approach to the implementation and evaluation of prognostics-based decision making techniques will be complicated by the potential for high uncertainty in estimating the future effects of avail-

able control actions, and the need to define a computationally tractable space of present and future control decisions to be optimized over.

Several recent publications have suggested metrics for quantifying prognostic uncertainty and evaluating the ability of control modifications to affect meaningful change on fault growth predictions (Saxena et al., 2008; P. Wang et al., 2012; Edwards et al., 2010). This paper describes a process for abstracting uncertain models of environmental and fault growth dynamics into the generalized notation of a non-deterministic Markov process, in order to promote an application independent analysis of prognostics-based control strategies.

Markov decision processes have been widely applied to the representation of problems involving decision making in the presence of uncertain or stochastic modeling information in the contexts of economics (Haurie & Moresino, 2006), supply chain management (Parlara et al., 1995), scheduled maintenance (Smilowitz & Madanat, 1994), health care (Sonnenberg & Beck, 1993), and many other disciplines, in addition to being a widely used tool for describing fault-adaptive and robust control problems (Zhang & Jiang, 2008). A formal description of fault growth modeling and remaining useful life (RUL) estimation in terms of Markov process models, as well as a survey of similar stochastic modeling techniques, are given in Banjevic and Jardine (2006).

The state transition probabilities in a Markov process description of fault dynamics may be chosen to approximate an analytical formulation of a stochastic process, such as a Gaussian process model of fault growth dynamics, as described in Sankararaman et al. (2009); alternatively, state transition probabilities may be defined purely based on empirical observations of the fault growth process, as is the case with hidden Markov model learning techniques (Baruah & Chinnam, 2005), or they may be derived from a combination of a priori and empirical information.

Brian Bole et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

The derivation of stochastic fault growth models for a particular application is not specifically addressed in this publication; however, an effort is made to clearly identify how various sources of uncertain modeling information would be incorporated into the Markov process representation of fault dynamics. Arguments are provided for defining the state space of the fault growth process in terms of a metric representing the relative health of system components, as well as for defining the action space of the fault growth process in terms of a metric representing the relative deviation between the system's nominal output response and the net system output that is actually enacted by an implemented prognostics-based control routine at each control time-step. Operational constraints on minimum acceptable system output performance and maximum acceptable fault growth risk will be formulated as bounds on the action space available to implemented controllers.

## 2. GENERALIZED STOCHASTIC MODELING OF THE FAULT GROWTH PROCESS

Consider a discretized fault growth process of the form:

$$\gamma_l \in S_l, \quad S_l = \{\alpha_l, \alpha_l + \Delta_l, \alpha_l + 2\Delta_l, \dots, \alpha_l + m_l \Delta_l\} \quad (1)$$

$$p_{i,j}^l(\mathbf{x}(k), \mathbf{a}(k), \mathbf{w}(k)) = P(\gamma_l(k+1) = s_j | \gamma_l(k) = s_i, \mathbf{a} = \mathbf{a}(k), \mathbf{w} = \mathbf{w}(k)), \quad (2)$$

$$s_i, s_j \in S_l, \quad \mathbf{a}(k) \in \mathbf{A}(k), \quad \mathbf{w}(k) \in \mathbf{W}(k), \quad k \in \mathbb{N}$$

$$\sum_{j=0}^m p_{i,j}^l = 1, \quad \forall i \in \{0, 1, \dots, m\} \quad (3)$$

where  $\gamma_l(k)$  is a random variable representing the magnitude of the  $l^{\text{th}}$  component fault mode at time-index  $k$ ,  $S_l$  represents a uniformly quantized state space of potential fault magnitudes, and  $p_{i,j}^l(\mathbf{x}, \mathbf{a}, \mathbf{w})$  represents the probability of transitioning from damage state  $s_i$  to damage state  $s_j$ , given estimates of the system state  $\mathbf{x}$ , a set of low-level control commands  $\mathbf{a}$ , and the states of environmental and other exogenous inputs to the system  $\mathbf{w}$ . Eq. (3) specifies that the sum of all transition probabilities defined at each system state must always be equal to one.

The Markov process notation given in Eqs. (1)-(3) may be used to describe fault growth processes for all systems in which the following assumptions are satisfied:

**Assumption 1.** State transition probabilities  $p_{i,j}^l$  are only dependent on the current states of  $\gamma_l$ ,  $\mathbf{x}$ ,  $\mathbf{a}$ , and  $\mathbf{w}$ , and not on any states or inputs occurring previously. This is referred to as the memoryless assumption, or the Markov assumption. For cases in which the fault growth process is not completely memoryless, but future states only depend on a finite number  $m$  of previous states, the Markov assumption can be satisfied by defining the state space of the process to be the ordered  $m$ -

tuple of the current state and the  $m$  previously visited states (H. S. Wang & Chang, 1996).

**Assumption 2.** State transition probabilities are considered to be time invariant; although, it may be the case that fault growth models are not precisely known a priori and must be adapted online using techniques such as particle filtering (Orchard et al., 2008) or Bayesian learning (Saha et al., 2009). Online adaptation of the prognostic model on the basis of past observations will technically violate Assumption 1; however, the error induced by this effect is typically ignored because model adaptation transients are generally difficult to characterize and they will die out as the model is adapted.

**Assumption 3.** At all discrete time-steps, the state space  $S$ , the action space  $A$ , and the space of environmental and other exogenous inputs to the system  $W$  are adequately represented by a finite set of states, which will be bounded from above by the availability of computational resources. In the event that fault growth must be modeled as a continuous time process, such as the Poisson process (Shetty et al., 2008), a representation of fault growth modeling similar to that given in Eq. (2) may be expressed in terms of a continuous time Markov process (Serfozo, 1979) or a semi-Markov process (Dong & He, 2007).

The required assumptions are mild enough to allow a wide array of fault growth processes to be described in terms of the Markov process notation given in Eqs. (1)-(3) (Guidaa & Pulcini, 2011; Tang et al., 2009).

### 2.1. Formulating the Markov process in terms of component health rather than fault magnitude

Component fault magnitudes can generally be described by a real number corresponding to a measurable physical property such as crack length, spall width, or pitting depth; although, in many cases, faults cannot be directly measured in situ and diagnostic routines are needed to approximate current fault magnitudes based on the secondary effects observed in available sensor measurements. Sensor noise and modeling uncertainties will often result in significant diagnostic uncertainty, and it is common practice for diagnostic estimates to be reported in terms of a probability distribution over the potential fault magnitudes that could correspond to a given set observations.

The notation introduced in this subsection will add a layer of abstraction to the fault magnitude estimates produced by online fault diagnostic and fault growth prognostic routines. Rather than formulating the Markov process in terms of component fault magnitudes, a transformation is assumed to exist that will express the fault growth process in terms of an application independent metric representing the state of health (SOH) of each system component.

The SOH for component  $l$  at time  $t$  will be represented in

terms of a probability distribution over a uniformly quantized set of component health percentages between 0% and 100%;

$$\begin{aligned} \gamma_l(t) \in S, \quad S = \{s_1, s_2, \dots, s_m\}, \\ s_1 = 0\%, \quad s_i = s_{i-1} + \Delta\%, \quad 1 < i \leq 100\%/\Delta\% \quad (4) \end{aligned}$$

where  $\Delta\%$  is a quantization step-size for the state space of  $\gamma_l$ . The notation given in Eq. (2) is now considered to define the probability of transitioning from one component SOH to another, on the basis of the current states of  $\mathbf{x}$ ,  $\mathbf{a}$ , and  $\mathbf{w}$  at time-index  $k$ .

Component fault magnitudes are generally expected to monotonically increase with time; correspondingly, component SOH should monotonically decrease with time. A mandate of monotonically decreasing component health is incorporated into the Markov process notation as follows:

$$p_{i,j}^l = 0, \text{ if } j > i, \quad \sum_{j=0}^m p_{i,j}^l = 1 \quad \forall i \in \{0, 1, \dots, m\} \quad (5)$$

This constraint will be problematic for techniques that model process uncertainty with an analytical distribution that lacks an explicit lower bound. For example, in the case of fault growth prognostic techniques such as Kalman filtering and Gaussian process modeling, an assumption of Gaussian uncertainty will introduce some probability that the fault mode does not monotonically increase. In such cases, it would be necessary to assure that the probability attributed to the non-realizable outcomes,  $P(\gamma_l(\tau) < \gamma_l(t))$  for  $\tau > t$ , will be acceptably small. It will not always be the case that component health is strictly monotonically decreasing; however, consideration of this constraint serves to illustrate the flexibility provided by representing modeling uncertainties in terms of a quantized probability mass function (pmf), when compared to techniques that assume a continuous probability distribution function (pdf).

## 2.2. Formulating the action space of the fault growth process in terms of commanded and applied loads

Consider a process model for component damage accumulation that is expressed in terms of a metric representing the load or stress applied to a component at each control time-step:

$$\begin{aligned} p_{i,j}^l(u_l) = P(\gamma_l(k+1) = s_j | \gamma_l(k) = s_i, u_l = u_l(k)), \\ s_i, s_j \in S, \quad u_l \in U_l(k) \quad (6) \end{aligned}$$

where  $u_l$  represents the load applied to component  $l$  and  $U_l(k)$  represents the domain of feasible component load allocations for component  $l$  at time-index  $k$ . The component loading variable may represent pressure, force, torque, or a wide variety of other stressors that drive component damage.

Determination of  $U_l(k)$  at present and future control time-

steps will require a mapping function to translate system hardware limitations and estimated environmental loading conditions into the component loading domain. A mapping between available low-level control actions and feasible component loadings, as well as an inverse mapping are both assumed to be known:

$$\begin{aligned} F(\mathbf{a}(k), \mathbf{x}(k), \gamma(k), \mathbf{w}(k)) : \quad \mathbf{A}(k) \rightarrow \mathbf{U}(k) \\ F(\mathbf{u}(k), \mathbf{x}(k), \gamma(k), \mathbf{w}(k))^{-1} : \quad \mathbf{U}(k) \rightarrow \mathbf{A}(k) \quad (7) \end{aligned}$$

where  $F$  represents a mapping from a given system state, health state, exogenous demand state, and a set of low-level control actions available at time-index  $k$ , onto the domain of feasible component load allocations available at time-index  $k$ .

### 2.2.1. Performance constraints

In addition to being defined in terms of the hardware limitations of control effectors, the domain of feasible component load allocations will generally also be bounded by operational constraints on minimum allowable output performance and maximum allowable fault growth risk.

Consider a constraint on minimum allowable system performance that is defined in terms of a maximum allowable deviation from some commanded system state:

$$|\mathbf{x}_c(k) - \mathbf{x}(k)|_i \leq \Delta_i(k), \quad i \in \{1, 2, \dots, n\} \quad (8)$$

where  $\mathbf{x}$  and  $\mathbf{x}_c$  are  $n$  dimensional vectors that represent the actual and commanded output states of a system respectively, and  $\Delta_i$  specifies a maximum acceptable error between the  $i^{\text{th}}$  dimensions of  $\mathbf{x}$  and  $\mathbf{x}_c$ .

If the system's kinematics are known, then Newton's laws of motion can be applied to express the system's dynamics in terms of the instantaneous forces on its constituent components,

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{w}) \quad (9)$$

If the system is overactuated, then formulating the system's dynamics in terms of instantaneous component loads will result in actuation redundancies being identified by the nullspace of  $\mathbf{u}$ . Consider the following generic representation of nonlinear system dynamics:

$$\dot{\mathbf{x}} = A(\mathbf{x}, \mathbf{w}) + B(\mathbf{x}) \mathbf{u} \quad (10)$$

where  $A(\mathbf{x}, \mathbf{w}) \in \mathbb{R}^n$ ,  $B(\mathbf{x}) \in \mathbb{R}^{n \times q}$ ,  $\mathbf{x}(t) \in \mathbb{R}^n$ , and  $\mathbf{u}(t) \in \mathbb{R}^q$ . If  $B(\mathbf{x})$  does not have full column rank, i.e.,  $\text{rank}\{B(\mathbf{x})\} = p < q \quad \forall \mathbf{x}$ , then the system is overactuated, and  $B(\mathbf{x})$  can be factorized as:

$$B(\mathbf{x}) = B_\nu(\mathbf{x}) B_u(\mathbf{x}) \quad (11)$$

where  $B_\nu(\mathbf{x}) \in \mathbb{R}^{n \times p}$  and  $B_u(\mathbf{x}) \in \mathbb{R}^{p \times q}$  both have rank  $p$ .

The system can now be rewritten as:

$$\begin{aligned}\dot{\mathbf{x}} &= A(\mathbf{x}, \mathbf{w}) + B_\nu(\mathbf{x}) \boldsymbol{\nu} \\ \boldsymbol{\nu} &= B_u(\mathbf{x}) \mathbf{u}\end{aligned}\quad (12)$$

where  $\boldsymbol{\nu}(t) \in \mathbb{R}^p$  represents the net output control effort produced by the system's  $q$  components.

Inverting the dynamics given in Eq. (12) enables the performance constraint given in Eq. (8) to be expressed as a maximum allowable deviation from a given output control effort profile:

$$\begin{aligned}\mathbf{r} &= B_\nu(\mathbf{x}_c)^{-1} \cdot (\dot{\mathbf{x}}_c - A(\mathbf{x}_c, \mathbf{w})) \\ |\nu_i(k) - r_i(k)| &\leq \tilde{\Delta}_i(k), \quad i \in \{1, 2, \dots, p\}\end{aligned}\quad (13)$$

where  $\boldsymbol{\nu}$  and  $\mathbf{r}$  are  $p$  dimensional vectors that represent an actual and a desired net output force to be exerted by the system at a given time, and  $\tilde{\Delta}_i(k)$  specifies a maximum acceptable error between the  $i^{\text{th}}$  dimensions of  $\boldsymbol{\nu}$  and  $\mathbf{r}$ .

Because  $B_\nu(\mathbf{x})$  has full column rank, a system response,  $\dot{\mathbf{x}}$ , is uniquely determined by  $\boldsymbol{\nu}(t)$ ; however, if the system is overactuated, then the allocation of load among functionally redundant components may be specified by minimizing the aggregate component damages corresponding to load allocations in the nullspace of  $B_u(\mathbf{x})$ . Prognostics-based control in terms of component load allocations has been analyzed for an overactuated electro-mechanical actuator and an unmanned ground vehicle in previous publications (Bole et al., 2010, 2011).

### 2.2.2. Prognostic constraints

Prognostic constraints are typically specified in terms of a lower bound on the failure time of system components. Constraints on the minimum acceptable component failure time may be specified in terms of a maximum acceptable probability that the component will reach 0% health by a given time:

$$P(\gamma_l(t_M) = 0\% | \gamma_l(t_p), \mathbf{w}(t_p), u_l(t_p)) < \alpha_l \quad (14)$$

where  $t_p$  is the time at which the fault growth prediction is made and  $\alpha_l$  is an upper bound on the probability that component  $l$  is failed at time  $t_M$ .

If  $u_l$  was known over the domain  $t = [t_p, t_M]$ , then forward induction could be used to evaluate Eq. (14) from Eq. (6). Many publications on the topic of prognostics-based control opt to simplify the prognostics problem by assuming that component loadings will be unvarying over the prediction horizon:

$$u(t) = u(t_p) \quad \forall t \in [t_p, t_M] \quad (15)$$

However, in most cases, time-varying environmental loading conditions and time-varying component health estimates are expected to result in time-varying loadings on a system's

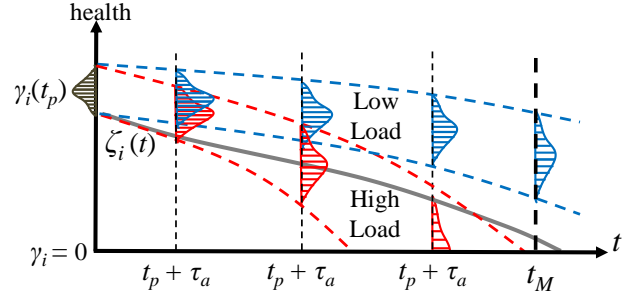


Figure 1. Uncertainty in fault growth predictions for high and low loads at various prognostic horizons

components. In such cases, the production of prognostic estimates with the highest degree of realism will require future component loadings to be modeled as a stochastic process that accounts for the statistics of all parameters affecting component load allocations within the controlled system.

Consider the drawing in Figure 1, the growth of uncertainty in component health estimates at prognostic horizons of increasing length is shown for two potential loading conditions, denoted as 'high load' and 'low load'. As shown in the figure, fault growth estimates at different loading conditions are expected to diverge over time. In cases where prognostic uncertainty becomes very large, due to high uncertainty on predictions of future component loading conditions, or high uncertainty on predictions of component damage as a function of loading profiles, the magnitude of prognostic uncertainty may be managed by limiting the length of the prediction horizon.

The specification of a lower bound on acceptable value at risk (VaR) assessments of system health over the range  $t = [t_p, t_M]$  may be used to enforce constraints on component longevity, using fixed horizon prognostic predictions. The VaR of a random variable  $X$  at a confidence level  $\psi$  is defined as:

$$VaR_\psi(X) = \inf \{x \in \mathbb{R} : P(X < x) > \psi\} \quad (16)$$

A finite horizon prognosis constraint is written as follows:

$$VaR_{\beta_l}(\gamma_l(t_p + \tau)) > \zeta_l(t_p + \tau) \quad (17)$$

where  $\tau$  specifies a time horizon at which prognostic constraints will be evaluated,  $\zeta_l$  specifies a lower bound on the depletion of component health over the range  $t = [t_p, t_M]$ , and  $\beta_l$  defines the maximum acceptable probability that the health of component  $l$  is less than  $\zeta_l$  at time  $t_p + \tau$ .

An example definition of  $\zeta_l(t)$  is given in Figure 1. The specification of an appropriate profile for  $\zeta_l(t)$  will be left as a design choice. Some general considerations for the specification of an appropriate profile include:

1. If component health is expected to be monotonically decreasing, then  $\zeta_l(t)$  must also be a monotonically de-

creasing function. Additionally, if it is ever the case that  $VaR_{\beta_l}(\gamma_l(t_p)) < \zeta_l(t_p + \tau)$ , then the prognostic constraint is unsatisfiable.

2. The greater the difference between  $VaR_{\beta_l}(\gamma_l(t_p))$  and  $\zeta_l(t_p + \tau)$ , the greater the control freedom allowed under the prognostic constraint. Online or a priori information could be used to make adjustments to  $\zeta_l(t)$  so as to allow for greater control freedom during higher priority time-periods at the expense of potentially incurring greater component health deterioration over those time-periods.

### 2.3. Fault prediction in terms of relative deviations from nominal system outputs

Consider the existence of a nominal control system, which would adequately control a system in the absence of any component degradation modes. This section introduces a metric to represent the relative deviation between the net output control effort that would have been exerted by the system on its environment if a nominal control law were used, and the net output control effort that is actually exerted by a given prognostics-based control routine at each control time-step.

The proposed metric is defined for each of the system's output degrees of freedom as:

$$\rho_l(k) = \frac{|\nu_l(k)|}{|\nu_l^*(k)|}, \quad l \in \{1, 2, \dots, p\} \quad (18)$$

where  $\nu^*$  and  $\nu$  represent the net output control effort that would have been commanded by a nominal control law and the net control effort output that is actually commanded by an implemented control routine at time-index  $k$ .

The Markov process defined in Eq. (6) is rewritten in terms of this new metric as:

$$\begin{aligned} p_{s_i, s_j}^l(\rho(k)) &= P(\gamma_l(k+1) = s_j | \gamma_l(k) = s_i, \rho = \rho(k)) \\ &= \int_{\nu_r \in N_r(k)} P(\nu_r(k) | \rho_r = \rho_r(k)) \cdot \end{aligned} \quad (19)$$

$$\begin{aligned} P(\gamma_l(k+1) = s_j | \gamma_l(k) = s_i, \mathbf{u}(k) = H(\nu(k))) & du \\ s_i, s_j \in S, u_l \in U_l(k), r \in \{1, 2, \dots, p\} \end{aligned}$$

where  $P(\nu_r(k) | \rho_r)$  can be estimated using available stochastic modeling of the future net output control effort demands on the system, and  $H(\nu(k))$  represents a mapping from a net system output force vector to a component loading vector. As described in Section 2.2.1, if no overactuation is present in the system, then component loadings are uniquely specified by a net system output force profile; however, if the system is overactuated, then the nullspace of the component loading domain can be resolved by an optimization routine that seeks to minimize the aggregate loss

of health among functionally redundant degrees of freedom. Note that all modeling of internal and external dynamics that affect component degradation are now described by the probability transition matrix  $p_{s_i, s_j}^l(\rho)$ . This formulation of fault growth dynamics provides a convenient means for analyzing the prognostics-based control problem, because it directly relates modifications to system output performance to predictions of component degradations.

The performance constraint, defined in terms of allowable net system output control effort in Eq. (13), is now expressed in terms of  $\rho$  as:

$$|1 - \rho_i(k)| \leq \bar{\Delta}_i(k), \quad i \in \{1, 2, \dots, p\} \quad (20)$$

where  $\bar{\Delta}_i$  defines a constraint on the maximum allowable deviation from a system's nominal control effort output in dimension  $i$  at time-index  $k$ .

A uniformly quantized state space for  $\rho_i$  under the performance constraint is defined as:

$$\begin{aligned} \rho_i(k) \in \Theta(k), \quad \Theta(k) &= [\theta_1, \theta_2, \dots, \theta_m], \\ \theta_1 &= 1 - \bar{\Delta}_i(k), \quad \theta_m = 1 + \bar{\Delta}_i(k) \end{aligned} \quad (21)$$

#### 2.3.1. An example of output control effort regulation

A simple example is considered here to examine the regulation of a system's net control effort output using the performance metric  $\rho$ . The example system to be regulated is a linear actuator attached to a simple mass-spring-damper system, defined by a mass  $m$ , a spring constant  $k$ , and a damping coefficient  $c$ :

$$m\ddot{x} = -kx - c\dot{x} + \nu \quad (22)$$

A nominal control law for the system is represented by the following proportional feedback equation:

$$\nu^* = k_p \cdot (x - x_c) \quad (23)$$

where  $x$  and  $x_c$  represent an actual and a commanded actuator position respectively,  $\nu$  and  $\nu^*$  represent the net actuator output force and that commanded by the nominal control law respectively, and  $k_p$  is a gain coefficient within the nominal controller. Diagrams of the mass-spring-damper system and the proposed control law are shown in Figure 2. Values for all variables in the controlled mass-spring-damper system are given in Table 1.

In this example, the net output force exerted by the controlled effectors on the environment and the performance metric used to regulate that output are both one dimensional:

$$\nu = \rho\nu^* \quad (24)$$

Figure 3 shows the behavior of the system as  $\rho$  is linearly varied over the domain  $[0.2, 1]$ . It should generally be expected that smaller values of  $\rho$  will induce greater errors in tracking



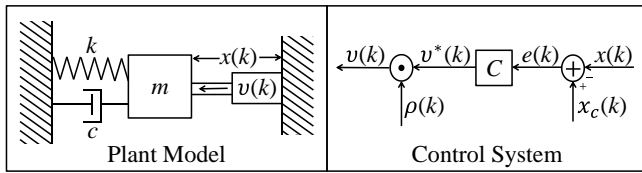


Figure 2. Diagram of mass-spring-damper plant model and control system

Coefficient	Value
m	1
k	2
c	5
$k_p$	15

Table 1. Coefficients used in mass-spring-damper simulation

a desired output profile, and these induced errors will result in greater net actuator output force demands by the nominal control law. Both of these features are clearly visible in the experimental results given in Figure 3. Another observation that can be made from the results given in Figure 3 is that while  $\rho < 1$  will always result in an apparent reduction in the net output control effort that would have been commanded by a nominal control law at a given control time-step, the error dynamics that are induced by lowering  $\rho$  will not necessarily result in lowered net system output control effort over a finite window of observation. A marginally increasing trend in the peak-to-peak actuator loads over a cycle is observed as  $\rho$  is lowered; this type of behavior illustrates the fact that achieving a good tradeoff between the reduction of component loading and induced errors in trajectory tracking will generally require closed loop regulation of  $\rho$ .

### 3. THE PROGNOSTICS-BASED CONTROL PROBLEM

The prognostics-based control problem can generally be viewed as an optimization problem, in which implemented control routines will select values of  $\rho_i$  at each control time-step in an attempt to minimize the risks posed by the application of load to degrading components, while also minimizing any deviation from a system's nominal performance. The problem of specifying appropriate metrics for assessing the risk posed by probabilistic prognostic predictions of future component health deterioration may generally be considered independently from the problem of building prognostic models. Assuming that discrete Markov modeling will provide a sufficiently accurate approximation of a system's fault dynamics, then the Markov process notation described in Section 2 could be used to evaluate any given risk metric for use in any given prognostics-based control method. Future work will address the specification of appropriate risk metrics and the use of this Markov process formulation for deriving and evaluating prognostics-based control policies on sample applications.

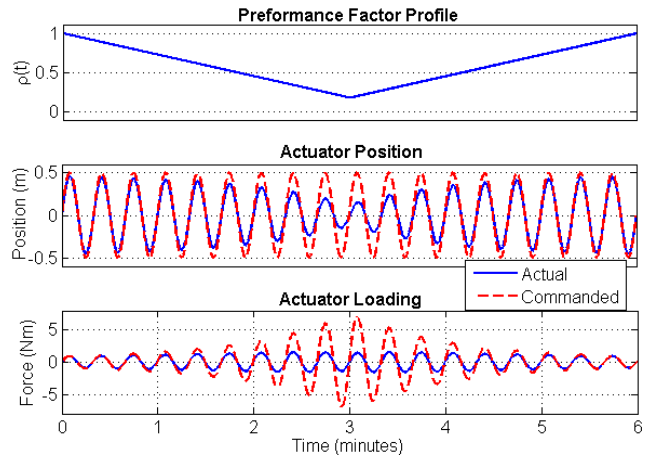


Figure 3. Plots showing position profile tracking and actuator loading dynamics as  $\rho$  is varied over the range  $[0.2, 1]$

If aversion to the potential degradation of a system's nominal output loading performance and aversion to the potential degradation of component SOH are adequately expressed in terms of an expectation of accumulated state transition costs, then the search for a control policy that optimizes a stochastic system of the form disclosed in Eq. (19) is expressible as a Markov decision process (MDP). MDPs are commonly used to analyze problems involving decision making in the presence of uncertain or stochastic information, and optimizing control policies may be identified using well studied MDP optimization techniques such as backwards induction for finite horizon optimization problems, and linear programming, value iteration, and policy iteration for discounted and average-reward infinite horizon optimization problems. The requirement that fault risk must be expressed as an expectation of accumulated state transition costs over a finite or infinite horizon may seem to be an overly restrictive constraint on the general problem of quantifying risk; however, over the past several decades much has been published on the theory of encoding various forms of risk aversion into the specification of MDP state transition costs (Hernandez & Marcus, 1996; Ruszczyrski, 2009).

### 4. CONCLUSIONS

This paper introduced a novel Markov process representation of component health dynamics. A metric representing the relative deviation between instantaneous samples of the net output force that would have been enacted by a nominally controlled system, and the net system outputs that are actually enacted by an implemented prognostics-based control routine at each time-step, was used to define the action space of the Markov process. The state space of the proposed Markov process formulation was defined in terms of an abstracted metric representing the relative health remaining in each of the system's components. Operational constraints on minimum

acceptable system output performance and maximum acceptable fault growth risk were formalized, and the mappings necessary to impose those constraints on the domain of feasible control actions available to implemented prognostics-based control routines were specified. Arguments were provided for the potential convenience and robustness of the proposed notation for evaluating prognostics-based controllers.

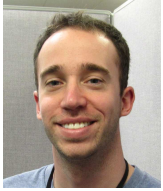
#### ACKNOWLEDGMENT

This work was supported by the United States National Aeronautics and Space Administration (NASA) under the Graduate Student Research Program (GSRP). The GSRP provides funding for Brian Bole to perform graduate research concerning the development and implementation of prognostics based reasoning techniques, under the direction of Dr George Vachtsevanos, Professor Emeritus at the Georgia Institute of Technology, and Dr Kai Goebel, Director of the Prognostics Center of Excellence at NASA AMES.

#### REFERENCES

- Banjevic, D., & Jardine, A. (2006). Calculation of reliability function and remaining useful life for a Markov failure time process. *IMA Journal of Management Mathematics*, 17, 115-130.
- Baruah, P., & Chinnam, R. B. (2005). HMMs for diagnostics and prognostics in machining processes. *International Journal of Production Research*, 43(6), 1275-1293.
- Bole, B., Brown, D. W., Pei, H.-L., Goebel, K., Tang, L., & Vachtsevanos, G. (2010, Oct.). Fault adaptive control of overactuated systems using prognostic estimation. In *Annual conference of the prognostics and health management society*.
- Bole, B., Tang, L., Goebel, K., & Vachtsevanos, G. (2011). Adaptive load-allocation for prognosis-based risk management. In *Annual conference of the prognostics and health management society*.
- Dong, M., & He, D. (2007). Hidden semi-Markov model-based methodology for multi-sensor equipment health diagnosis and prognosis. *European Journal of Operational Research*, 178, 858-878.
- Edwards, D., Orchard, M., Tang, L., Goebel, K., & Vachtsevanos, G. (2010). Impact of input uncertainty on failure prognostic algorithms: Extending the remaining useful life of nonlinear systems. In *Annual conference of the prognostics and health management society*.
- Guidaa, M., & Pulcini, G. (2011). A continuous-state Markov model for age- and state-dependent degradation processes. *Structural Safety*, 33(6), 354-366.
- Hauriea, A., & Moresino, F. (2006). A stochastic control model of economic growth with environmental disaster prevention. *Automatica*, 42(8), 1417-1428.
- Hernandez, D., & Marcus, S. (1996). Risk sensitive control of Markov processes in countable state space. *Systems & Control Letters*, 29, 147-155.
- Orchard, M., Kacprzyński, G., Goebel, K., Saha, B., & Vachtsevanos, G. (2008). Advances in uncertainty representation and management for particle filtering applied to prognostics. In *Annual conference of the prognostics and health management society*.
- Parlara, M., Wang, Y., & Gerchak, Y. (1995). A periodic review inventory model with Markovian supply availability. *International Journal of Production Economics*, 42(2), 131-136.
- Ruszczyski, A. (2009). Risk-averse dynamic programming for Markov decision processes. In *20th international symposium on mathematical programming*.
- Saha, B., Goebel, K., Poll, S., & Christophersen, J. (2009). Prognostics methods for battery health monitoring using a Bayesian framework. *IEEE Transactions on Instrumentation and Measurement*, 58(2), 291-296.
- Sankararaman, S., Ling, Y., Shantz, C., & Mahadevan, S. (2009). Uncertainty quantification in fatigue damage prognosis. In *Annual conference of the prognostics and health management society*.
- Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., & Schwabacher, M. (2008). Metrics for evaluating performance of prognostic techniques. In *Annual conference of the prognostics and health management society*.
- Serfozo, R. F. (1979). An equivalence between continuous and discrete time Markov decision processes. *Operations Research*, 27, 616-620.
- Shetty, P., Mylaraswamy, D., & Ekambaram, T. (2008). A hybrid prognostic model formulation and health estimation of auxiliary power units. *Journal of engineering for gas turbines and power*, 130(2).
- Smilowitz, K., & Madanat, S. (1994). Optimal inspection and repair policies for infrastructure facilities. *Transportation science*, 28, 55-62.
- Sonnenberg, F., & Beck, R. (1993). Markov models in medical decision making. *Medical Decision Making*, 13(4), 322-338.
- Tang, L., Kacprzyński, G. J., Goebel, K., & Vachtsevanos, G. (2009). Methodologies for uncertainty management in prognostics. In *IEEE aerospace conference*.
- Wang, H. S., & Chang, P.-C. (1996). On verifying the first-order Markovian assumption for a Rayleigh fading channel model. *IEEE Transactions on Vehicular Technology*, 45(2), 353-357.
- Wang, P., Youn, B., & Hu, C. (2012). A generic probabilistic framework for structural health prognostic and uncertainty management. *Mechanical Systems and Signal Processing*, 28, 622-637.
- Zhang, Y., & Jiang, J. (2008). Bibliographical review on reconfigurable fault-tolerant control systems. *Annual Reviews in Control*, 32(2), 229-252.

## BIOGRAPHIES



**Brian M. Bole** graduated from the FSU-FAMU School of Engineering in 2008 with a B.S. in Electrical and Computer Engineering and a B.S. in Applied Math. Brian received a M.S. degree in Electrical Engineering from the Georgia Institute of Technology in 2011, and he is currently pursuing a Ph.D. Brian's research interests

include: analysis of stochastic processes, risk analysis, and optimization of stochastic systems. Brian is currently investigating the use of risk management and stochastic optimization techniques for optimal adaptation of active component load allocations in robotic and aviation applications. In a previous project, Brian worked with the Georgia Tech EcoCar team to develop an energy management controller for optimizing the fuel economy of a charge sustaining hybrid electric vehicle.



**Kai Goebel** received the degree of Diplom-Ingenieur from the Technische Universität München, Germany in 1990. He received the M.S. and Ph.D. from the University of California at Berkeley in 1993 and 1996, respectively. Dr. Goebel is a senior scientist at NASA Ames Research Center where he leads the Diagnostics & Prognostics groups

in the Intelligent Systems division. In addition, he directs the Prognostics Center of Excellence and he is the Associate Principal Investigator for Prognostics of NASA's Integrated Vehicle Health Management Program. He worked at General

Electric's Corporate Research Center in Niskayuna, NY from 1997 to 2006 as a senior research scientist. He has carried out applied research in the areas of artificial intelligence, soft computing, and information fusion. His research interest lies in advancing these techniques for real time monitoring, diagnostics, and prognostics. He holds ten patents and has published more than 100 papers in the area of systems health management.



**George J. Vachtsevanos** is a Professor Emeritus of Electrical and Computer Engineering at the Georgia Institute of Technology. He was awarded a B.E.E. degree from the City College of New York in 1962, a M.E.E. degree from New York University in 1963 and the Ph.D. degree in Electrical Engineering from the City University of New

York in 1970. He directs the Intelligent Control Systems laboratory at Georgia Tech where faculty and students are conducting research in intelligent control, neurotechnology and cardiotechnology, fault diagnosis and prognosis of large-scale dynamical systems and control technologies for Unmanned Aerial Vehicles. His work is funded by government agencies and industry. He has published over 240 technical papers and is a senior member of IEEE. Dr. Vachtsevanos was awarded the IEEE Control Systems Magazine Outstanding Paper Award for the years 2002-2003 (with L. Wills and B. Heck). He was also awarded the 2002-2003 Georgia Tech School of Electrical and Computer Engineering Distinguished Professor Award and the 2003-2004 Georgia Institute of Technology Outstanding Interdisciplinary Activities Award.

# Condition Indicators for Gears

Dr. Renata Klein

*R.K. Diagnostics, Gilon, POB 101, D.N. Misgav, 20103, Israel*

*renata.klein@rkdiagnostics.co.il*

## ABSTRACT

Diagnostics of faults in gears requires development of reliable condition indicators. A large number of condition indicators, which are based on statistical moments of the synchronous average and its derivatives (difference and residual signals) were previously suggested. This study evaluates the efficiency of different gear condition indicators that are based on statistical moments and compares them with two new types of condition indicators that are suggested. The two new types of condition indicators are based on the order spectrum and the spectral kurtosis of the synchronous average.

The study was conducted on the labeled data of PHM'09 challenge. This data included recordings of vibrations in helical and spur gearboxes with seeded faults.

## 1. INTRODUCTION

Vibration based diagnostics of gears based on statistical moments of the synchronous average was investigated before (Dempsey & Zakrajsek 2001, Lebold, McClintic, Campbell, Byington & Maynar, 2000, Mosher, Pryor & Huff, 2002, Zacksenhous, Braun, Feldman & Sidahmed, 2000). Many different condition indicators have been proposed for different types of gear malfunctions. An automatic diagnostic process for gears requires selecting a set of the relevant condition indicators and aggregating them into health indicators.

The selection of condition indicators is based on their ability to detect the faults with minimum false alarms, i.e. their ability to differentiate between faulty and healthy states with the maximum contrast, and if possible to indicate the type of fault. Hence, the evaluation of condition indicators should be conducted with healthy recordings as well as recordings of different seeded faults.

The evaluation of the condition indicators was performed using the labeled data from PHM'09 challenge. It contains

records of both spur and helical gears, split between two load levels, healthy or with three types of faults: broken tooth, chipped tooth, and eccentricity. The presence of different faults allows selection of condition indicators covering more than one type of fault.

The data that was used is briefly described in chapter 2. Chapter 3 presents the signal processing procedure, and chapter 4 describes the condition indicators that were evaluated. Chapter 5 presents and discusses the results.

## 2. PHM'09 CHALLENGE LABELED DATA

The PHM'09 data set included 280 recordings of 4 seconds each, measured on the gearbox described in Figure 1 (from Klein, Rudyk, Masad & Issacharoff, 2011b), using two vibration sensors (Sin and Sout) and a tachometer. All the bearings were similar. Some of the signals were recorded when the gearbox was in 'spur' configuration, and others when it was in 'helical' configuration. Data were collected at 30, 35, 40, 45 and 50 Hz shaft speeds, under high and low loading (HL and LL) (Klein, Rudyk, Masad & Issacharoff, 2011b).

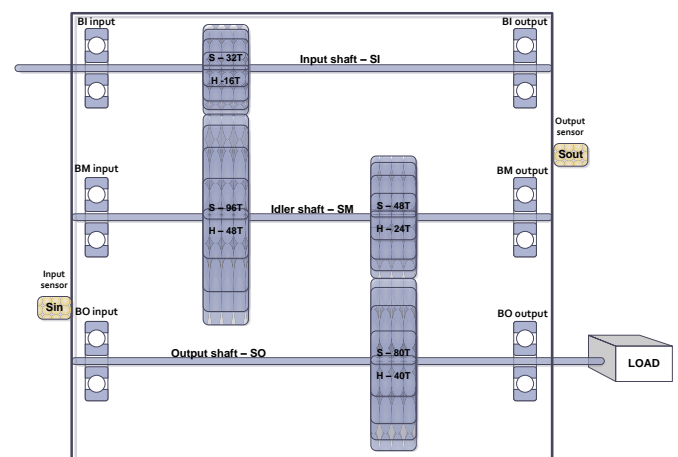


Figure 1. Challenge apparatus: spur (S) and helical (H) configurations

In the challenge apparatus, in spur and helical configurations, the idler shaft (SM) and the output shaft

Renata Klein. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

(SO) rotated at 1/3 and 1/5 of the rotating speed of the input shaft (SI) correspondingly. The gear ratios generated overlapping characteristic frequencies that made the separation between the manifestations of different gearwheels especially challenging.

Table 1 summarizes the recordings of the PHM'09 data set and the damages that were present on the gears.

Case	Gears				Bearings			Shaft	
	32T	96T	48T	80T	bSI	bSM	bSO	Input	Output
Spur 1	Good		Good						
Spur 2	Chipped		Eccentric	Good		Good			
Spur 3			Eccentric						Good
Spur 4	Good		Eccentric	Broken	Ball	Good	Good		
Spur 5	Chipped	Good	Eccentric	Broken	Inner	Ball	Outer		
Spur 6	Good		Good	Broken	Inner	Ball	Outer	Imbalance	Good
Spur 7	Good		Good		Inner	Good	Good	Good	Key
Spur 8	Good		Good		Good	Ball	Outer	Imbalance	Good
	16T	48T	24T	40T					
Helical 1									
Helical 2			Chipped						
Helical 3			Broken		Comb	Inner		Bent	Good
Helical 4	Good		Good	Good	Comb	Ball		Imbalance	Good
Helical 5			Broken			Inner			
Helical 6								Bent Shaft	

Table 1. PHM'09 challenge dataset faults

### 3. GEAR SIGNAL ANALYSIS

The most widely used signatures in gear analysis are computed in the order, the quefrequency of orders and the cycle domains (Klein et al. 2011b, Lebold et al. 2000, Zacksenhouse et al. 2000). The time history is mapped into the cycle domain after synchronization (resampling) according to the shaft rotating speed.

The cepstrum of the orders representation was generated as follows:

$$C_v := \Re\{\mathcal{F}^{-1}[\ln(\mathcal{F}[v(t)])]\} \quad (1)$$

where:  $C_v$  is the real cepstrum of  $v(t)$ .

The cepstrum reflects the repetition rate of the sidebands (due to frequency modulation) and their average level in several peaks in the quefrequency of orders domain (Zacksenhouse et al. 2000, Antoni & Randall 2002).

The separation of the vibrations that belong to a certain gearwheel is achieved by calculating the synchronous average according to the respective shaft speed (Antoni & Randall 2002, Mosher et al. 2002, Zacksenhouse et al. 2000). Averaging is applied to enhance deterministic effects synchronized with the rotation of the relevant shaft. The synchronous average signal reveals the vibration induced by the meshing of each tooth on the relevant gearwheel.

The synchronous average removes the asynchronous components by averaging the resampled signal in each rotation cycle. All the signal elements that are not in phase with the rotation speed are eliminated, leaving the periodic elements present in one cycle, i.e. the elements corresponding to the harmonics of the shaft rotating speed (Klein, Rudyk, Masad & Issacharoff, 2011a).

Synchronous average with frequency  $f$  is designed to remove elements in  $v$  which are not periodic with the period  $N=1/f$ .

$$y_n := \frac{1}{M} \sum_{m=0}^{M-1} v_{n+mN} \quad n = 1, \dots, N \quad (2)$$

Note that  $y$  is a vector in  $\mathbb{R}^N$  representing a single cycle.

The synchronous average ability to filter out the asynchronous elements depends on the number of cycles averaged (in this case  $M$ ). Therefore, it is preferable to average as many cycles as possible. In the case of the PHM recordings, the number of cycles averaged differed pending on the shaft rotating speed. For the input shaft, the number of cycles averaged was 120-200 depending on the rotating speed (RPS of 30-50Hz respectively). The number of cycles averaged for the idler shaft was in the range 40-66, and for the output shaft in the range of 24-40.

Detection of abnormal meshing of an individual tooth is achieved by further processing of the averaged signal in three types of signals: regular, residual, and difference (Klein et al. 2011b, Mosher et al. 2002, Zacksenhouse et al. 2000). The regular signals are obtained by passing the synchronous-averaged signal through a multi-band-pass filter, with pass-bands centered at the meshing frequency and its harmonics (1÷5). It is essentially the cycle-domain average of the vibrations induced by a single tooth. The residual signals are obtained by removing the meshing frequency harmonics. The difference signals are obtained by removing the meshing frequencies and the adjacent sidebands (in that study, two sidebands have been removed), i.e. the frequency and amplitude modulations were separated.

The envelopes of the regular, difference and residual parts of all the harmonics describe the characteristics of the amplitude modulation.

In order to better separate the excitations from the gearwheels on different shafts, the synchronous averages according to the idler and output shafts were calculated using a dephased signal (Klein et al. 2011a) by the input shaft (see Figure 16). As a result, the integer multiples of the input shaft that coincide with every third multiple of the idler shaft and every fifth multiple of the output shaft were removed, leaving only partial information related to the respective gearwheels but isolating the excitations from the gearwheel on the input shaft.

The spectral kurtosis denoted SK of the synchronous average according to every shaft was calculated according to equation 3 (Antoni, 2006).

$$SK = K_y(f) := \frac{S_{4y}(f)}{S_{2y}^2(f)} - 2, \quad f \neq 0 \quad (3)$$

where:  $S_{2ny}(f) = \langle S_{2ny}(t, f) \rangle_t = E \{ |S_y(t, f)|^{2n} \} / df$  is the average  $2n$  power of the spectrum,  $f$  denotes frequency, and  $t$  denotes time. For  $n=1$ , we obtain the power spectral density.

The spectral kurtosis is a statistical tool, which can indicate the presence of transients and their location in the frequency domain (in our case order domain). The SK provides a robust way of detecting incipient faults even in the presence of a strong masking noise (Antoni & Randall 2004).

In order to diagnose each gearwheel separately, the spectral kurtosis was not calculated on the raw or resampled signals as it is usually done. Instead, it was calculated on the synchronous average according to each shaft (see Figure 16). This approach was especially needed for the PHM'09 challenge data because in most of the recordings with faulty gears, there were other faulty components that could affect the spectral kurtosis (e.g. bearings, see Table 1). The spectral kurtosis was calculated on windows of  $\frac{1}{4}$  cycles with 50% overlapping, generating a spectrum with resolution of four orders. The averaging of windows corresponding to  $\frac{1}{4}$  cycle was a compromise to overcome the problem of short recordings.

When longer recordings are available, (e.g. more than 200 cycles for all the operating conditions), it is recommended to calculate several synchronous averages on running windows of more than 20 cycles, and then calculate the spectral kurtosis on windows corresponding to one cycle.

#### 4. FEATURE EXTRACTION

The proposed process for feature extraction is in essence a comparison of the analyzed signatures to the "baseline" population (a model of normality), i.e. determination of the Mahalanobis distance between the analyzed signature and the baseline (Klein, Rudyk & Masad, 2011c).

The usage of distance signatures generates features (condition indicators) in terms of distance from the "normal" in  $\sigma$  units, allowing application of generic decision and fusion algorithms for each type of component. The weakness of the process is in the hidden assumption of Gaussian distribution of the values in the signatures when using Mahalanobis distances. As will be shown below, some methods will be used to overcome this weakness.

##### 4.1. Baselines

Baselines are signatures derived from healthy recordings, in the considered domains. Each set of such signatures represents the statistics of healthy signatures in a certain predefined operating mode. The baselines are usually the spectra (orders) or statistical moments, representing average, variance, median and estimator of the standard deviation (normalized IQR inter-quartile range  $\tilde{\sigma} = IQR \cdot 0.7413$ ) of the signatures that compose it.

For the gear diagnostics of the PHM'09 data, the baseline signatures were generated using an expanded set of records that included all the records of healthy machines plus all the records in which the gears were healthy but not necessarily the bearings or the shafts. This was done to enhance the quality of the baselines by including as many records as possible. The justification for this approach emerges from the facts that it is possible to screen out the excitations from bearing faults by using the synchronous average, and it is possible to screen out the excitations from shaft faults by using special filters.

##### 4.2. Gear Features

Generally, gear meshing components, some low-order amplitude modulation components and/or phase modulation components dominate the synchronous average. These modulation effects are generated by transmission errors related to geometric and assembly errors of the gear pairs. When a localized gear fault is present, a short period impulse will appear in each complete revolution. This produces additional amplitude and phase modulation effects. Due to its short period, the impulse produces high order low-amplitude sidebands surrounding the meshing harmonics in the spectrum. The removal of the regular gear meshing harmonics (residual part) sometimes with their low-order sidebands (difference part) from the synchronous average emphasizes the portion predominantly caused by gear fault and geometrical and assembly errors. Statistical measures of the difference and residual parts are used to quantify the fault-induced shocks, e.g. condition indicators like FM0, FM2, FM4, FM6, FM8, NA4, NB4, etc. (Lebold et al. 2000, Mosher et al. 2002, Dempsey & Zakrajsek, 2001).

The short impulse generated in each revolution is expected to be revealed in the spectral kurtosis of the synchronous average at the orders corresponding to the frequency contents of the impulse. It should be mentioned that high levels are not expected at the gearmesh or sidebands orders and probably will be only manifested at high orders where the spectrum levels are relatively low (Antoni, 2006).

##### 4.3. Gear Feature Extraction for PHM'09 data

The features (condition indicators) of gears were extracted for both the spur and helical configurations based on the distance signatures.

The features extracted for each gear wheel in the cycles domain included: the even statistical moments (RMS and kurtosis) of the regular, residual and difference parts, the even moments of the envelope of the regular, residual and difference.

The distances of the cepstrum representing the orders-quefrequency of the respective shafts where extracted as well.

The spectral kurtosis distances from the baselines according to each shaft were calculated. The purpose of this was to investigate their ability to detect gear faults. The condition indicator extracted from the spectral kurtosis distance signatures was the sum of values exceeding  $3\sigma$ .

The distance signatures of the order representations of the synchronous average according to each shaft were generated. The peak values at each harmonic of the shaft rotating speed (representing gearmesh orders and sidebands) were stored as features characterizing the gears. These peak values distances have been separated into three categories and grouped by their corresponding harmonics (Figure 2):

- AM – the sidebands representing amplitude modulation, i.e. the two low-order sidebands around the gear meshing orders
- FM – the sidebands representing frequency modulation, i.e. high-order sidebands around gear meshing orders
- GM – the gear meshing orders

The average distances  $D_h$  of the FM and AM peaks per every harmonic  $h$  were defined:

$$D_h = \frac{\sum_{i=1}^N (Y_i - \mu_i)}{\sqrt{\sum_{i=1}^N \sigma_i^2}} \quad (4)$$

where:  $i$  is the index of the peak,  $N$  is the pattern size (total number of sidebands considered for the FM or AM category in harmonic  $h$ ),  $Y_i$  – the peak levels,  $\mu_i$  – the corresponding mean level in the baseline, and  $\sigma_i$  – the corresponding standard deviation of the baseline. Because the average was used, it was possible to overcome the potential problem of non-Gaussian distributions (according to the Central Limit theorem, sums of variables tends to have a Gaussian distribution).

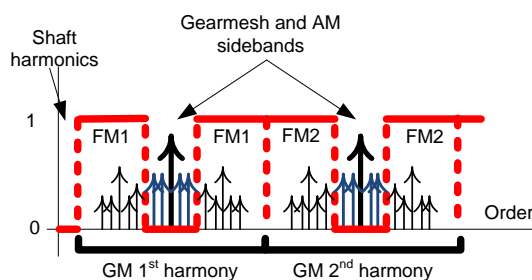


Figure 2. Illustration of the categories of peaks extracted in the orders of the synchronous average: FM1, FM2 denotes the group of sidebands of frequency modulation of the corresponding harmonic, the AM sidebands are drewed with blue arrows and the GM peaks with heavy black arrows.

## 5. CONDITION INDICATORS EVALUATION

It should be pointed out that in the PHM apparatus, it was especially difficult to identify the exact location of the

faults, because the number of teeth on the gearwheels were exact multiples of each other. Since the gear mesh orders were exact multiples, demodulation was impractical. All the signatures and features were affected by mechanisms associated with the cross-gear interference (vibrations induced by one gear are modulated by vibrations of another gear). For instance, a fault in the input pinion gear may cause a modulation of the tooth meshing frequency with the tooth meshing of the large gearwheel, resulting in an erroneous identification of the fault location.

The features that have been evaluated and compared are averaged AM and FM modulations, the even statistical moments of the regular synchronous average, residual and difference signals, and the spectral kurtosis exceptions from baseline.

The kurtosis of all the signals did not reveal any type of fault in the PHM'09 dataset.

The RMS of the regular, difference, residual signals and their envelopes revealed similar detection abilities. Because of that, only the results of the RMS of the difference signal are shown and discussed.

Except for the spectral kurtosis, the evaluation was performed on features corresponding to the first three harmonics of the gearmesh only (Figure 2). We found out that the features of the higher harmonics did not contain additional meaningful fault detection information.

The following graphs present results for all the records in a specific configuration (spur or helical). The colors in the graphs are proportional to the distances from the baseline, usually in the range  $6-30\sigma$  (a color bar is displayed on each figure). The vertical axis represents recordings corresponding to different rotating speed and low or high loads, denoted by the nominal rotating speed and HL for high load or LL for low load. The horizontal axis represents the different records for the specific configuration.

### 5.1. Helical configuration

In the helical configuration, the gearwheel with 48 teeth (GM2) on the idler shaft (SM) had a chipped tooth in run 2, and a broken tooth on runs 3 and 5. GM2 features corresponding to the first harmonic are displayed in Figure 3, Figure 4, and Figure 5. The results for higher harmonics were similar and therefore not displayed.

As can be observed in Figure 3 and in Figure 4, the RMS and the average distance of the AM and FM peaks were not able to detect the faulty gearwheel. In Figure 3, results of the FM and AM averages of the input sensor are presented separately showing that both were not able to reveal the faulty gearwheel.

Only the spectral kurtosis seems to reveal the faulty gearwheel reliably (the color range in Figure 5 is

approximately  $6-100\sigma$  for both sensors). The levels of the spectral kurtosis sum are higher for the high load records.

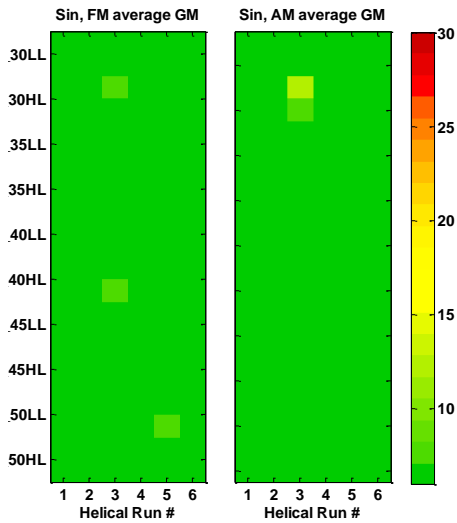


Figure 3. Average FM and AM sidebands, gearwheel GM2, harmonic 1, sensor Sin, helical runs: left average distance of FM sidebands, right average distance of AM sidebands.

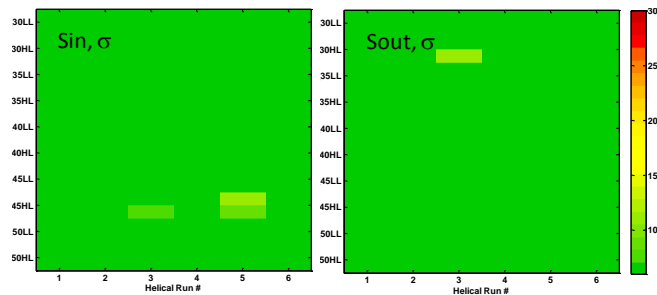


Figure 4. RMS of difference signal, harmonic 1, gearwheel, GM2, helical runs: left sensor Sin, right sensor Sout.

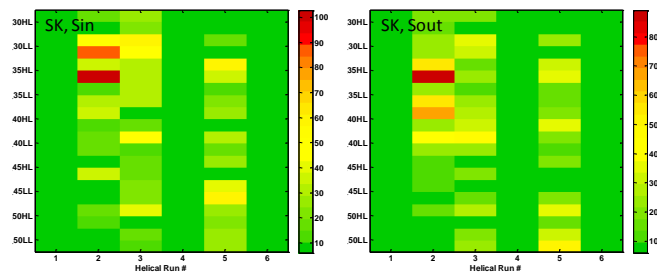


Figure 5. Sum of SK distances, gearwheel GM2, helical runs: left sensor Sin, right sensor Sout.

In general, the gear faults in the helical configuration displayed lower levels in all types of features and therefore were more difficult to detect. Usually, faults in helical gears can be better detected if the accelerometers are positioned in

the axial direction. Unfortunately, both sensors in the PHM'09 dataset were radial.

## 5.2. Spur configuration

In spur configuration, runs 2-6 contained gear faults, mostly combinations of two or more faulty gearwheels.

### 5.2.1. Input gear results

In the spur configuration, one tooth of the input gearwheel (GI) was chipped in runs 2 and 5. The analysis results are presented in Figure 6, Figure 7, and Figure 8.

As can be observed in Figure 6, the average distance of the AM and FM sidebands allows good detection of the faulty runs (2 and 5) mainly in runs with high loads. Run 2 displayed lower distance values (probably because in run 5 there were 3 faulty gears that interfered with each other while in run 2 there were only two faulty gears). Run 2 was better detected in the sensor at the output (Sout).

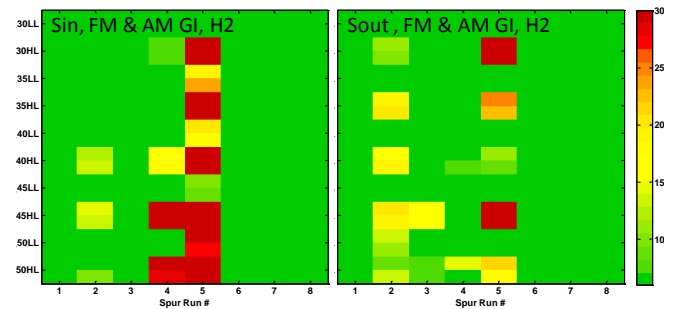


Figure 6. Average FM and AM sidebands, gearwheel GI, 2<sup>nd</sup> harmonic, spur runs: left sensor Sin, right sensor Sout.

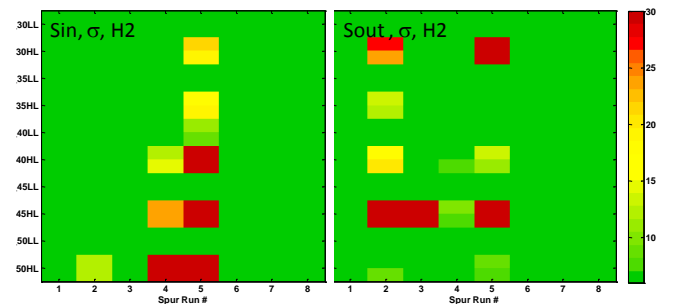


Figure 7. RMS of difference signal, 2<sup>nd</sup> harmonic, gearwheel GI, spur runs: left sensor Sin, right sensor Sout.

As can be observed in Figure 7, the RMS of the difference signal was elevated in runs 2 and 5 at high load runs but it was not consistent over the different rotating speeds.

The spectral kurtosis (Figure 8) provided a good discrimination ability (the color scale is  $6-80\sigma$ , different from the  $6-30$  in all the other graphs). It was elevated



mainly for runs 2, 5, and 6 but it had high levels in all the runs with gear faults.

The cross-gear interference (figure not displayed) was accentuated in the first harmonic of both the RMS of the difference signal and in the average FM and AM sidebands, especially in runs 4 and 5.

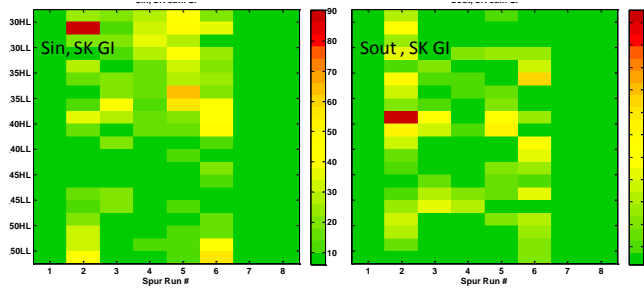


Figure 8. Sum of SK distances, gearwheel GI, spur runs: left sensor Sin, right sensor Sout.

In the spectral kurtosis results, the effect of the cross-gear interference was very clear in all the runs with a faulty gear. Apparently, the spectral kurtosis is a reliable indicator of the presence of a fault without the power to identify its exact source.

In the special case of the PHM apparatus, for the gearwheel on the input shaft, strong cross-gear interference is expected. This is because the sidebands of faulty gearwheels on the idler or output shaft always coincide with the sidebands corresponding to the input shaft.

### 5.2.2. Idler gear results

In the spur configuration, the 48 teeth gearwheel on the idler shaft (GM2) was eccentric in runs 2-5. The results are presented in Figure 9, Figure 10, Figure 11, and Figure 12.

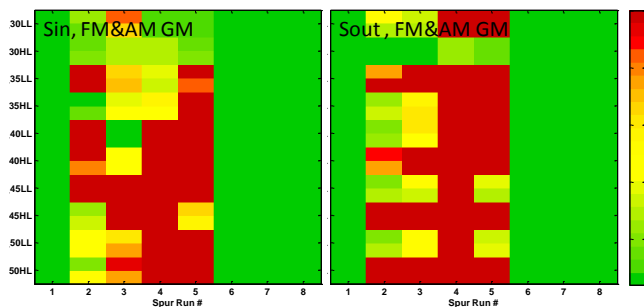


Figure 9. Average FM and AM sidebands, gearwheel GM2, 1<sup>st</sup> harmonic, spur runs: left sensor Sin, right sensor Sout.

As can be observed in Figure 10 the RMS of the difference signal is elevated in runs 2-5, mainly at high loads.

As can be observed in Figure 9, the average distance of the AM and FM peaks allows detection of the faulty runs (2-5).

Moreover, when inspecting the separate averages for FM and AM modulations (Figure 11) it can be observed that the AM average is especially high.

Similar results have been obtained for harmonics 1-3 for both the RMS of the difference and the average FM and AM sidebands.

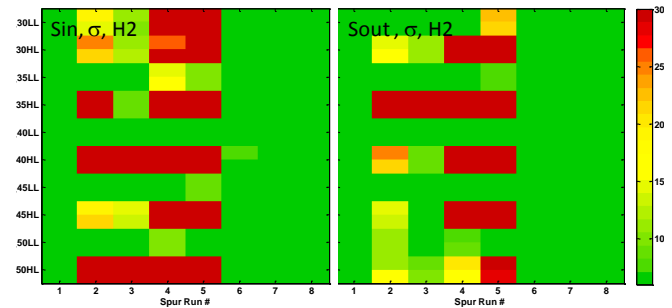


Figure 10. RMS of difference signal, 2<sup>nd</sup> harmonic, gearwheel GM2, spur runs: left sensor Sin, right sensor Sout.

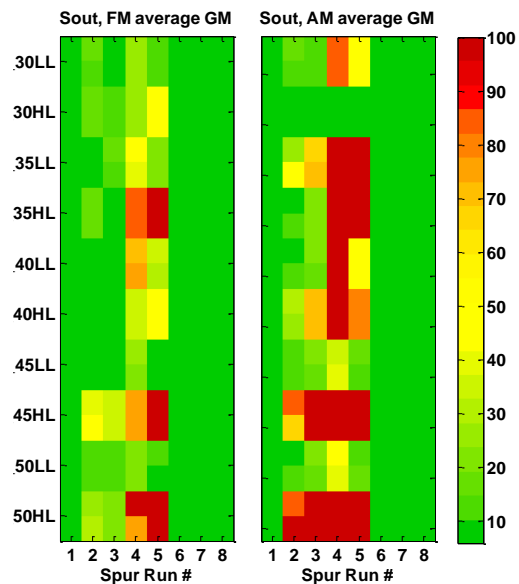


Figure 11. Average FM and AM sidebands, gearwheel GM2, harmonic 1, sensor Sout, spur runs: left average distance of FM sidebands, right average distance of AM sidebands.

Again, the spectral kurtosis (Figure 12) displayed good discrimination ability (note that the color scale is approximately 6-250 $\sigma$ ). It is elevated mainly for runs 2-5 and with lower levels in run 6.

All the evaluated condition indicators can detect the eccentric gearwheel with a good discrimination power. The average of the FM and AM sidebands provided detailed

information about the fault nature (high amplitude modulation at few harmonics).

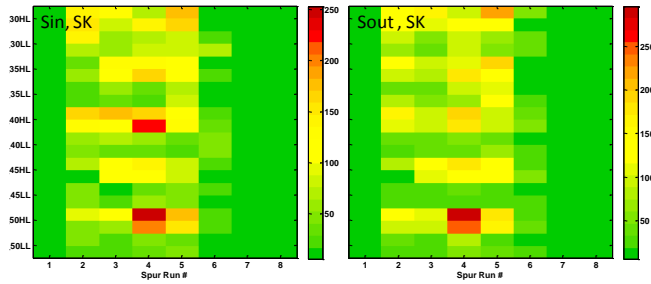


Figure 12. Sum of SK distances, gearwheel GM2, spur runs: left sensor Sin, right sensor Sout.

### 5.2.3. Output gear results

In the spur configuration, the gearwheel on the output shaft (GO) had a broken tooth in runs 4-6. The results are presented in Figure 13, Figure 14, and Figure 15.

As can be observed in Figure 13, the RMS of the difference signal was elevated in runs 2-5 mainly at high RPMs and high loads and only in the output sensor Sout. It meant that the findings of the RMS were not good enough to detect the broken tooth. The results were influenced by inter-component interaction and not from the faults of the output gearwheel.

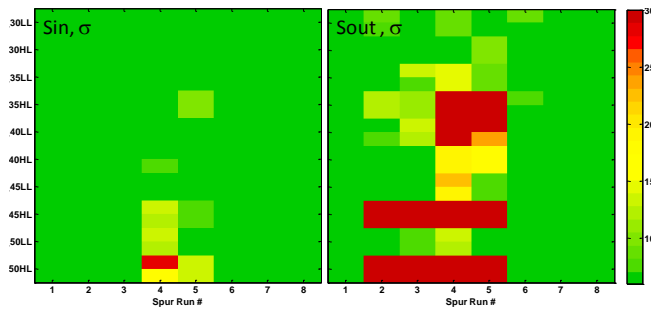


Figure 13. RMS of difference signal, 1<sup>st</sup> harmonic, gearwheel GO, spur runs: left sensor Sin, right sensor Sout.

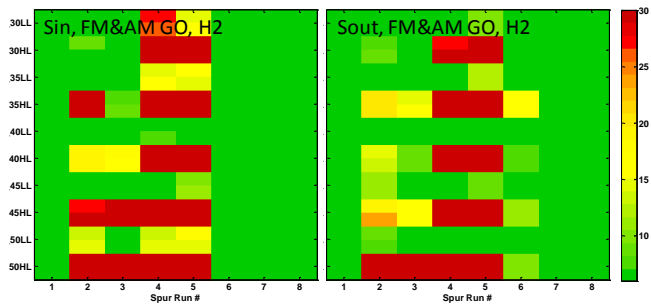


Figure 14. Average FM and AM sidebands, gearwheel GO, 2<sup>nd</sup> harmonic, spur runs: left sensor Sin, right sensor Sout.

In Figure 14, the average distance of the AM and FM sidebands allowed detection of the faulty runs 4 and 5. In sensor Sout some traces can be observed in run 6.

Run 6 had only one gear fault, the broken tooth of the output gearwheel (GO), which was not detected in the average distance of the AM and FM sidebands. Therefore, it is suspected that the high levels of the average distance of the FM and AM sidebands in runs 4 and 5 represent mainly the cross-gear interference with the eccentricity of the idler gearwheel (GM2), while the effect of the broken tooth is of low level.

The spectral kurtosis (Figure 15) provided a good discrimination capability (the color scale is approximately  $6-180\sigma$ ). It is elevated mainly for runs 4 and 6.

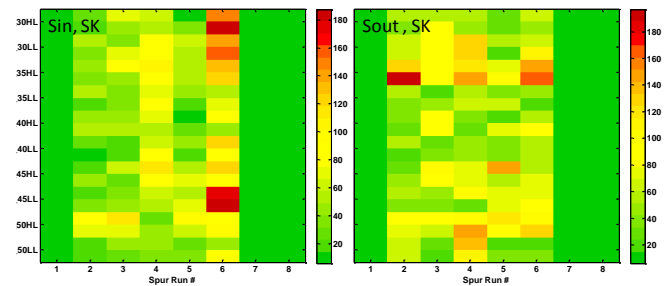


Figure 15. Sum of SK distances, gearwheel GM2, spur runs: left sensor Sin, right sensor Sout.

## 6. CONCLUSIONS

The evaluation of condition indicators for gear diagnostics was carried out on the PHM'09 challenge dataset in both helical and spur configurations.

Two new condition indicators for gears have been defined: the average distance of the FM and AM sidebands in the order representation of the synchronous average, and the sum of exceptional values in the distance of the spectral kurtosis based on the synchronous average.

The detection powers of the new condition indicators were compared with the even statistical moments (RMS and kurtosis) of the regular synchronous average, residual, and difference signals and their envelope.

Since all the records without gear faults have been used for the baseline generation, the comparison of features is related only to their capability of detection ignoring the potential of false alarms.

The detection powers of the statistical moments (RMS and kurtosis) were similar over the three types: regular, residual and difference.

In some cases, the RMS gave indications on the presence of faulty gears, but its detection power was found to be inferior to the other condition indicators.

The kurtosis did not discriminate between the healthy and faulty gears.

The average FM and AM sidebands displayed the best results in most of the cases, emerging as the most promising condition indicators.

The sum of exceptional spectral kurtosis values seems to allow a reliable detection of the presence of a faulty gear but without identification of the specific gearwheel or type of fault. In the helical configuration, only the spectral kurtosis was able to detect the presence of faults. However, one needs to be cautious about the spectral kurtosis condition indicator. Though it showed a good potential to detect faults, we believe that this new indicator requires more research regarding the potential for false alarms. The dataset that was used in this study was not large enough to address the issue sufficiently.

The new condition indicators demonstrated the best discrimination between healthy and faulty gears in the PHM'09 dataset. Usage of these condition indicators on additional datasets with faulty gears is recommended, for further solidifying of this conclusion.

**REFERENCES**

Antoni J. (2006), The Spectral Kurtosis: a Useful Tool for Characterizing Non-stationary Signals, *Mechanical Systems and Signal Processing*, 20 (2004), pp282-307.  
 Antoni J. & Randall R. B. (2002), Differential Diagnosis of Gear and Bearing Faults, *Journal of Vibration and Acoustics*, Vol. 124, pp165-171.  
 Antoni J. & Randall R. B. (2004), The Spectral Kurtosis: Application to the Vibratory Surveillance and

Diagnostics of Rotating Machines, *Mechanical Systems and Signal Processing*, 20 (2006), pp308-331.  
 Dempsey P. J. & Zakrajsek J. J. (2001), Minimizing Load Effects on NA4 Gear Vibration Diagnostic Parameter, NASA/TM—2001-210671.  
 Klein R., Rudyk E., Masad E. & Issacharoff M. (2011a), Emphasizing Bearing Tones for Prognostics, *The International Journal of Condition Monitoring*, Vol. 1, Issue 2, pp. 73-78.  
 Klein R., Rudyk E., Masad E. & Issacharoff M. (2011b), Model Based Approach for Identification of Gears and Bearings Failure Modes, *International Journal of Prognostics and Health Management*, ISSN 2153-2648, 2011 008.  
 Klein R., Rudyk E. & Masad E. (2011c), Decision and Fusion for Diagnostics of Mechanical Components, *Annual Conference of the Prognostics and Health Management Society*, 2011.  
 Lebold M., McClintic K., Campbell R., Byington C. & Maynard K. (2000), Review of vibration analysis methods for gearbox diagnostics and prognostics, *Proceedings of 54th Meeting of the Society for Machinery Failure Prevention Technology*, Virginia Beach, VA, May 1-4, 2000, pp. 623-634.  
 Mosher M., Pryor A. H. & Huff E. M. (2002), Evaluation of Standard Gear Metrics in Helicopter Flight Operation, *Proceedings of 56th Mechanical Failure Prevention Technology Conference*, Virginia Beach, VA, April 15-19, 2002.  
 Zacksenhouse M., Braun S., Feldman M. & Sidahmed M. (2000), Toward Helicopter Gearbox Diagnostics from a Small Number of Examples, *Mechanical Systems and Signal Processing*, 14(4) pp 523-543.

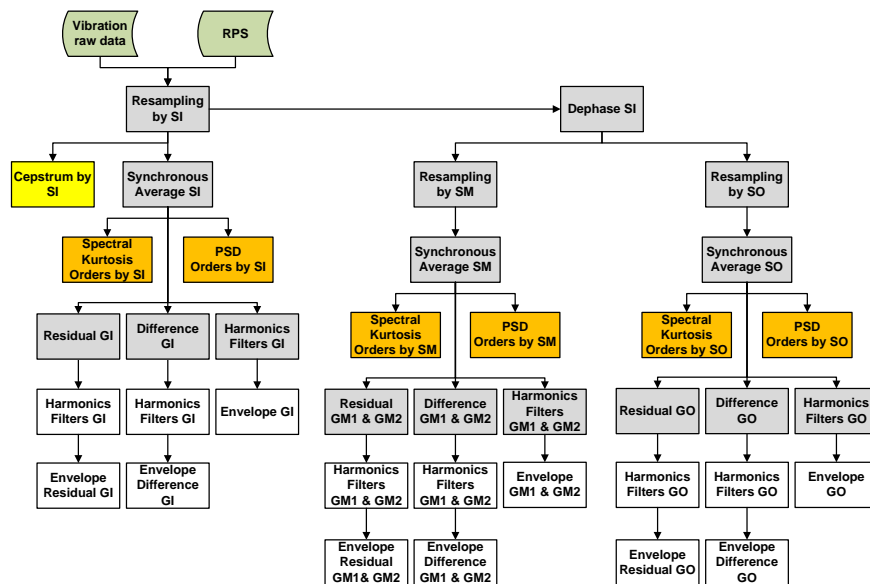


Figure 16. Flow chart of Signal Processing stage for gears of PHM'09 challenge data

# Statistical Health Grade System against Mechanical failures of Power Transformers

Chao Hu<sup>1</sup>, Pingfeng Wang<sup>2</sup>, Byeng D. Youn<sup>3,\*</sup>, Wook-Ryun Lee<sup>4</sup> and Joung Taek Yoon<sup>3</sup>

<sup>1</sup>*University of Maryland at College Park, Maryland, 20742, United States (Currently at Medtronic, Inc.)  
huchaost@umd.edu*

<sup>2</sup>*Wichita State University, Wichita, Kansas, 67260, United States  
pingfeng.wang@wichita.edu*

<sup>3</sup>*Seoul National University, Seoul, 151-742, Korea  
bdyoun@snu.ac.kr  
kaekol@snu.ac.kr*

<sup>4</sup>*Korea Electric Power Research Institute, Daejeon, 305-760, Korea  
maerong@kepri.re.kr*

## ABSTRACT

A health grade system against mechanical faults of power transformers has been little investigated compared to those for chemical and electrical faults. This paper thus presents a statistical health grade system against mechanical faults in power transformers used in nuclear power plant sites where the mechanical joints and/or parts are the ones used for constraining transformer cores. Two health metrics—root mean square (RMS) and root mean square deviation (RMSD) of spectral responses at harmonic frequencies—are first defined using vibration signals acquired via *in-site* sensors on fifty-four power transformers in several nuclear power plants in sixteen months. We then investigate a novel multivariate statistical model, namely copula, to statistically model the populated data of the health metrics. The preliminary study shows that the proposed health metrics and statistical health grade system are feasible to monitor and predict the health condition of the mechanical faults in the power transformers.

## 1. INTRODUCTION

Power transformer is one of the most critical power elements in nuclear power plants and an unexpected transformer breakdown could cause a complete plant shut-down with substantial societal expenses. It is very important to ensure high reliability and maintainability of the

transformer during its operation. Investigations of the fault causes have revealed that mechanical and electric faults are primarily responsible for unexpected breakdowns of the transformers (Lee et al., 2005). In total, 32 breakdowns of main power transformers in Korean nuclear power plants have been reported since 1978. Table 1 classifies these breakdown causes into three groups (electrical, chemical, and mechanical problems) and ways to manage them. Preventive health management for power transformers has been developed and implemented mainly for chemical and electrical faults. Although mechanical failures are responsible for about 40% of the transformer breakdowns, the non-existence of generic health metrics or a health grade system makes it difficult to perform preventive maintenance actions for mechanical faults in a timely manner and only corrective maintenance has been employed.

In the literature, substantial research has been carried out for the health monitoring and diagnosis of power transformers. An extensive review of health monitoring and diagnosis methods of power transformers was provided in (Wang et al., 2002) with a focus on all types of transformer failure causes, and in (Pradhan, 2006; Saha, 2003) with a focus on Insulation deterioration. Techniques commonly used for health monitoring of power transformers can be summarized as: (1) online partial discharge (PD) analysis (McArthur et al., 2004), (2) dissolved gas analysis (DGA) (IEEE std., 2008), (3) frequency response analysis (FRA) (Dick & Erven, 1978), (4) moisture-in-oil analysis (Garcia et al., 2005), (5) oil temperature analysis (Lee et al., 2005; Tang et al., 2004), (6) winding temperature analysis (Picanço et al., 2010), (7) load current and voltage analysis (Muhamad &

\* Corresponding author.

† Chao Hu et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

	Details (Occurrence)	Occurrence	Health Analysis
Electrical failures	<ul style="list-style-type: none"> <li>- Natural disasters (1)</li> <li>- Winding burnouts (2)</li> <li>- Operator mistakes (2)</li> <li>- Accidents in electric power transmission (1)</li> <li>- Mal-operation (4)</li> <li>- Product defects (1)</li> <li>- Manufacturing defect (1)</li> <li>- Product aging (1)</li> </ul>	13	Insulation Diagnosis Test
Chemical failures	<ul style="list-style-type: none"> <li>- Oil burnouts (1)</li> <li>- Impurities in winding (1)</li> <li>- Product defects (1)</li> <li>- Increase of combustible gas (3)</li> </ul>	6	Insulating Oil Analysis
Mechanical failures	<ul style="list-style-type: none"> <li>- Design defect (1)</li> <li>- Manufacturing defect (1)</li> <li>- Part corrosion (3)</li> <li>- Joint failure (3)</li> <li>- Crack, wear failure (5)</li> </ul>	13	N.A.

Table 1. Breakdown Classification of Main Power Transformers in Korean Nuclear Power Plants from 1978 to 2002

Ali, 2006), and (8) online power factor analysis (Gong et al., 2007). We note that the usage of the vibration signals in monitoring the transformer health has been quite limited. The transformer vibration generated by the core and windings propagates through the transformer oil to the transformer walls where vibration sensors can be placed for vibration measurements. Bartoletti et al. (2004) transformed measured acoustic and vibration signals into a frequency domain and suggested a few metrics that could represent the health status of transformers. Ji et al. (2006) acquired the fundamental frequency component of the core vibration signal as essential features to monitor and assess the transformer health condition. García et al. (2006) proposed a tank vibration model to detect the winding deformations in power transformers and conducted the experimental verification of the proposed model under different operating conditions and in the presence of winding deformation .

Once sensory data are acquired through the health monitoring, the data must be carefully analyzed for health diagnosis in order to identify and classify failures modes. Artificial intelligence (AI) techniques for pattern recognition have been prevailing for this purpose. Among a wide variety of AI techniques, ANNs have been most widely used in the research dealing with transformer health diagnosis (Huang, 2003; Hao & Cai-xin, 2007). Despite the good accuracy reported in the literature, the use of ANNs is limited by the intrinsic shortcomings including the danger of over-fitting, the need for a large quantity of training data

and the numerical instability. In addition to ANNs, the fuzzy logic (Hong-Tzer & Chiung-Cho, 1999; Su et al., 2000) and expert systems (Purkait & Chakravorti, 2002; Saha & Purkait, 2004) were also developed for transformer health diagnosis. These two approaches take advantage of human expertise to enhance the reliability and effectiveness of health diagnosis systems. Recently, the support vector machine (SVM) has been receiving growing attention with remarkable diagnosis results (Fei et al., 2009; Fei & Zhang, 2009). The SVM, which employs the structural risk minimization principle, achieves better generalization performance than ANNs employing the traditional empirical risk minimization principle, especially in cases of a small quantity of training data (Shin & Cho, 2006).

The status of research on prognostics and health management (PHM) of a power transformer can be summarized as:

- (1) Most health monitoring works for power transformers are focused on chemical and electrical failures, but very little on mechanical failures;
- (2) Power transformer oil, gas and temperatures have been widely used for health monitoring and diagnosis of power transformers. In contrast, the vibration signal has seldom been used for PHM in power transformer applications;
- (3) The PHM studies for power transformers currently stay at the level of monitoring and diagnosis only, with few works on the health prognostics and remaining useful life (RUL) prediction.

This summary suggests the need to construct a health management database, to formulate a health grade system against mechanical faults, and to investigate the health prognostics for power transformers. To this end, this study presents a copula-based statistical health grade system against mechanical faults of power transformers. The rest of this paper is organized as follows: Section 2 introduces the collection and pre-processing of the vibration data for power transformer health monitoring; Section 3 presents the developed copula based statistical health grade system for power transformer health monitoring and prognostics against mechanical faults followed by the conclusion in Section 4.

## 2. DATA ACQUISITION AND PRE-PROCESSING

Failures of mechanical joints and/or other parts of power transformers can be detected by analyzing mechanical vibration properly. This section discusses the fundamentals of transformer vibration, measurement procedures, and data pre-processing.

## 2.1. Fundamentals of Transformer Vibration

Power transformer vibrations are primarily generated by the magnetostriction and electrodynamic forces acting on the core and windings during the operation. The vibration of the core and windings propagates through the transformer oil to the transformer walls where vibration sensors can be placed for vibration measurements. The sensors cannot be placed onto the joints because of the transformer oil and magnetic and electric fields that can distort sensory signals. This subsection gives a brief review of the fundamental physics explaining vibrations in the transformer. Various vibration sources exist inside a transformer, contributing to the tank vibration. The transformer vibration mainly consists of the core vibration originating from magnetostriction and the winding vibration caused by electrodynamic forces resulting from the interaction of the current in a winding with leakage flux (García et al., 2006). Other vibration sources include the characteristic acoustic wave produced by the tap changer and periodic vibrations generated by the elements of the cooling system (i.e., oil pumps and fans).

Alternating current (AC) with a constant frequency in power transformers forms a magnetic field in the transformer core. The magnetic field changes the shape of ferromagnetic materials and produces mechanical vibration in the transformer. This phenomenon is called “magnetostriction.” As shown in Fig. 1, one cycle of the AC yields two peaks in the magnetic field. Assume that an AC source with the amplitude  $U_0$  and frequency  $f$  is applied to the drive and that the amplitude is less than or just sufficient to saturate the core. Then, the core vibration acceleration caused by magnetostriction can be expressed as (Ji et al., 2006)

$$a_c \propto U_0^2 \cos^2 4\pi ft \quad (1)$$

We can observe from the above equation that the magnitude of core vibration exhibits a linear relationship with the square of the AC amplitude. Furthermore, the fundamental frequency of the core vibration is twice that of the AC frequency, as we can also observe in Fig. 1.

Winding vibrations are caused by electrodynamic forces resulting from the interaction of the current in a winding with leakage flux (Ji et al., 2006). These forces  $F_w$  are proportional to the square of the load current  $I$ , expressed as

$$F_w \propto I^2 \quad (2)$$

Since the electromagnetic forces  $F_w$  are proportional to the vibration acceleration  $a_w$  of the windings we then conclude that (García et al., 2006)

$$a_w \propto I^2 \quad (3)$$

Thus, similar to the case of core vibration, the fundamental frequency of winding vibrations is also twice the AC

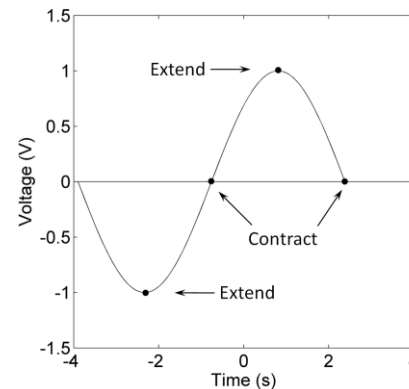


Figure 1. Magnetostriction in the transformer

frequency. The difference between these two types of vibrations is that the magnitude of core vibration relies on the voltage applied to the primary windings and is not affected by the load current, while the magnitude of winding vibrations is proportional to the square of the loading current. In addition to the variables (voltage and current) causing transformer vibration, the other factors (i.e., temperature, power factor) also have an influence on vibration (Bartoletti et al., 2004), but due to the relatively small influence, these factors are not considered in this study. In fact, since power transformers in nuclear power plants always operate at 100% full power, the variables (voltage and current) and the power factor generally exhibit very small variations over time. And since cooling systems can effectively keep transformers cores and windings at suitably low temperatures, the temperature factor also has very small fluctuation.

## 2.2. Vibration Signal Acquisition

In this study, fifty-four in-service power transformers in four nuclear power plants were employed for acquiring vibration signals. Among these transformers, three are triple-phase transformers and the others are single-phase transformers (see Table A in Appendix). These fifty-four transformers have a wide range of ages, from less than one year to about twenty-two years. This study employed B&K 4381 and PCB 357B33 accelerometers, which are charge types with charge amplifiers (RION UV-06A). Depending on the transformer size, 36 to 162 accelerometers were used to acquire the vibration signals from the transformers. The sensors were evenly positioned within 1m on the single-phase and triple-phase main transformers, as respectively shown in Figs. 2(a) and (b). Measurements were conducted along two directions (X and Y) on the surface of the transformer frame and one perpendicular direction (Z) to the surface. The accelerometers were installed on the flat surface with a magnet base in order for easy measurement.

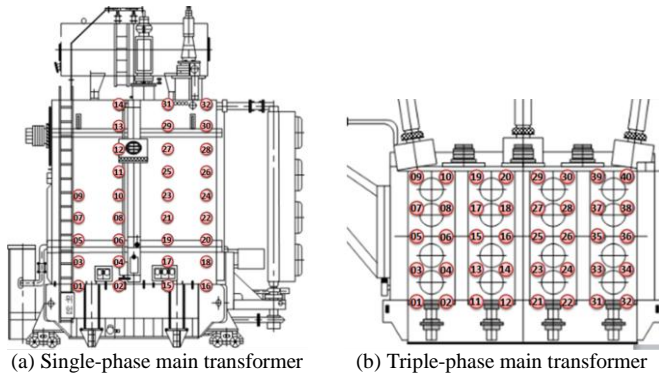


Figure 2. Sensor locations (marked by red circles with numbers) on the main transformers

All measurements were obtained in the form of time-domain signals in a full-power operation state of the power transformers. In the state, all other subsidiary units affecting vibration under normal operating conditions, such as forced cooling systems and hydraulic pumps, were turned on. The subsidiary units were supplied with 480 V AC power. In most cases, the transformers convert primary electrical values, i.e. voltages 22 kV and currents about 32 kA, to proportional secondary values, i.e. voltages 345 kV and current about 2 kA. The measurement system was powered by an independent battery power system. Vibration velocity [mm/sec] was measured at every 1.25 Hz in the frequency range of 0-2000 Hz. The rated voltage always has a frequency of 60 Hz. It is desirable to avoid taking the measurement immediately after turning the transformer on because the initial operation state of the transformer causes transient vibration signals. It is certainly important to acquire better sensory data and thus improve the performance of power transformer health diagnostic by optimizing the number of measure points and the allocation of the sensors. For the study regarding the sensor network optimization, readers are advised to the reference (Wang et al., 2010).

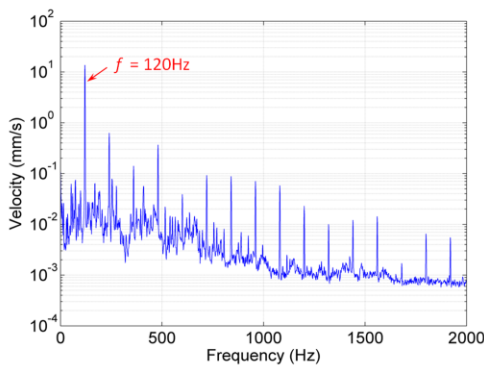


Figure 3. Frequency spectral signal of a vibration velocity (YK)

### 2.3. Data Pre-Processing

Given the fundamentals of the transformer vibration (see Section 2.1), the use of the spectral response is strongly recommended for health metrics against the mechanical faults in the transformers. The vibration signals are thus processed using a fast Fourier transform (FFT). Fig. 3 displays a spectral response of a vibration signal, which has harmonic frequencies at every 120 Hz. The amplitudes at these harmonic frequencies are significantly higher (more than ten times) than those at the other frequencies. As introduced in Section 2.1, the fundamental frequencies of both core and winding vibrations are twice the AC frequency (60Hz), which is quite consistent with our observation in Fig. 3. Since the harmonic frequencies remain constant at every 120 Hz, the amplitudes at the harmonic frequencies could imply a degree of health state against the mechanical faults in the power transformers.

The spectral response amplitudes of the vibration velocities at 120 Hz were obtained from the fifty-four transformers. We computed the mean and maximum amplitudes of the vibration velocities measured by all sensors installed on each transformer and plotted these two quantities for all fifty-four transformers in Fig. 4 and Fig. 5, respectively.

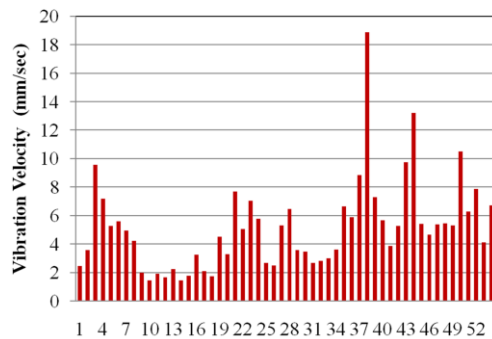


Figure 4. Mean amplitudes of spectral responses at 120 Hz for fifty-four transformers

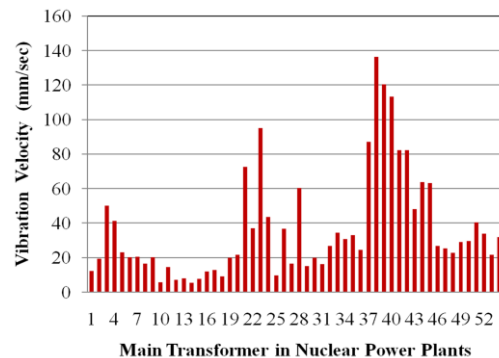


Figure 5. Maximum amplitudes of spectral responses at 120 Hz for fifty-four transformers

Two observations can be made from the two figures. Firstly, both quantities exhibit large variations among different transformers. Specifically, the mean amplitudes of the vibration velocities have a wide range of variation from 1.43 mm/sec to 18.87 mm/sec and the maximum amplitudes from 5.8 mm/sec to 136.43 mm/sec. It is believed that the aging effect of the transformers and local resonance of the transformer frame primarily causes the variation in the mean and maximum amplitudes. Secondly, the maximum velocity amplitude of each transformer is in general far greater than the mean velocity amplitude of that transformer. This observation can be attributed to the fact that, among more than forty measurement points selected for each transformer, two or three points at the upper part (closer to the top) of the transformer wall typically gave much larger velocities than the others.

### 3. HEALTH METRICS AND GRADE SYSTEM

This section presents the copula-based statistical health grade system against mechanical faults of power transformers.

#### 3.1. Health Metrics

The frequency spectral signals from multiple sensors are employed to monitor the health condition of the power transformers. Two scalar health metrics are proposed in this study: (1) root mean square (RMS) and (2) root mean square deviation (RMSD). Their definitions and physical meanings are given as follows:

**RMS** – The RMS is the quadratic measure of the vibration mean velocities measured at every 2.5 Hz in the frequency range of 2.5-2000 Hz. The RMS metric can be defined as

$$RMS_i = \left( \sum_{f=2.5\text{Hz}}^{2000\text{Hz}} \mu_f^2 \right)^{1/2}, \quad i = 1, \dots, 54 \quad (4)$$

where  $\mu_f$  is the mean of the vibration velocity measured from all sensors at a frequency  $f$ . It is generally known that measured vibration velocities in the transformers become greater as their health state degrades over years. This metric is thus a useful health metric for transformer health monitoring. However, the magnitudes of the mean velocity also vary depending on the operating condition, the transformer capacity and manufacturer. The RMS metric may fail to classify a health condition of different transformers experiencing mechanical degradation. This underscores the need of another health monitoring metric.

**RMSD** – The RMSD is the quadratic measure of the vibration deviation velocities measured at every 2.5 Hz in the frequency range of 2.5-2000 Hz. The RMSD metric can be defined as

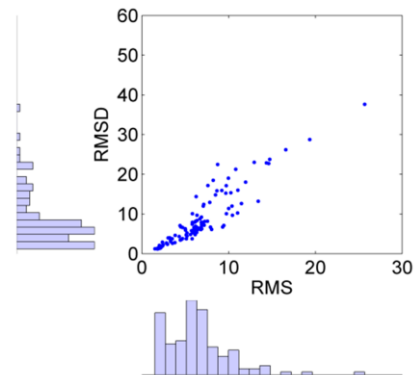


Figure 6. Scatter plot and histograms of RMS and RMSD data (from fifty-four transformers in October 2006, February 2007, and August 2007).

$$RMSD_i = \left( \sum_{f=2.5\text{Hz}}^{2000\text{Hz}} \sigma_f^2 \right)^{1/2}, \quad i = 1, \dots, 54 \quad (5)$$

where  $\sigma_f$  is the standard deviation of the vibration velocity measured from all sensors at a frequency  $f$ . The same mean velocities could indicate different health conditions if the vibration measurements come from different transformers under random operating conditions. The undesirable situation above can be avoided by using both the RMS and RMSD since the randomness in operating conditions and the difference in transformers could affect the deviation of the vibration velocity.

In cases where we have mechanical defects (for example, winding deformations or loosened clamps in the core), the magnitudes of winding or core vibration typically increase because, as aged, electrodynamic forces (for winding) generally grow; mechanical constraints (for core) loosen, and structural strength becomes weaker. Moreover, the winding or core vibration typically becomes more stochastic and, to some degree, has variation over different transformer samples. For the very reason, the magnitude (mean) and randomness (standard deviation) of tank vibration amplitude increase. The RMS and RMSD measures are capable of capturing the transformer health degradation and its variation. For the power transformers we investigated (i.e., step-up transformers used in power plants), the mean and deviation of the vibration velocity at 120 Hz was generally observed to become higher as transformers get older.

The vibration signals measured from the fifty-four transformers in June 2006, February 2007, and August 2007 were processed to acquire a populated RMS and RMSD dataset as shown in Fig. 6 (see Table B in Appendix). Since older transformers generally have larger RMS and RMSD values than newer ones, the two health metrics are highly correlated in a positive sense (see Fig. 6) with a Pearson's



linear correlation coefficient  $\rho$  being 0.9161. The transformers with a relatively good health condition are located at the lower left corner and the others at the upper right corner.

### 3.2. Copula-Based Statistical Health Grade System

As shown in Fig. 6, a strong statistical correlation exists between the proposed health metrics, RMS and RMSD. In what follows, we intend to exploit this correlation using a joint statistical model, copulas. We start with a brief introduction on copulas. Next, we present three popular types of copulas. Finally, we detail the procedures to construct an appropriate copula for dependence modeling based on available data.

#### 3.2.1. Introduction of Copulas

In statistics, a copula is defined by Roser (1999) as “a function that joins or couples multivariate joint distribution functions to their one-dimensional marginal distribution functions”, or “multivariate distribution functions whose one-dimensional margins are uniform on the interval [0,1]”. In other words, a copula formulates a joint cumulative distribution function (CDF) based on marginal CDFs and a dependence structure. In the following description, we will see that copulas allow one to decouple the univariate marginal distribution modeling from multivariate dependence modeling.

Let  $\mathbf{x} = (x_1, x_2, \dots, x_N)$  be an  $N$ -dimensional random vector with real-valued random variables,  $F$  be an  $N$ -dimensional CDF of  $\mathbf{x}$  with continuous marginal CDFs  $F_1, F_2, \dots, F_N$ . Then according to Sklar’s theorem, there exists a unique  $N$ -copula  $C$  such that

$$F(x_1, x_2, \dots, x_N) = C(F_1(x_1), F_2(x_2), \dots, F_N(x_N)) \quad (6)$$

It then becomes clear that a copula formulates a joint CDF with the support of separate marginal CDFs and a dependence structure. This decoupling between marginal distribution modeling and dependence modeling is an attractive property of copulas, since it leads to the possibility of building a wide variety of multivariate densities. In real applications, this possibility can be enabled by employing different types of marginal CDFs or dependence structures. Based on Eq. (6) and under the assumption of differentiability, we can derive the joint probability density function (PDF) of the random vector  $\mathbf{x}$ , expressed as

$$f(x_1, x_2, \dots, x_N) = c(F_1(x_1), F_2(x_2), \dots, F_N(x_N)) \cdot \prod_{i=1}^N f(x_i) \quad (7)$$

where  $c$  is the joint PDF of the copula  $C$ . The above equation suggests that a joint PDF of  $\mathbf{x}$  can be constructed as the product of its marginal PDFs and a copula PDF. The PDF formulation in Eq. (7) is useful in formulating a likelihood function and estimating the parameters of marginal PDFs and a copula, as will be discussed later.

#### 3.2.2. Copula Types

Various general types of dependence structures can be represented, corresponding to various copula families. In what follows, we will briefly introduce four popular copula types, that is, Gaussian, Clayton, Frank, and Gumbel. More detailed information on copula families can be found in (Roser, 1999).

Let  $u_i = F_i(x_i)$ ,  $i = 1, 2, \dots, N$ , an  $N$ -dimensional Gaussian copula with a linear correlation matrix  $\Sigma$  is defined as

$$C_G(u_1, u_2, \dots, u_N | \Sigma) = \Phi_N \left( \Phi^{-1}(u_1), \Phi^{-1}(u_2), \dots, \Phi^{-1}(u_N) | \Sigma \right) \quad (8)$$

where  $\Phi$  denotes the joint CDF of an  $N$ -dimensional standard normal distribution and  $\Phi^{-1}$  denotes the inverse CDF of a one-dimensional standard normal distribution. It is noted that  $\Sigma$  is a symmetric matrix with diagonal elements  $\rho_{ii}$  being ones, for  $i = 1, 2, \dots, N$ , and off-diagonal elements  $\rho_{ij}$  being the pair-wise correlations between the pseudo Gaussian random variables  $z_i = \Phi^{-1}(u_i)$  and  $z_j = \Phi^{-1}(u_j)$ , for  $i, j = 1, 2, \dots, N$  and  $i \neq j$ .

Another popular copula family is an  $N$ -dimensional Archimedean copula, defined as

$$C_A(u_1, u_2, \dots, u_N | \alpha) = \Psi_\alpha^{-1} \left( \sum_{i=1}^N \Psi_\alpha(u_i) \right) \quad (9)$$

where  $\Psi_\alpha$  denotes a generator function with a correlation parameter  $\alpha$  and satisfies the following conditions:

$$\begin{aligned} \Psi_\alpha(1) &= 0; \quad \lim_{u \rightarrow 0} \Psi_\alpha(u) = \infty; \\ \frac{d}{du} \Psi_\alpha(u) &< 0; \quad \frac{d^2}{du^2} \Psi_\alpha(u) > 0 \end{aligned} \quad (10)$$

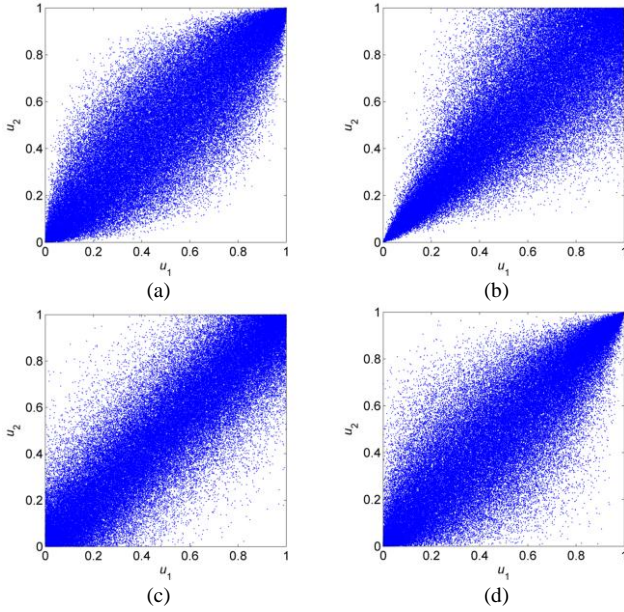


Figure 7. Scatter plots of various copulas with Kendall's tau ( $\tau$ ) coefficients being 0.70: (a) Gaussian; (b) Clayton; (c) Frank; and (d) Gumbel.

Commonly used Archimedean copulas are Clayton, Frank and Gumbel copulas which are summarized in Table 2. To exemplify the diversity of copulas, we present the scatter plots of the above four copulas with Kendall's tau ( $\tau$ ) coefficients being 0.70 in Fig. 7, where we can observe significant difference in dependence patterns modeled by different copulas.

### 3.2.3. Fitting Copula Model

In this section, we aim to determine the most appropriate copula model  $C$  with marginal CDFs  $F$  to model the dependence of a random vector  $\mathbf{x}$ . Suppose that we have  $M$  independent random samples from a multivariate distribution,  $\{\mathbf{x}_j = (x_{1j}, x_{2j}, \dots, x_{Nj}), j = 1, 2, \dots, M\}$ . Let  $\boldsymbol{\beta}$  be the vector of marginal distributional parameters and  $\boldsymbol{\alpha}$  be the vector of copula parameters. The procedure to fit a copula model is detailed as the following three steps:

*Step 1 (Parameter Estimation):* The aim of this step is to estimate the parameters  $\boldsymbol{\beta}$  and  $\boldsymbol{\alpha}$ . This can be done with the maximum likelihood method (MLE). According to Eq. (7), the log-likelihood function to be maximized is

$$f(x_1, x_2, \dots, x_N) = \sum_{j=1}^M \log c \left( F_1(x_{1j}; \boldsymbol{\beta}), F_2(x_{2j}; \boldsymbol{\beta}), \dots, F_N(x_{Nj}; \boldsymbol{\beta}); \boldsymbol{\alpha} \right) + \sum_{i=1}^N \sum_{j=1}^M \log f_i(x_{ij}; \boldsymbol{\beta}) \quad (11)$$

Family	Generator $\Psi_\alpha(u)$	Bivariate copula $C_A(u_1, u_2   \alpha)$	Parameter space
Clayton	$u^{-\alpha} - 1$	$(u_1^{-\alpha} + u_2^{-\alpha} - 1)^{-1/\alpha}$	$\alpha > 0$
Frank	$-\ln \frac{e^{-\alpha u} - 1}{e^{-\alpha} - 1}$	$-\frac{1}{\alpha} \left\{ 1 + \frac{[(e^{-\alpha u_1} - 1)][e^{-\alpha u_2} - 1]}{e^{-\alpha} - 1} \right\}^{-1/\alpha}$	$\alpha > 0$
Gumbel	$(-\ln u)^\alpha$	$\exp \left\{ - \left[ (-\ln u_1)^\alpha + (-\ln u_2)^\alpha \right]^{1/\alpha} \right\}$	$\alpha > 0$

Table 2. Summary of three Archimedean copulas

Since we generally do not have closed form solutions to globally maximize the above likelihood function, the simultaneous estimation of the marginal distributional and copula parameters is computationally expensive. To alleviate the computational burden, we employ a two-stage estimation method called the inference functions for margins (IFM) method, proposed by Joe (1997). The IFM method decomposes the estimation of the parameters  $\boldsymbol{\beta}$  and  $\boldsymbol{\alpha}$  into two steps. In the first step, it estimates the marginal distributional parameters  $\boldsymbol{\beta}$  by maximizing the second log-likelihood term in Eq. (11). With the estimated parameters  $\boldsymbol{\beta}$  and thus known marginal distributions, the second step then estimates the copular parameters  $\boldsymbol{\alpha}$  by maximizing the first log-likelihood term in Eq. (11).

*Step 2 (Goodness-of-Fit Test):* In this step, we intend to test whether a specific copula model with estimated parameters from Step 1 fits the samples with sufficient accuracy. For this purpose, we propose to employ the Kolmogorov-Smirnov (K-S) distance (Chakravartiet al., 1967), expressed as

$$D_{KS} = \int |F_e - F_n| dF_n \quad (12)$$

where  $F_e$  denotes the empirical CDF derived from the random samples, and  $F_n$  denotes the hypothesized CDF. We note that, to reduce the influence of outliers on the K-S distance and reflect the overall fitting quality, we computed an average absolute difference instead of a maximum one. When we test the fit of a specific marginal distribution, both  $F_e$  and  $F_n$  are univariate CDFs with the inputs being random variables  $x_i, i = 1, 2, \dots, N$ . In contrast, when we test the fit of a specific copula,  $F_e$  and  $F_n$  respectively become an empirical joint CDF and a hypothesized copula both of which take vectors of marginal probabilities  $\mathbf{u}$  as inputs.

*Step 3 (Goodness-of-Fit Retest):* To decide whether the distance measure in Eq. (12) provides sufficient evidence on the good fit of the copula, we retest the good-of-fit of the copula model by generating random samples of the size  $M$  under the assumption that the null hypothesis of an accurate fit is true (Kole et al., 2007). We repeatedly execute the retesting process  $K$  times to generate  $K$  sets of random samples and, correspondingly, obtain  $K$  distance measures by executing the aforementioned Steps 1 and 2. For each retest, we generate random samples with two steps: (i) generate sample pairs  $(u_{1j}, u_{2j})$  of  $[0, 1]$  uniformed distributed random variables  $u_{1j}$  and  $u_{2j}$  according to the copula model with the parameters  $\alpha$  estimated in Step 1; and (ii) transform the sample pairs  $(u_{1j}, u_{2j})$  to observation pairs  $(x_{1j}, x_{2j})$  with the inverse marginal CDFs  $F_1^{-1}$  and  $F_2^{-1}$ . Finally, we construct a probability distribution of the distance measure  $D_{KS}$  and determine the  $p$ -value, or the probability of observing a distance measure at least as extreme as the value obtained in Step 2 under the assumption of an accurate copula fit.

### 3.2.4. Building Copula-Based Statistical Health Grade System

In this section, we apply the aforementioned copula model to representing the joint distribution of the RMS and RMSD metrics by modeling the dependence between these two. Upon the construction of the joint distribution, we then define a statistical health grade system based on the joint CDF of the two health metrics.

#### a) Data Statistics and Marginal Distributions

Table 3 presents summary statistics on the populated RMS and RMSD data as well as the types and parameters of fitted marginal distributions. Compared to the RMS, the RMSD yields a larger mean value and a much larger variance. Kurtosis values are very high for both metrics, indicating that a large portion of the variance is contributed to by infrequent extreme deviations. This can also be observed from the histograms of the two metrics in Fig. 8, where we observe a considerable amount of extreme data for any of the two metrics. Results from the K-S test suggest that the RMS and RMSD data be statistically modeled with the Weibull and gamma distributions, respectively. Parameters of the fitted marginal distributions are given in Table 3 and their plots are presented in Fig. 8.

#### b) Copula Model

We used the aforementioned procedure to identify an appropriate copula model from the four candidates, that is, Gaussian, Clayton, Frank and Gumbel copulas. Table 4 summarizes the copula fitting results based on the populated RMS and RMSD data. Both the correlation estimate in

Health metric	Data statistics				
	Mean	Std <sup>a</sup>	Skewness	Kurtosis	Minimum Maximum
RMS	6.57	3.81	1.66	7.99	1.51 25.68
RMSD	8.27	6.64	1.64	6.04	1.20 37.63
Health metric	Fitted marginal distribution				
	Type	Parameters <sup>b</sup>			
RMS	Weibull	$\beta_1 = 7.42, \beta_2 = 1.84$			
RMSD	Gamma	$\beta_3 = 1.78, \beta_4 = 4.64$			

<sup>a</sup> Standard deviation

<sup>b</sup> Scale and shape parameters for Weibull and gamma distributions

Table 3. Summary of data statistics and fitted marginal distributions

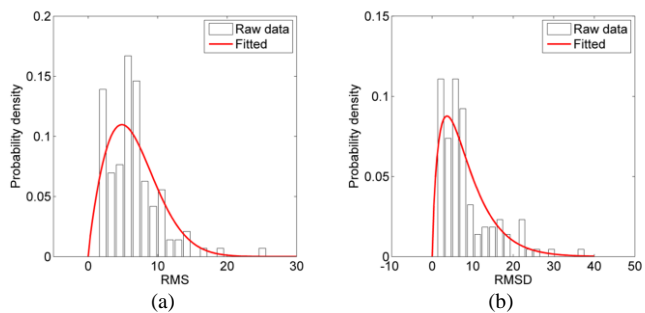


Figure 8. Histograms and fitted distributions of RMS (a) and RMSD (b).

Gaussian copula and  $\alpha$  estimates in three Archimedean copulas indicate a strong correlation between the RMS and RMSD. Regarding the retest, we generated 10,000 sets (i.e.,  $K = 10,000$ ) of random samples under the null hypothesis of an accurate copula fit and ran each of the 10,000 sets through the aforementioned Step 1 (Parameter Estimation) and Step 2&3 (Goodness-of-Fit Test & Retest) to obtain 10,000 distance measures. It can be observed that any of the four copulas cannot be rejected under the commonly used significance level 0.05. This can be partially attributed to the fact that we only have a relatively small number of data. We conjecture that, as we have more data, the  $p$ -values yielded by different copulas will become more distinctive and an appropriate copula model can be selected with more confidence. Out of the four copulas, Gaussian copula produced the smallest distance measure  $D_{KS}$  and the largest  $p$ -value, which offers us a supporting evidence of the best fit provided by Gaussian copula. The histogram of  $D_{KS}$  of Gaussian copula is plotted with the estimated  $D_{KS}$  in Fig. 9(a). To verify the accuracy of fit, we synthetically generated 1000 random samples from the fitted Gaussian copula model and plot these samples together with the raw

data in Fig. 9(b). We can observe a generally accuracy representation of the raw data, especially in the lower-left region. The synthetic samples were generated by following the same steps we used to generate the 10,000 random sets for the retest: first drawing uniform samples from the copula and then transforming these samples back to the original Weibull and gamma samples using the inverse CDFs of these distributions.

c) Health Grade System

We quantify the health condition for a specific transformer unit (i.e., a specific RMS and RMSD pair) by the proportion of the population with larger RMS and RMSD values than that unit. Let  $x_1$  and  $x_2$  denote the health metrics RMS and RMSD, respectively. Let  $C(F_1(x_1), F_2(x_2))$  denote the copula model we derived from the previous section with  $F_1(x_1)$  and  $F_2(x_2)$  being the marginal CDFs of  $x_1$  and  $x_2$ . Mathematically, the health condition  $h$  of an health metric pair  $(x_{1d}, x_{2d})$  can be defined in terms of marginal CDFs and a joint CDF or copula, expressed as

$$h(x_{1d}, x_{2d}) = \Pr(x_1 > x_{1d}, x_2 > x_{2d}) \quad (13)$$

This can be further derived as a function of the marginal CDFs of  $x_1$  and  $x_2$  and the copula, expressed as

$$\begin{aligned} h(x_{1d}, x_{2d}) &= \Pr(u_1 > F_1(x_{1d}), u_2 > F_2(x_{2d})) \\ &= \Pr(u_1 > F_1(x_{1d})) - \Pr(u_2 \leq F_2(x_{2d})) \\ &\quad + \Pr(u_1 \leq F_1(x_{1d}), u_2 \leq F_2(x_{2d})) \\ &= 1 - F_1(x_{1d}) - F_2(x_{2d}) + C(F_1(x_{1d}), F_2(x_{2d})) \end{aligned} \quad (14)$$

It is noted that, in the above equation, we define the health condition of a transformer unit as the probability of a joint event rather than a union one. The aim of this definition is to achieve a certain level of conservativeness since the mechanical failure of a power transformer causes significant monetary and societal losses and is rather undesirable.

Based on the health condition defined in Eq. (14), we further defined three health grades which, from the perspective of probability, can be mapped to three ranges in a zero-mean normal distribution, that is, below  $1.0\sigma$ , between  $1.0\sigma$  and  $2.0\sigma$  and above  $2.0\sigma$ , as shown in Table 5. Table 6 relates the three health grades to suggested maintenance actions. Experts' experience and historic information on inspection and maintenance of the power transformers over years were employed to derive the relationship. Fig. 10 visualizes the three health grades in the RMS-RMSD map, where the boundaries were identified by equating the health condition in Eq. (14) to the two critical health conditions in Table 5 and deriving the corresponding

Copula type	Parameter estimation <sup>a</sup>			Test and retest	
	Point estimate	Standard error	95% confidence interval <sup>b</sup>	$D_{KS}$ estimate	$p$ -value <sup>c</sup>
Gaussian	0.930	0.012	(0.906, 0.955)	<b>0.0300</b>	<b>0.3684</b>
Clayton	6.035	0.688	(4.685, 7.385)	0.0330	0.3216
Frank	15.095	1.384	(12.382, 17.809)	0.0324	0.2916
Gumbel	3.484	0.194	(3.103, 3.864)	0.0326	0.2642

<sup>a</sup> Linear correlation coefficient for Gaussian,  $\alpha$  in Table 2 for the rest  
<sup>b</sup> Point estimate  $\pm 1.96 \cdot$  standard error  
<sup>c</sup> With 10,000 simulations

Table 4. Copula fitting results

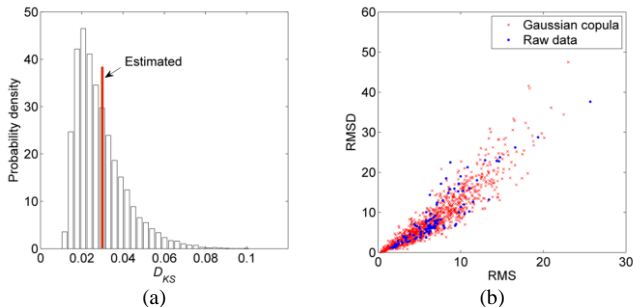


Figure 9. Histograms of DKS (a) and scatter plot (b) of Gaussian copula.

Health Grade	A	B	C
Health condition	$h > 0.16$	$0.02 < h \leq 0.16$	$h \leq 0.02$
$\sigma$ -level of standard normal distribution	$z < 1.0\sigma$	$1.0\sigma \leq z < 2.0\sigma$	$z \geq 2.0\sigma$

Table 5. Definition of three health grades

Health Grade	Health Conditions and Suggested Maintenance Actions
Grade A (Healthy)	Excellent health condition – Health condition is excellent; transformer requires least frequent inspection and maintenance.
Grade B (Warning)	Transitional health condition – Health condition has partial degradation; transformer requires more frequent inspection (e.g., in-situ monitoring) to obtain health metric data that can be related to health condition; condition-based maintenance (CBM) should be considered on the basis of remaining useful life (RUL) prediction by health prognostics.
Grade C (Faulty)	Critical health condition – Health condition is close to failure due to mechanical faults in a component level; field engineers need to identify fault type, location, and severity; transformer requires an immediate replacement of faulty mechanical components to avoid entire transformer failure if they can be identified.

Table 6. Maintenance actions on health grades

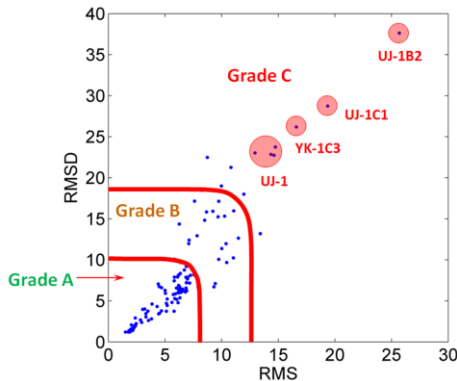


Figure 10. Statistical health grade map.

joint probability contours. The transformers that are classified into Grade A turned out to be relatively newer transformers (average life about 6 year-old), whereas those with Grades B and C were relatively older transformers (average life about 30 year-old). To verify the feasibility of the proposed health grade system, we looked at the oldest transformers (UJ1, YK1) more closely. The health grades of the transformers were identified with “Grade C” which indicates that inspection and maintenance actions must be executed immediately. It has been confirmed by the experts that the transformers’ health conditions were critical and they were recently replaced with new transformers. This indicates that the proposed grade system properly defines the health condition of the transformers against mechanical faults. Finally, we note that the boundaries ( $1.0\sigma$  and  $2.0\sigma$  standard normal lines) adopted in this paper might not be directly applicable for all practical use cases and certain customizations need to be made to satisfy a particular need. It is noted, however, that the procedure to build a statistical health grade system is general in the sense that it is directly applicable to all use cases. Moreover, field experts will make a final decision on maintenance while using this classification as a reference.

### 3.3. Feasibility Study of Health Prognostics

Prognostics is the discipline of predicting the remaining useful lives (RULs) of engineered systems over the lifetime. To make the life prognostics useful, a significant amount of health condition (RMS and RMSD) data must be acquired from a set of homogeneous transformers. Given the limited available data sets obtained in June 2006, February 2007, and August 2008, this study is intended not to develop a rigorous life prognostics model but to conduct a feasibility study for the life prognostics. This feasibility study was performed with the data sets from the power transformers in the WS nuclear power plant. The copula model transformed the two-dimensional health metrics (RMS and RMSD) into the one-dimensional health condition using Eq. (14) which

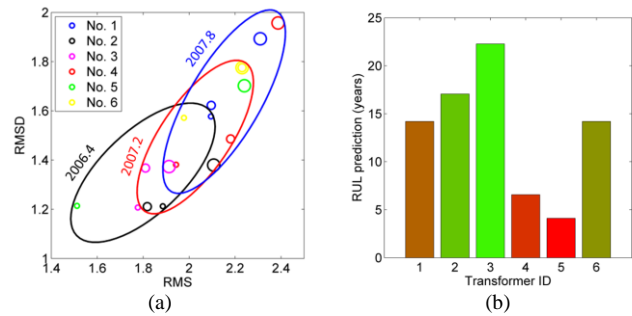


Figure 11. Health degradation history (a) and predicted RULs (b) for power transformers (WS).

was then used for observing and predicting the health degradation in the lifetime of the transformers. The RMS and RMSD data from the transformers in the WS plant were plotted and sorted by the measured time, as shown in Fig. 11(a). Note that an older transformer is plotted with a larger circle. The vibration measurements were taken from all 54 transformers in June 2006, February 2007 and August 2007, providing totally  $3 \times 54$  data samples. The health degradation of transformer core joints is demonstrated over time in Figure 11, in which six same-type (or homogeneous) transformers out of 54 are used and indeed shows a clear degradation trend. The variation in two health metrics is mainly due to different capacities of the transformers and randomness in the operation conditions. As the time passed from June 2006 to August 2007, the health condition metrics became higher. This indicates that the health condition degradation can be distinctively observed by monitoring the health condition metrics. Generally, the linear, exponential, power and logarithmic models are basic mathematical models that can be used to extrapolate the degradation measurements to the defined failure level in order to estimate the failure time. As shown in Fig. 10, the health degradation behaves exponentially, increasing slowly at the early life of the transformer but rapidly at the end of the life. Thus, we used the exponential model to capture the transition trend of the health condition and extrapolated the exponential model to a failure threshold ( $h = 0$ ) to obtain the remaining useful life. The health prognostics results for the six transformer units are graphically shown in Fig. 11(b), where the predicted RULs range from below 5 years to above 20 years.

### 4. CONCLUSIONS

This paper presented a copula-based statistical health grade system against mechanical faults of power transformers in nuclear power plants. The vibration signal signatures acquired from the power transformers were used to define two health metrics (RMS and RMSD). The populated metrics data from fifty-four power transformers were used to identify an appropriate copula model, based on which a

statistical health grade system is built with corresponding health conditions and suggested maintenance actions. The copula-based statistical health grade system can be useful for making maintenance decisions, while monitoring the health conditions of the power transformers. It is noted that uncertainties in manufacturing conditions, operation conditions and measurements further propagate to uncertainties in the two health metrics. Thus, a health grade system should not only be characterized by its diagnostic accuracy but also by its ability to perform the diagnostics in a statistical manner. In this light, the proposed statistical health grade system offer researchers and industrial practitioner a powerful tool to systematically capture the aforementioned uncertainties and build statistical power in defining health grades. To investigate the feasibility of the proposed statistical health grade system for health prognostics, we established an exponential model to capture the transition trend of the health condition and predicted the remaining useful life by extrapolation. Finally, we conclude that the copula model is capable of characterizing the statistical dependence between the two health metrics, and that the health condition defined based on this model is an attractive health measure suitable for health prognostics.

**APPENDIX**

Location	Unit	Type	Manufacture	Voltage (High/Low, kV)	Capacity (MVA, at 55°C)
KORI	1	3 phase	hyosung	362/22	750
	2	3 phase	hyosung	362/22	790
	3	1 phase	hyosung	362/22	385 * 3
	4	1 phase	hyosung	362/22	385 * 3
YK	1	1 phase	hyosung	362/22	403 * 3
	2	1 phase	hyosung	362/22	403 * 3
	3	1 phase	hyosung	345/20.9	353.3 * 3
	4	1 phase	hyosung	345/20.9	353.3 * 3
	5	1 phase	hyosung	345/20.9	353.3 * 3
	6	1 phase	hyosung	345/20.9	353.3 * 3
WS	1	3 phase	Hyundai	362/26	840
	2	1 phase	Hyundai	345/22	277 * 3
	3	1 phase	Hyundai	345/22	277 * 3
	4	1 phase	Hyundai	345/22	277 * 3
UJ	1	1 phase	hyosung	362/22	372.8 * 3
	2	1 phase	hyosung	362/22	372.8 * 4
	3	1 phase	hyosung	345/20.9	353.3 * 3
	4	1 phase	hyosung	345/20.9	353.3 * 3
	5	1 phase	hyosung	345/20.9	353.3 * 3
	6	1 phase	hyosung	345/20.9	353.3 * 3

Table A. Specifications for transformers in KORI, YK, UJ, and WS.

Location	Unit	Phase	04/2006		02/2007		08/2007	
			RMS	RMSD	RMS	RMSD	RMS	RMSD
YK	1	A						
		B			3.4935	3.7586	4.4083	4.7433
		C	10.8184	21.2638	9.6774	15.2434	11.9149	18.7117
	2	A	6.1644	8.0906	5.8147	7.4488	7.2082	9.1807

3	A	B	8.2268	18.4866	6.2618	14.3723	7.7437	17.7022
		C	7.0891	11.9838	5.8223	7.8991	7.2067	9.7351
			3.5302	2.6278	2.9425	2.3837	3.7121	3.0416
4	A	B	8.7348	22.4859	4.4078	7.0695	5.4724	8.7757
		C	6.2146	8.1801	5.7957	3.7542	7.4235	4.8832
			7.0048	7.9036	6.3626	9.68	8.0392	12.0284
5	A	B	5.8535	5.2815	4.4444	3.4313	5.5571	4.295
		C	4.5239	3.5049	3.3184	3.4511	4.1653	4.2948
			8.0345	6.1175	3.3179	3.0496	4.2151	3.8115
6	A	B	6.4082	6.0413	3.8493	4.6327	4.8549	5.7959
		C	10.2612	15.3348	4.6928	6.6165	5.7945	8.1362
			6.2767	6.7398	5.002	6.2901	6.1577	7.7343
UJ	1	A						
		B			25.683	37.6344		
		C	19.3516	28.7425	14.738	23.756	8.4941	14.8265
2	A	B	9.9755	19.0113	12.9438	23.0255	14.619	22.7439
		C	11.0612	15.9743	8.6555	15.8371	11.9566	18.0058
			14.3538	22.8637	9.2198	15.9236	7.2417	8.1277
3	A	B	6.1895	5.9806	10.0143	11.3763	6.5462	6.4366
		C	11.4765	12.6484	13.4102	13.199	9.7375	10.0747
			9.7347	17.1477	6.8758	9.2086	6.8016	6.0879
4	A	B	7.1267	12.411	6.1393	6.0104	7.8248	12.9405
		C	4.6828	3.7881	6.0354	5.198	6.9082	6.2445
			3.0354	3.976	6.1473	4.7841	6.8478	7.8904
5	A	B	6.8923	7.2407	5.1924	5.2341	6.7011	8.7737
		C	11.0317	10.2376	9.4405	7.1195	10.4389	9.6835
			5.7305	5.792	6.0117	6.5637	5.8994	5.8661
6	A	B	6.195	6.8874	7.1765	7.1809	7.5634	8.1893
		C	6.1051	7.9166	5.114	4.7147	5.8188	10.0799
			6.4567	6.1449	6.3587	6.4674	6.8919	7.2125
WS	1	A						
		B					2.2402	1.702
		C	1.9431	1.3812	2.1804	1.4858	2.3866	1.9573
3	A	B	2.0948	1.5776	2.0962	1.6223	2.3091	1.8932
		C	1.8853	1.2124	1.8178	1.2107	2.1066	1.3798
			1.7767	1.207	1.8101	1.3677	1.9125	1.3735
4	A	B	2.3797	2.9213	2.3555	2.0227	2.3555	2.0227
		C	2.0295	2.0761	2.4585	2.2441	2.4585	2.2441
			1.9772	1.5718	2.2301	1.7756	2.2301	1.7756

Table B. RMS and RMSD for all transformers in YK, UJ, and WS.

**ACKNOWLEDGEMENT**

This work was partially supported by a grant from the Energy Technology Development Program of Korea Institute of Energy Technology Evaluation and Planning (KETEP), funded by the Korean government’s Ministry of Knowledge Economy and by the SNU-IAMD.

**REFERENCES**

Bartoletti, C., Desiderio, M., Carlo, D., Fazio, G., Muzi, F., Sacerdoti, G., & Salvatori, F. (2004). Vibro-acoustic Techniques to Diagnose Power Transformers. *IEEE Transactions on Power Delivery*, v19, n1, p221–229.

- Chakravarti, M., Laha, R.G., & Roy, J., (1967). *Handbook of methods of applied statistics, volume I*. John Wiley and Sons, p392–394.
- Dick, E.P., & Erven, C.C. (1978). Transformer Diagnostic Testing by Frequency Response Analysis. *IEEE Transactions on Power Apparatus and Systems*, vPAS-97, n6, p2144–2153.
- Fei, S.-W., Liu, C.-L., & Miao, Y.-B. (2009). Support Vector Machine with Genetic Algorithm for Forecasting of Key-gas Ratios in Oil-immersed Transformer. *Expert Systems with Applications*, v36, n3, p6326–6331.
- Fei, S., & Zhang, X. (2009). Fault Diagnosis of Power Transformer based on Support Vector Machine with Genetic Algorithm. *Expert Systems with Applications*, v36, n8, p11352–11357.
- García, B., Burgos, J.C., Alonso, A.M., & Sanz, J. (2005). A Moisture-in-Oil Model for Power Transformer Monitoring—Part II: Experimental Verification. *IEEE Transactions on Power Delivery*, v20, n2, p1423–1429.
- García, B., Member, Burgos, J.C., & Alonso Á.M. (2006). Transformer Tank Vibration Modeling as a Method of Detecting Winding Deformations—Part I: Theoretical Foundation. *IEEE Transactions on Power Delivery*, v21, n1, p157–163.
- García B., Burgos J.C. & Alonso, Á.M. (2006). Transformer Tank Vibration Modeling as a Method of Detecting Winding Deformations—Part II: Experimental Verification. *IEEE Transactions on Power Delivery*, v21, n1, p164–169.
- Gong, L., Liu, C.-H., & Zha, X.F. (2007). Model-Based Real-Time Dynamic Power Factor Measurement in AC Resistance Spot Welding with an Embedded ANN. *IEEE Transactions on Industrial Electronics*, v54, n3, p1442–1448.
- Hao, X., & Cai-xin, S. (2007). Artificial Immune Network Classification Algorithm for Fault Diagnosis of Power Transformer. *IEEE Transactions on Power Delivery*, v22, n2, p930–935.
- Hong-Tzer, Y., & Chiung-Chou, L. (1999). Adaptive Fuzzy Diagnosis System for Dissolved Gas Analysis of Power Transformers. *IEEE Transactions on Power Delivery*, v14, n4, p1342–1350.
- Huang, Y. C. (2003). Evolving Neural Nets for Fault Diagnosis of Power Transformers. *IEEE Transactions on Power Delivery*, v18, n3, p843–848.
- IEEE std. C57.104 (2008). *IEEE guide for the interpretation of gases generated in oil-immersed transformers*.
- Ji, S., Luo, Y., & Li, Y. (2006). Research on Extraction Technique of Transformer Core Fundamental Frequency Vibration Based on OLCM. *IEEE Transactions on Power Delivery*, v21, n4, p1981–1988.
- Joe, H. (1997). *Multivariate models and dependence concepts: Monographs on Statistics and Applied Probability*, vol. 73. Chapman & Hall, London, UK.
- Kole, E., Koedijk, K., & Verbeek, M. (2007). Selecting Copulas for Risk Management. *Journal of Banking and Finance*, v31, n8, p2405–2423.
- Lee, W.R., Jung, S.W., Yang, K.H., & Lee, J.S. (2005). A study on the determination of subjective vibration velocity ratings of main transformers under operation in nuclear power plants. In *Proceedings of the 12th International Congress on Sound and Vibration* (Paper No. 1017), July 11–14, Lisbon, Portugal.
- McArthur, S.D.J., Strachan, S.M., & Jahn, G. (2004). The Design of a Multi-Agent Transformer Condition Monitoring System. *IEEE Transactions on Power Delivery*, v19, n4, p1845–1852.
- Muhamad, N.A., & Ali, S.A.M. (2006). Simulation Panel for Condition Monitoring of Oil and Dry Transformer Using LabVIEW with Fuzzy Logic Controller. *Journal of Engineering, Computing & Technology*, v14, p187–193.
- Picanço, A.F., Martinez, M.L.B., & Paulo, C.R. (2010). Bragg System for Temperature Monitoring in Distribution Transformers. *Electric Power System Research*, v80, n1, p77–83.
- Pradhan, M. K. (2006). Assessment of the Status of Insulation during Thermal Stress Accelerated Experiments on Transformer Prototypes. *IEEE Transactions on Dielectrics and Electrical Insulation*, v13, n1, p227–237.
- Purkait, P., & Chakravorti, S. (2002). Time and Frequency Domain Analyzes based Expert System for Impulse Fault Diagnosis in Transformers. *IEEE Transactions on Dielectrics and Electrical Insulation*, v9, n3, p433–445.
- Roser, B.N. (1999). *An Introduction to Copulas*. New York: Springer.
- Saha, T. K., & Purkait, P. (2004). Investigation of an Expert System for the Condition Assessment of Transformer Insulation based on Dielectric Response Measurements. *IEEE Transactions on Power Delivery*, vol. 19, no. 3, p1127–1134.
- Saha, T.K. (2003). Review of Modern Diagnostic Techniques for Assessing Insulation Condition in Aged Transformers. *IEEE Transactions on Dielectrics and Electrical Insulation*, v10, n5, p903–917.
- Shin, H.J., & Cho, S. (2006). Response Modeling with Support Vector Machines. *Expert Systems with Applications*, v30, n4, p746–760.
- Su, Q., Mi, C., Lai, L.L., & Austin, P. (2000) A Fuzzy Dissolved Gas Analysis Method for the Diagnosis of Multiple Incipient Faults in a Transformer. *IEEE Transactions on Power Systems*, v15, n2, p593–598.

- Tang, W.H., Wu, Q.H., & Richardson Z.J. (2004). A Simplified Transformer Thermal Model Based on Thermal-Electric Analogy. *IEEE Transactions on Power Delivery*, v19, n3, p1112–1119.
- Wang, M., Vandermaar, A. J., & Srivastava, K. D. (2002) Review of condition assessment of power transformers in service. *IEEE Electrical Insulation Magazine*, v18, n6, p12–25.
- Wang, P., Youn, B.D., & Hu, C. (2010). A generic sensor network design framework based on a detectability measure. *ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, August 15- 18, Montreal, Quebec, Canada.

#### BIOGRAPHIES



**Chao Hu:** Dr. Chao Hu is currently working as a senior reliability engineer at Medtronic, Inc. in Minneapolis, MN. He received his B.E. degree in Engineering Physics from Tsinghua University in Beijing, China in 2003, and the Ph.D. degree in mechanical engineering at the University of Maryland, College Park in USA. His research interests are system reliability analysis, prognostics and health management (PHM), and battery power and health management of Li-ion battery system. Dr. Hu's research work has led to more than 30 journal and conference publications in the above areas.



**Pingfeng Wang:** Dr. Pingfeng Wang received his B.E. degree in Mechanical Engineering from The University of Science and Technology in Beijing, China in 2001, the M.S. degree in Applied Mathematics in Tsinghua University in Beijing, China in 2006, and the Ph.D. degree in mechanical engineering at the University of Maryland, College Park in USA. He is currently an Assistant Professor in the Department of Industrial and Manufacturing Engineering at Wichita State University. His research interests are system reliability analysis, risk-based design, and prognostics and health management (PHM).



**Byeng D. Youn:** Dr. Byeng D. Youn received his Ph.D. degree from the department of Mechanical Engineering at the University of Iowa, Iowa City, IA, in 2001. He was a research associate at the University of Iowa (until 2004), an assistant professor in Michigan Technical University (until 2007), and an assistant professor in the University of Maryland College Park (until 2010). Currently, he is an assistant professor at the School of Mechanical and

Aerospace Engineering at Seoul National University, Republic of Korea. His research is dedicated to well-balanced experimental and simulation studies of system analysis and design, and he is currently exploring three research avenues: system risk-based design, prognostics and health management (PHM), and energy harvester design. Dr. Youn's research and educational dedication has led to: six notable awards, including the ISSMO/Springer Prize for the Best Young Scientist in 2005 from the International Society of Structural and Multidisciplinary Optimization (ISSMO), and more than 100 publications in the area of system-risk-based design and PHM and energy harvester design.



**Wook-Ryun Lee:** Mr. Wook-Ryun Lee received his Master's degree in Mechanical Engineering from Chungnam National University in 2004 and his Bachelor's degree from Yonsei University in 1997. He is currently a senior researcher in the Green Energy Laboratory of the Research Institute in the Korea Electric Power Corporation, Daejeon, Korea. His research interests are control of noise and vibration generated from power plants, and kinetic energy storage system such as flywheel energy storage, energy-harvesting etc.



**Joung Taek Yoon:** Mr. Joung Taek Yoon received his B.S. degree in mechanical engineering from Seoul National University in 2011 and is currently pursuing a combined master's and doctorate program in mechanical engineering from Seoul National University. His research interests are prognostics and health management (PHM) and resilient system design.



# A Comparison Study of Methods for Parameter Estimation in the Physics-based Prognostics

Dawn An<sup>1</sup>, Joo-Ho Choi<sup>2</sup>, and Nam Ho Kim<sup>3</sup>

<sup>1,2</sup>*Korea Aerospace University, Goyang-City, Gyeonggi-do, 412-791, Korea*

*skal34@nate.com*

*jhchoi@kau.ac.kr*

<sup>3</sup>*University of Florida, Gainesville, FL, 32611, USA*

*nkim@ufl.edu*

## ABSTRACT

Prediction of remaining useful life of a system is important for safety and maintenance scheduling. In the physics-based prognostics, the accuracy of predicted remaining useful life is directly related to that of estimated model parameters. It, however, is not a simple task to estimate the model parameters because most real systems have multivariate model parameters, which are often correlated each other. This paper mainly discusses the difference in estimating model parameters among different prognostics methods: the particle filter method, the overall Bayesian method, and the incremental Bayesian method. These methods are based on the same theoretical foundation, Bayesian inference, but they are different from each other in the sampling scheme and/or uncertainty analysis process. A simple analytical example and the Paris model for crack growth are used to demonstrate the difference among the three methods in terms of prognostics metrics. The numerical results show that particle filter and overall Bayesian methods outperform the incremental Bayesian method. Even though the particle filter shows slightly better results in terms of prognostics metrics, the overall Bayesian method is efficient when batch data exist.

## 1. INTRODUCTION

Structural health monitoring (SHM) facilitates condition-based maintenance that provides a safe and cost-effective strategy by predicting the level of degradation or damage without intrusive and time-consuming inspections (Giurgiutiu, 2008). Since SHM systems can assess damage frequently, they can also be used to predict the future behavior of damage, which is important for safety and

maintenance schedule management. SHM systems can have a significant impact on increasing safety by allowing evaluation of the system's health status and prediction of the remaining useful life (RUL), which is called prognostics.

In general, prognostics methods can be categorized into data-driven (Schwabacher, 2005), physics-based (Luo et al., 2008), and hybrid (Yan & Lee, 2007) approaches. The data-driven method that does not use any particular physical model is powerful in predicting near-future behaviors, while the physics-based method has advantages in predicting long-term behaviors of the system by identifying model parameters. Since fatigue failures slowly progresses, the physics-based method fits better than the data-driven method. In the physics-based method, model parameter estimation has a great effect on evaluating the system's health status and predicting the RUL.

The objective of this paper is to compare different methods in estimating model parameters and predicting RUL: the particle filter (PF), the overall Bayesian method (OBM), and the recursive Bayesian method (RBM). These methods are based on the Bayesian inference in which the uncertainty in model parameters can be reduced using SHM data, which makes the estimated RUL more reliable. Each method is introduced in the following paragraphs.

PF is also known as a sequential Monte Carlo method (Orchard & Vachtsevanos, 2007; DeCastro et al., 2009), for sequentially updating a time-dependent system model based on Bayesian inference, where the posterior distribution is expressed as a number of particles and their weights. PF has been widely employed to estimate system states and model parameters (Orchard & Vachtsevanos, 2007; DeCastro et al., 2009; Li et al., 2003; Zio & Piloni, 2011; Gašperin, 2011). Although many efforts have been directed to improve PF (Wang et al., 2009; Daigle & Goebel, 2011; Storvik, 2002; Campillo & Rossi, 2009) only sequential important

Dawn An et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

resampling (SIR) is considered in this paper as it is the most commonly used algorithm.

OBM is to estimate the unknown parameters using a batch of measurement data. Samples of the updated posterior distribution of the model parameters based on Bayesian inference are drawn from a viable sampling method, such as the Markov Chain Monte Carlo (MCMC) simulation (An et al., 2011; An et al., 2011; Payne, 2005; Park et al., 2010; An et al., 2012; Andrieu et al., 2003). RBM is similar to OBM, but instead of using the final updated distribution, the update is performed incrementally with each measurement data.

It, however, should be noted that the three methods are based on the same theoretical background, the Bayesian estimation technique. Despite the same theoretical background, the results from these methods are different from each other. The reason is that the PF and RBM predict the time evolution of the growing damage state, while the OBM, strictly speaking, is based on the given entire set of data, not the time evolution. Another important reason is from the difference in the sampling scheme: resampling for PF and MCMC for OBM and RBM.

The purpose of this paper is to discuss the difference between the three methods, which has not been founded yet. The methods are compared by using a simple analytical example as well as the Paris model (Paris & Erdogan, 1963) that describes crack growth in a plate under mode I loading, and are evaluated by using established prognostics metrics (Saxena et al., 2009). The paper is organized as follows. In Section 2, PF, OBM, and RBM are introduced with simple mathematical example. In Section 3, the differences between the three methods, i.e., PF, OBM, and RBM, are compared by applying crack growth problem. In Section 4, the differences are discussed, and conclusions are presented in Section 5.

## 2. THREE METHODS FOR BAYESIAN-BASED PROGNOSTICS

Bayesian inference is a statistical method in which observations are used to estimate unknown model parameters or system states. The unknown parameters are represented as a probability density function (PDF), which is updated with observed data. Bayesian inference is based on the following Bayes' theorem (Bayes, 1763):

$$p(\Theta | \mathbf{z}) \propto L(\mathbf{z} | \Theta) p(\Theta) \quad (1)$$

where  $\Theta$  is a vector of model parameters or system states,  $\mathbf{z}$  is a vector of observed data,  $L(\mathbf{z} | \Theta)$  is the likelihood or the PDF value of  $\mathbf{z}$  conditional on the given  $\Theta$ ,  $p(\Theta)$  is the prior PDF of  $\Theta$ , and  $p(\Theta | \mathbf{z})$  is the posterior PDF of  $\Theta$  conditional on  $\mathbf{z}$ . In the following subsections, three Bayesian-based methods, the particle filter (PF), the overall Bayesian method (OBM), and the incremental Bayesian method (RBM), are explained in a general way.

### 2.1. Particle Filter (PF)

The particle filter (PF, a.k.a. sequential Monte Carlo method (Orchard & Vachtsevanos, 2007; DeCastro et al., 2009)) is a method of estimating and sequentially updating a time-dependent system model based on Bayesian inference, where the posterior distribution is expressed as a number of particles and their weights. In the case of a linear system, the exact posterior distribution with a Gaussian noise can be obtained analytically using the Kalman filter (Kalman, 1960) though the convergence of KF largely depends on the initial condition of the parameter and variance of the parameter. On the other hand, the posterior distribution of a non-linear system with non-Gaussian noise cannot be obtained exactly. In such a case, there are several suboptimal filters such as the extended Kalman filter that requires linearization (Ristic, et al., 2004), the multiple model filter that performs the state estimation algorithm for multiple degradation models (Rago et al., 1998), the unscented Kalman filter (Julier & Uhlmann, 1997), and PF (Orchard & Vachtsevanos, 2007; DeCastro et al., 2009; Li et al., 2003; Zio & Peloni, 2011; Gašperin, 2011) that are based on the sampling theory. This paper focuses on the original PF because it contains fundamental characteristics of other filtering methods.

It would be better to explain the importance sampling method (Glynn & Iglehart, 1989) first to understand PF. In sampling-based methods, a number of samples can be used to approximate the posterior distribution. In order to approximate the distribution better with a limited number of samples, the importance sampling method assigns a weight to each sample (or particle) in proportion to an arbitrarily chosen importance distribution; therefore, the quality of estimation depends on the selected importance distribution. The weight is expressed as

$$w(\Theta^i) = \frac{\pi(\Theta^i)}{q(\Theta^i)} = \frac{L(\mathbf{z} | \Theta^i) p(\Theta^i)}{q(\Theta^i)} \quad (2)$$

where  $\pi(\Theta^i)$  and  $q(\Theta^i)$  are  $i$ th particle's PDF value of the posterior distribution and an arbitrarily chosen importance distribution, respectively. From the viewpoint of Bayes' theorem, it is possible to use the prior distribution as an importance distribution because it is already available and close to the posterior distribution. Then, the Eq. (2) is reduced to the likelihood function by substituting the prior  $p(\Theta^i)$  for  $q(\Theta^i)$ ; this is called the Condensation (CONDitional DENSITY propaGATION) algorithm, which is employed in this paper.

PF can be considered as a sequential importance sampling, which continuously performs the importance sampling whenever a new observed data is available. In addition, a resampling process, called sequential importance resampling (SIR), is required to solve the degeneracy

phenomenon that decreases the accuracy in the posterior distribution. In SIR, those particles with a very small weight are eliminated, while those particles with a high weight are duplicated.

Conventionally, PF has been used to estimate the system state using measurement data given model parameters, which are obtained from laboratory tests (Orchard & Vachtsevanos, 2007; Zio & Peloni, 2011). However, the parameters from laboratory tests can be different from those in service due to different conditions. In such a case, PF can be used to estimate both the system state and model parameters; detailed procedure is as follows (Li et al., 2003; Zio & Peloni, 2011):

\* Given information:

- State transition function  $f$  and measurement function  $h$ :

$$x_k = f(x_{k-1}, \theta_k, v_k) \quad (3)$$

$$z_k = h(x_k, \omega_k) \quad (4)$$

where  $k$  is the time step index,  $x_k$  is the system state,  $\theta_k$  is a vector of model parameters,  $z_k$  is measurement data. The measurement function is needed when the system state cannot be measured directly.  $v_k$  and  $\omega_k$  are, respectively, process and measurement noise; in the case of Gaussian distribution, the noises are represented as  $v_k \sim N(0, \sigma_p)$  and  $\omega_k \sim N(0, \sigma_m)$ .

1. Assume initial distribution of  $x_0$  and  $\theta_0$ . Set  $k = 1$ .
2. **Prediction**- First, generate  $N$  particles of  $\theta_k$  from  $p(\theta_k | \theta_{k-1})$ , which means a predicted distribution of  $\theta_k$  conditional on the previous distribution,  $p(\theta_{k-1})$ . It, however, should be noted that model parameters inherently do not depend on time evolution. Therefore,  $N$  particles of  $\theta_k$  become the same as the previous particles. In other words, there is no transition between  $\theta_{k-1}$  and  $\theta_k$ , only update process is required. Next, sample  $N$  particles of  $x_k$  based on  $p(x_k | x_{k-1}, \theta_k)$ , that is,  $N$  particles of each variables; process noise generated from  $N(0, \sigma_p)$  as well as  $x_{k-1}$  and  $\theta_k$  are used in Eq. (3) to obtain  $N$  particles of  $x_k$ .
3. **Update**- Calculate weight for each particle in Step 2. In the case that measurement noise is normally distributed with zero mean and standard deviation,  $\sigma_m$ , the likelihood function and weight become as follows:

$$L(z_k | x_k^i, \theta_k^i) = \frac{1}{\sqrt{2\pi}\sigma_m} \exp\left[-\frac{1}{2} \frac{(z_k - h(x_k^i))^2}{\sigma_m^2}\right], \quad (5)$$

$$w_k^i = \frac{L(z_k | x_k^i, \theta_k^i)}{\sum_{j=1}^N L(z_k | x_k^j, \theta_k^j)} \quad (6)$$

4. **Resample**- Resample the particles proportional to their weights. Several methods can be used for this purpose, and one thing among them is to repeat  $N$  times with the following way; (1) construct CDF from the likelihood function in Eq. (5). In other words, CDF is based on the weights in Eq. (6) which corresponds to PDF of  $z_k$  conditional on  $x_k^i, \theta_k^i$ . (2) find particles of  $x_k, \theta_k$  which make CDF of  $z_k$  be the same value as (or the closest to) randomly chosen value from  $U(0,1)$  which is the uniform distribution in the interval of  $[0,1]$ . By repeating  $N$  times this process,  $N$  samples of  $x_k, \theta_k$  are obtained, which represents an approximation of the posterior distribution  $p(x_k, \theta_k | z_{0:k})$  obtained based on the weighted samples  $(x_k^i, \theta_k^i | w_k^i)$ .
5. **Prognosis**- The system state  $x_{k+l}$  where  $t_{k+l}$  is the future time can be predicted based on the transition function and estimated model parameters until  $t_k$ . There are no new data and update process after  $t_k$ . Therefore, the weigh after  $t_k$  are kept unchanged with  $w_k^i = 1/N$  since every particle is assigned to the same weight after resampling.

The process of PF with one parameter is illustrated in Figure 1. The length of vertical bars represents magnitude of the weight, and the particles in the prediction step are assigned to the weight at the update step and duplicated as much as the magnitude of the weight at the resampling step.

As a simple example, the following linear system is considered.

$$x_k = x_{k-1} + \theta_k \Delta t + v_k \quad (7)$$

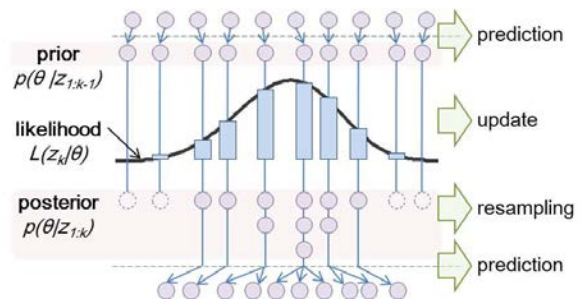


Figure 1. Illustration of the PF process with resampling

where  $\theta_k$  is a model parameter to be estimated, process noise  $v_k$  is normally distributed with zero mean and standard deviation  $\sigma_p=1$ . In this example, it is assumed that both the initial distributions of the system state and model parameter are uniform in the interval of  $[-25,50]$ . Let the measurement function  $z_k = x_k + \omega_k$ , where measurement noise  $\omega_k$  is normally distributed with zero mean and standard deviation  $\sigma_m=10$ . Then, the likelihood function to calculate the weight becomes as follows:

$$L(z_k | x_k^i, \theta_k^i) = \frac{1}{\sqrt{2\pi}\sigma_m} \exp\left[-\frac{1}{2} \frac{(z_k - x_k^i)^2}{\sigma_m^2}\right] \quad (8)$$

The results of estimated model parameter and system state with  $\Delta t=1$  are shown in the Figure 2.  $N=5,000$  particles (which is applied for all problems in this paper) are used with the data observed until  $t=20$  (or  $k=20$ ). The estimated parameter converges to the true value ( $\theta_{\text{true}}=2$ ) with small confidence intervals. In addition, the prediction result of the system state is also fairly accurate.

## 2.2. Overall Bayesian Method (OBM)

The overall Bayesian method (OBM) is to estimate the unknown parameters using the measurement data in which the final posterior distribution is expressed as an equation. The posterior distribution is obtained by multiplying all the likelihood functions given by  $n$  number of data and the prior as

$$p(\theta | \mathbf{z}) \propto L(z_1 | \theta) \times L(z_2 | \theta) \times \dots \times L(z_n | \theta) \times p(\theta) \quad (9)$$

Using Eq. (9), a sampling method, such as the Markov Chain Monte Carlo (MCMC) method (Andrieu et al., 2003), can be used to construct the posterior distribution. The Metropolis-Hastings (M-H) algorithm is a typical method of MCMC, and the procedure is summarized in the following

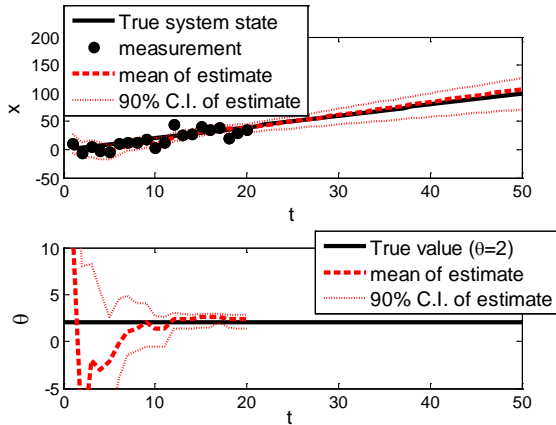


Figure 2. The results from PF in the simple example

pseudo-code:

1. Initialize  $\theta^0$
2. For  $i=1$  to  $N$ 
  - Sample  $u \sim U(0,1)$
  - Sample  $\theta^* \sim q(\theta^* | \theta^{i-1})$
  - if  $u < A(\theta^{i-1}, \theta^*) = \min\left\{1, \frac{p(\theta^* | \mathbf{z})q(\theta^{i-1} | \theta^*)}{p(\theta^{i-1} | \mathbf{z})q(\theta^* | \theta^{i-1})}\right\}$  (10)

$$\theta^i = \theta^*$$

else

$$\theta^i = \theta^{i-1}$$

where  $\theta^0$  is a vector of initial values of the unknown model parameters to estimate,  $N$  is the number of samples,  $p(\theta | \mathbf{z})$  is the posterior distribution in Eq. (9), and  $q(\theta^* | \theta^{i-1})$  is an arbitrary chosen proposal distribution which is used when a new sample  $\theta^*$  is to be drawn conditional on the current point  $\theta^{i-1}$ .  $q(\theta^{i-1} | \theta^*)$  is chosen to be a uniform distribution centered at  $[\theta^*]$  with the interval of  $\pm d$ , where  $d$  is a vector for setting the sampling interval and is selected arbitrarily based on the experience. The same interval is used for  $q(\theta^* | \theta^{i-1})$ . If the sample  $[\theta^*]$  is not accepted as an  $i$ th sample, the  $i-1$ th sample becomes the  $i$ th sample; that is, the particular sample is doubly counted. This process is illustrated in Figure 3. Compared to PF, there is no need for resampling, but subsequent samples are affected by previous samples.

In common with PF, the linear system in Eq. (7) is used for OBM, which can be rewritten in the following form after ignoring process noise,  $v_k$ :

$$x_k = \theta \times t_k \quad (11)$$

In this case, the posterior distribution is as follows:

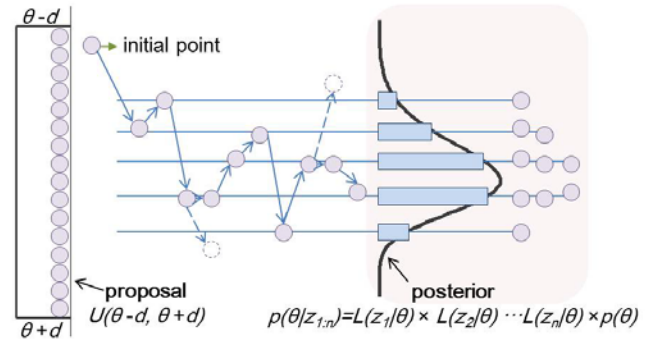


Figure 3. Illustration of the OBM process

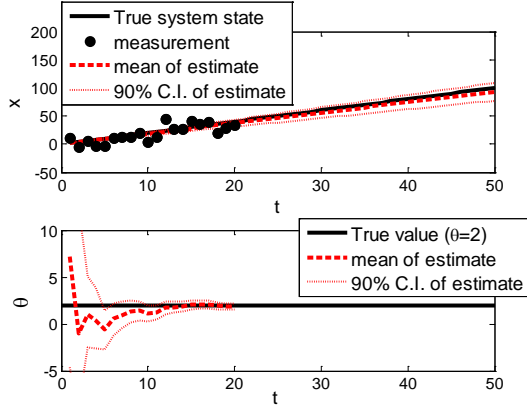


Figure 4. The results from OBM in the simple example

$$p(\theta | \mathbf{z}) \propto \frac{1}{\sqrt{2\pi}\sigma_m} \exp\left[-\frac{1}{2} \sum_{k=1}^n \frac{(z_k - x_k)^2}{\sigma_m^2}\right] \times U(-25, 50) \quad (12)$$

where  $n=20$  data are used. The system state is calculated from Eq. (11) after model parameter is estimated. The results are shown in the Figure 4. They are similar to those from PF, but the intervals are narrower than PF because there is no process noise in OBM.

### 2.3. Incremental Bayesian Method (RBM)

RBM sequentially updates the distribution instead of using the final posterior distribution based on a batch of data. In this case, Eq. (10) can be changed to the following procedure:

1. For  $k = 1$  to  $n$
2. Initialize  $\theta_k^0 \sim p(\theta_k | \theta_{k-1})$
3. For  $i = 1$  to  $N$ 
  - Sample  $u \sim U(0,1)$
  - Sample  $\theta_k^* \sim p(\theta_k | \theta_{k-1})$  (13)
  - if  $u < A(\theta_k^{i-1}, \theta_k^*) = \min\left\{1, \frac{L(z_k | \theta_k^*)}{L(z_k | \theta_k^{i-1})}\right\}$ 

$$\theta_k^i = \theta_k^*$$
  - else
 
$$\theta_k^i = \theta_k^{i-1}$$

where  $p(\theta_k | \theta_{k-1})$  and  $L(z_k | \theta_k)$ , respectively, are the sample set of prior and likelihood as shown in Eq. (8). The likelihood is the same as OBM, but the prior is given as a set of samples in RBM. Since it is not easy to express the sets of samples as equations, the samples are directly used as prior in the same way as PF. As a simple method to use samples itself as prior, samples obtained at the previous step

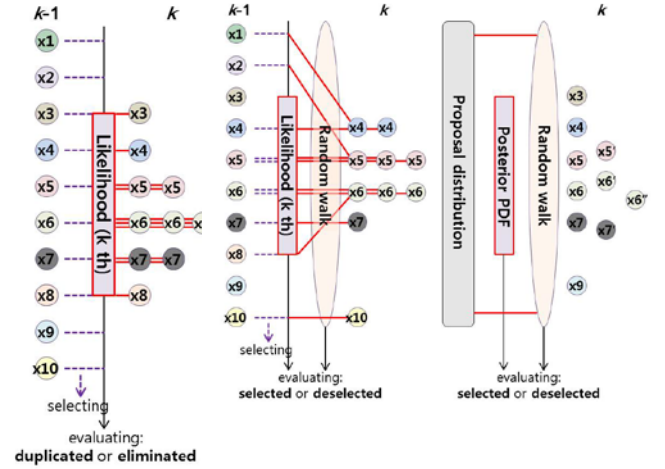


Figure 5. Illustration of sampling characteristic

replace the proposal distribution in MCMC as shown in Eq. (13). In this case, all prior samples are not selected for new sample,  $\theta_k^*$ , because  $\theta_k^*$  is randomly selected from prior. This is different from PF where all prior samples once are selected then updated by applying the weight. In this context, the relationship of samples between previous step and current step in OBM is similar to that of PF rather than that of RBM since the posterior in OBM can be regarded the result from accumulated weight in PF. These characteristics of sampling in the three methods are illustrated in the Figure 5.

### 2.4. Comparison of the Three Methods with the Same Configuration

In this subsection, the differences between the three methods are discussed using a same configuration. The model parameter estimation is focused rather than the state estimation, so the Eq. (11) is used for the state transition function or degradation model; that is, the system state is calculated from the estimated parameter without state transition. The procedure of three methods are illustrated as shown in Figure 6 and summarized in Table 1. The Figure 6(a) represents sequential updating process, in which distribution information to update is disseminated by

Step	PF	OBM	RBM
initial distribution	$\theta_0 \sim U(-25, 50)$	$U(-25, 50)$	$\theta_0 \sim U(-25, 50)$
prediction (prior)	$p(\theta_k   \theta_{k-1})$	$p(\theta   \mathbf{z})$ (Eq. (12))	$p(\theta_k   \theta_{k-1})$
update (likelihood)	$p(z_k   \theta_k^i)$ (Eq. (8))		$p(z_k   \theta_k)$ (Eq. (8))
sampling	resampling	MCMC	MCMC

Table 1. The procedure of the three methods

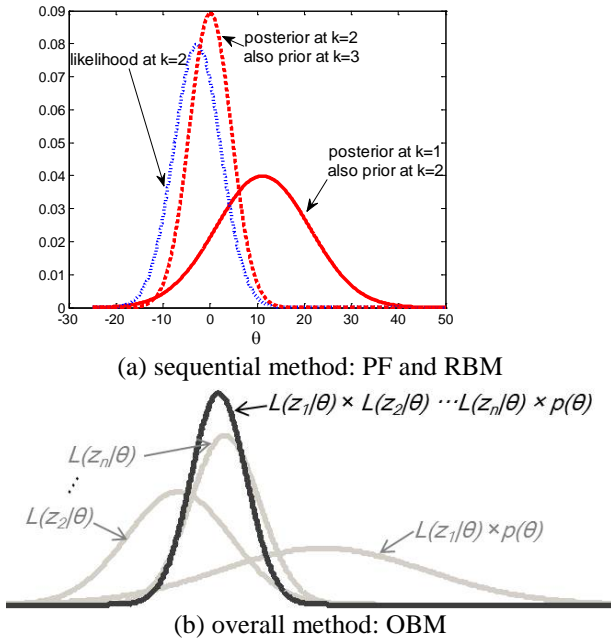


Figure 6. Illustration of the Bayesian inference

samples. The posterior at  $k=1$  depicted as solid curve becomes the prior at  $k=2$  which is multiplied by likelihood at  $k=2$  depicted as dotted curve to obtain the posterior distribution at  $k=2$  depicted as dashed curve, which also becomes the prior at the next step. On the other hand, the posterior distribution in the overall method depicted as black solid curve in the Figure 6(b) is obtained resulting from multiplying all likelihood functions and prior distribution depicted as blur solid curve. Once the posterior is obtained, samples are drawn using MCMC. The more detailed procedure of three methods is listed in the Table 1. First, initial distribution is prerequisite for PF and is assumed as  $\theta_0 \sim U(-25,50)$  which is wide enough not to affect the posterior distribution. For OBM and RBM, initial distribution is not required, but is given for the same configuration because initial distribution of PF corresponds to the prior information. Next, prediction and update step is performed sequentially in PF and RBM, while the posterior distribution expressed in the Eq. (12) is used for OBM. To implement the procedure, resampling and MCMC, respectively, are used for PF and RBM as well as OBM.

The three methods are performed at every discrete time step, and the results of estimated parameter using PF are shown in Figure 7 and Table 2. In the figure, it can be found that the PDF shape becomes sparse as the update is progressed. Similar results can be observed in the case of RBM. Table 2 lists the Kolmogorov-Smirnov (K-S) test results that are the maximum difference of CDF value between the posterior distribution obtained by grid method (Haldar & Mahadevan, 2000) and the distribution from the three methods. The CDF error between posterior obtained by grid method and sampling results becomes larger as the update is progressed

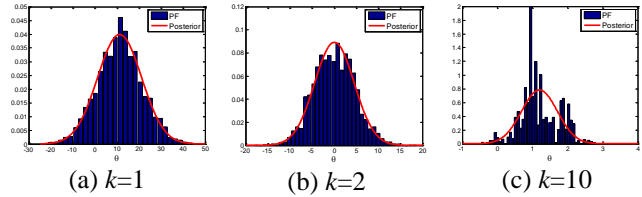


Figure 7. The parameter estimation results from PF

in both PF and RBM. The reason is that sampling error caused by expressing the posterior distribution is accumulated as time evolves. On the other hand, the table shows that the samples fairly well represent the posterior distribution at all times in OBM since there is no accumulated sampling error.

In the usage aspect, PF and RBM are related with the time evolution, while OBM uses batch data for estimating the posterior distribution; PF and RBM are useful for estimating in real time as the particles are directly used for the prior information, and it is not necessary to preserve the data. On the other hand, OBM has no cumulated error and no need to suffer from utilizing prior information. PF and RBM are fundamentally the same method except sampling schemes, and if there is no cumulated error in the two methods, they are the same as OBM. In the next section, performances of the three methods are evaluated as well as the differences between the methods are discussed.

### 3. CRACK GROWTH PROBLEM FOR COMPARISON OF THE THREE METHODS

#### 3.1. Crack Growth Model

In this paper, a physical model for crack growth in an aircraft panel is used to compare the three methods: PF, OBM and RBM. This example is to estimate damage growth parameters in Paris model (Paris, 1963) based on the measured crack size over a number of cycles (one cycle corresponds to one flight), which was addressed by An et al.(2012) When the stress range due to the pressure differential is  $\Delta\sigma$ , the rate of damage growth can be written using the Paris model (Paris, 1963) as

$$\frac{da}{dN} = C(\Delta K)^m, \quad \Delta K = \Delta\sigma\sqrt{\pi a} \quad (14)$$

where  $a$  is the half crack size,  $N$  is the number of cycles,  $\Delta K$  is the range of stress intensity factor, and  $m$  and  $C$  are damage growth parameters.

The synthetic measurement data are used for prognostics in

	$k=1$	$k=2$	$k=10$
PF	0.0262	0.0326	0.1186
OBM	0.0234	0.0286	0.0260
RBM	0.0278	0.0512	0.1130

Table 2. The K-S test to compare three methods

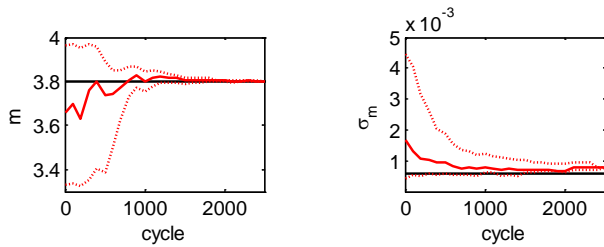


Figure 8. Convergence process of estimated parameters

this example, which are generated by (a) assuming that the true parameters,  $m_{true}$  and  $C_{true}$ , are known; (b) calculating the true crack sizes based on Eq. (14); and (c) adding random noise to the true crack size data. Once the synthetic data are obtained, the true values of crack sizes as well as the true values of model parameters are not used in the prognostics process. In this paper, the following values of parameters are used:  $m_{true} = 3.8$ ,  $C_{true} = 1.5 \times 10^{-10}$ ,  $\Delta\sigma = 78.6\text{MPa}$ , and initial half crack size,  $a_0 = 10\text{mm}$ . Also, it is assumed that the measurements are performed at every 100 cycles, and noise is uniformly distributed between -1mm and +1mm. In this study, the measurement noise is also considered as an unknown parameter.

### 3.2. Comparison of the Three Methods from Two Parameter Estimation

First, it is assumed that model parameter  $m$  and measurement noise  $\sigma_m$  are unknown and have prior  $U(3.3,4.0)$  and  $U(0,5)$ mm, respectively. In this case, true value of model parameter  $C$  is used. Figure 8 shows the results from PF, where the parameters are converged to the true values as the number of cycles increase; i.e., more data are obtained. Other two methods also show a similar convergence. The K-S test results listed in the Table 3 shows that the error in PF is the largest among the three methods. However, the result of crack growth prediction from PF with estimated parameters at 1000 cycles is fairly good as shown in the Figure 9. Also, the error between the

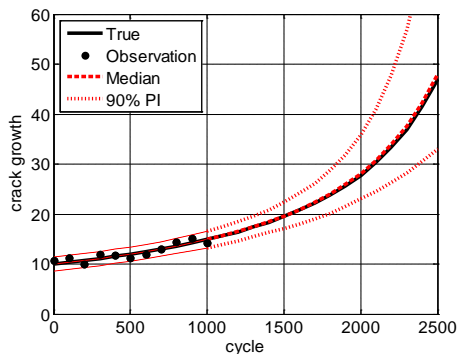


Figure 9. Crack growth prediction with estimated parameters at 1000 cycles

PF	OBM	RBM
0.0987	0.0632	0.0871

Table 3. The K-S test to compare three methods at 1000 cycles

	1000 cycle		1500 cycle		2000 cycle	
	median	std.	median	median	std.	median
true crack size	14.88		19.47	14.88		19.47
PF error (%)	14.95	1.02	19.62	14.95	1.02	19.62
OBM error (%)	14.95	1.08	19.71	14.95	1.08	19.71
RBM error (%)	14.92	1.03	19.68	14.92	1.03	19.68

Table 4. The median and std. of predicted crack growth and the error with the true value

	PF	OBM	RBM
data added in stages	1.0	2.13	1.70
batch data existence	1.0	0.08	1.70

Table 5. Relative computational costs of three methods

true crack growth and median of predicted crack growth ( $|a_{true} - a_{prediction}| / a_{true} \times 100$ ) and its standard deviation (std.) are listed in the Table 4. In the table, the errors of three methods are close, but the error of PF is the smallest, even though the error at the parameter estimation is the other way around. This means that while the parameter estimation cannot represent well the distribution shape due to sample duplication, the prediction results from PF can be accurate because information about the distribution are well contained in the duplicated samples. Therefore, it can be concluded that the future behavior of system can be predicted by estimating the model parameters based on the three methods.

Although the three methods represent similar results, Table 5 shows a difference in computational time, which is normalized by that of PF until 2500 cycles. In the case that data are added in stages (new data comes in at every 100 cycles), PF has the best efficiency. On the other hand, if batch data are available (all 25 data sets are available), computational time of OBM is considerably reduced in contrast to PF and RBM. This is because OBM can evaluate the posterior distribution by multiplying all likelihoods at the same time as in Eq. (9).

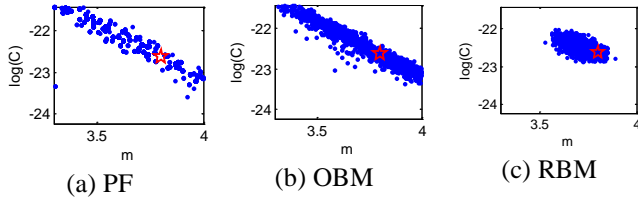


Figure 10. Samples of estimated parameters at 1000 cycles

### 3.3. Comparison of the Three Methods from Three Parameter Estimation

Next, two model parameters  $m$  and  $C$ , and measurement noise  $\sigma_m$  are considered as unknown parameters.  $m$  and  $\sigma_m$  have the same prior as in Section 3.3, and  $\log(C)$  has prior  $U(\log(3E-11), \log(5E-10))$ . The estimated model parameters from the three methods at 1000 cycles are shown in the Figure 10 with the true values in a star symbol. In this

case,  $m$  and  $C$  are strongly correlated, and the difference in sampling results caused by the different procedures is better exposed than the case of two parameters. Although the samples from PF look scattered due to duplication of high weighted particles, the shape is similar to that of OBM, which is close to the exact posterior distribution. On the other hand, the samples from RBM are less scattered because some samples at  $t-1$  are selected repeatedly.

RUL prediction is an ultimate goal in prognostics, and the results from three methods are shown in the Figure 11. After 1000 cycles, the median of predicted RUL is fairly accurate in PF and OBM. RBM produces somewhat worse results than PF and OBM, which is caused by the estimated parameters that do not represent overall posterior distribution as shown in the Figure 10(c). The results of RUL prediction are evaluated by using established prognostics metrics (Saxena et al., 2009) to compare the three methods more objectively. In this paper, five performance metrics are used to evaluate the three methods; prognostic horizon (PH),  $\alpha$ - $\lambda$  accuracy, relative accuracy (RA), cumulative relative accuracy (CRA), and convergence.

PH is defined as the difference between the time index  $t_i$  when more than  $\beta$  % of the predictions distribution first meet the  $\alpha$  accuracy zone and the time index  $t_{EOL}$  for end of life (EOL). The best score for PH is obtained when an algorithm always predicts within the desired accuracy zone. In  $\alpha$ - $\lambda$  accuracy,  $\alpha$  and  $\lambda$ , respectively, are an accuracy modifier and time window modifier.  $\lambda = 0$  when prediction is started,  $t_p$ , and  $\lambda = 1$  at  $t_{EOL}$ . The result of  $\alpha$ - $\lambda$  accuracy becomes true or false. True is when more than  $\beta$  % of the predictions distribution at  $t_\lambda = t_p + \lambda(t_{EOL} - t_p)$  is included in  $\alpha$ - $\lambda$  accuracy zone. RA is determined based on the error between true and prediction RUL at  $t_\lambda$ . CRA is the same to the accumulated RA from  $t_p$  to  $t_{EOL}$ . RA and CRA have the range from zero to one, and perfect value is one. The median of prediction RUL is used for RA and CRA. Lastly, convergence can be represented by the Euclidean distance between the center of mass of the area under the curve  $M(i)$  and  $(t_p, 0)$ . The lower the distance is, the faster the convergence is. In this study, the following conditions are used;  $\beta = 50\%$ ,  $t_p = 600$ , and  $t_{EOL} = 2400$ , and the results are

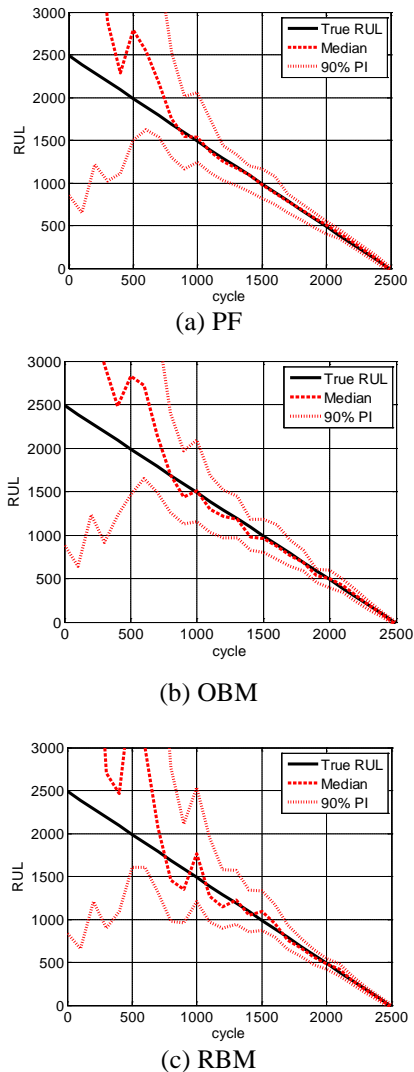


Figure 11. The RUL prediction

	PH ( $\alpha = 10\%$ )	$\alpha$ - $\lambda$ accuracy ( $\alpha = 10\%$ , $\lambda = 0.5$ )	RA ( $\lambda = 0.5$ )	CRA	Convergence ( $M(i)$ is a non-negative prediction error)
PF	<b>1600</b>	<b>true</b>	<b>0.9940</b>	<b>0.9391</b>	735
OBM	<b>1600</b>	<b>true</b>	0.9731	0.9262	642
RBM	1300	false	0.8953	0.8958	<b>524</b>

Table 6. Performance evaluation for three methods



listed in the Table 6.

In the table, the best performances are represented with a bold-typed text. The reason why RBM has the worst performance is that the prior given in samples cannot be used effectively, so there is much error as the time evolves as shown in the Figure 10(c) which does not represent the posterior distribution well. But, the convergence performance of RBM is the best because the median of RUL at  $t_p=600$  is much higher than other two methods. The performances of PF and OBM are similar, but PF has a little better performance than OBM. The reason is that sampling in OBM is not an easy task, especially under correlated parameters. It is difficult to confirm the convergence of sampling results at all time steps, because there might be improper samples. On the other hand, there are almost no samples which are not satisfied posterior distribution, and RUL can be exact since it is obtained in consequence of combination of the model parameters; scanty samples phenomenon is not too much trouble in the problem.

**4. DISCUSSION ABOUT THE DIFFERENCE BETWEEN THE THREE METHODS**

In this section, the differences between the three methods are summarized. In conclusion, PF is useful for estimating in real time, and OBM is efficient when batch data exist. RBM is also for estimating in real time even though it has

	PF	OBM	RBM
initial distribution	should be assumed	not necessary	not necessary
levels of difficulty for performance	simple	complex	normal
accuracy of model parameter estimation	good	very good (but it is not easy to obtain it)	normal
evaluation of RUL prediction	very good	good	fair
computational costs (data added in stages)	low	high	lower than OBM
computational costs (batch data existence)	normal	very low	high
proper usage	progressed problem by time step with proper initial distribution	when batch data exist	-

Table 7. Summary of the difference between the three methods

not been considered as a time concept, and the basic theory is the same to the PF. However, sampling method in PF (resampling) is more effective than one of RBM. The resampling method is a simple and effective way to represent the prior distribution. The sampling result from OBM is more accurate than that of PF because the latter can be changed from the initial distribution and shows scanty samples phenomenon. It is, however, difficult to obtain the accurate result because result from OBM can be significantly different from the proposal distribution. Since both PF and OBM have pros and cons, they can be used simultaneously in order to compensate for each other. Lastly, PF and RBM are fundamentally the same method except sampling schemes, and if there is no cumulated sampling error in the two methods, they will be the same to OBM. Summary of the difference between the three methods are listed in the Table 7.

**5. CONCLUSION**

In this paper, the three methods to estimate model parameters in the physics-based prognostics are compared and discussed; PF, OBM, and RBM are introduced with a simple mathematical example and evaluated by using established prognostics metrics with crack growth problem. The basic theory of RBM is the same to the PF in terms of estimating in real time. However, sampling methods of the two methods are different from each other, and sampling method in PF is more effective than one of RBM. On the other hand, OBM utilizes all the data available up to current time and evaluate the posterior distribution with one set of samples. The sampling result from OBM could be more accurate than that of PF. Consequently, PF and OBM have pros and cons, the former is useful for estimating in real time, but there is scanty samples phenomenon, while the latter is efficient under a batch of measurement data and has adequate samples, but the prediction results somewhat lack the accuracy comparing to PF. Therefore, they can be used complementally; e.g., PF is used from the latter stage after OBM is used until the early stage of growing damage state.

**ACKNOWLEDGMENTS**

This work was supported by the International Cooperation of the Korea Insitute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea government Ministry of Knowledge Economy (No. 20118520020010).

**REFERENCES**

An, D., Choi, J. H., Kim, N. H., & Pattabhiraman, S., (2011). Fatigue Life Prediction based on Bayesian Approach to Incorporate Field Data into Probability Model. *Structural Engineering and Mechanics*, vol. 37(4), pp. 427-442.

An, D., Choi, J. H., Schmitz, T. L., & Kim, N. H., (2011). In-Situ Monitoring and Prediction of Progressive Joint

- Wear using Bayesian Statistics, *Wear*, vol. 270(11-12), pp. 828-838.
- An, D., Choi, J. H., & Kim, N. H., (2012). Identification of Correlated Damage Parameters under Noise and Bias Using Bayesian Inference. *Structural Health Monitoring*, vol. 11(3), pp. 292-302.
- Andrieu, C., Freitas, de N., Doucet, A., & Jordan, M., (2003). An Introduction to MCMC for Machine Learning. *Machine Learning*, vol. 50(1), pp. 5-43.
- Bayes, T., (1763). An Essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London*, vol. 53, pp. 370-418.
- Campillo, F., & Rossi, V., (2009). Convolution Particle Filter for Parameter Estimation in General State-Space Models. *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, pp. 1063-1072.
- Daigle, M., & Goebel, K., (2011). Multiple Damage Progression Paths in Model-based Prognostics. *Aerospace Conference, 2011 IEEE*.
- DeCastro, J. A., Tang, L., Loparo, K. A., Goebel, K., and Vachtsevanos, G., (2009). Exact Nonlinear Filtering and Prediction in Process Model-based Prognostics. *Annual Conference of the Prognostics and Health Management Society*.
- Gašperin, M., Juričić, Đ., Boškoski, P., & Vižintin, J., (2011). Model-based Prognostics of Gear Health using Stochastic Dynamical Models. *Mechanical Systems and Signal Processing*, vol. 25, pp. 537-548.
- Giurgutiu, V., (2008). *Structural Health Monitoring: with Piezoelectric Wafer Active Sensors*, Academic Press (an Imprint of Elsevier). Burlington, MA.
- Glynn, P. W., & Iglehart, D. L., (1989). Importance Sampling for Stochastic Simulations. *Management Science*, vol. 35(11), pp. 1367-1392.
- Haldar, A., & Mahadevan, S., (2000). *Probability, reliability, and statistical methods in engineering design*. New York: John Wiley & Sons, Inc.
- Julier, S. J., & Uhlmann, J. K., (1997). A New Extension of the Kalman Filter to Nonlinear Systems. *In Proc. of AeroSense: The 11th Int. Symp. on Aerospace/Defence Sensing, Simulation and Controls*.
- Kalman, R. E., (1960). A New Approach to Linear Filtering and Prediction Problems. *Transaction of the ASME—Journal of Basic Engineering*, vol. 82(1), pp. 35-45.
- Li, P., Goodall, R., & Kadiramanathan, V., (2003). Parameter Estimation of Railway Vehicle Dynamic Model using Rao-Blackwellised Particle Filter. *In Proceedings of the European Control Conference*. Cambridge, UK.
- Luo, J., Pattipati, K.R., Qiao, L., & Chigusa, S., (2008). Model-based Prognostic Techniques Applied to a Suspension System. *IEEE Transactions on System, Man and Cybernetics*, vol. 38(5), pp. 1156-1168.
- Orchard, M. E., & Vachtsevanos, G. J., (2007). A Particle Filtering Approach for On-Line Failure Prognosis in a Planetary Carrier Plate. *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 7(4), pp. 221-227.
- Paris, P. C., & Erdogan, F., (1963). A Critical Analysis of Crack Propagation Laws. *ASME Journal of Basic Engineering*, vol. 85, pp. 528-534.
- Park, B. J., Zhang, Y., Lord, D., (2010). Bayesian Mixture Modeling Approach to Account for Heterogeneity in Speed Data. *Transportation Research Part B-methodological*, vol. 44(5), pp. 662-673.
- Payne, S. J., (2005). A Bayesian Approach for the Estimation of Model Parameters from Noisy Data Sets. *IEEE Signal Processing Letters*, vol. 12(8), pp. 553-556.
- Rago, C., Prasanth, R., Mehra, R. K., & Fortenbaugh, R., (1998). Failure Detection and Identification and Fault Tolerant Control using the IMM-KF with applications to the Eagle-Eye UAV. *Proceedings of the 37th IEEE, Conference on Decision & Control*. Tampa, Florida, USA, December.
- Ristic, B., Arulampalam, S., & Gordon, N., (2004). *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House.
- Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K., (2009). On Applying the Prognostic Performance Metrics. *Annual Conference of the Prognostics and Health Management Society*.
- Schwabacher, M. A., (2005). A survey of data-driven prognostics. *in AIAA Infotech@Aerospace Conference*. Reston, VA.
- Storvik, G., (2001). Particle Filters in State Space Models with the Presence of Unknown Static Parameters. *IEEE Transactions on, Signal Processing*, vol. 50(2), pp. 281-289.
- Wang, W., Liao, S., & Xing, T., (2009). Particle Filter for State and Parameter Estimation in Passive Ranging. *Intelligent Computing and Intelligent Systems (ICIS), 2009 IEEE International Conference on*.
- Yan, J. & Lee, J., (2007). A Hybrid Method for On-line Performance Assessment and Life Prediction in Drilling Operations. *in IEEE International Conference on Automation and Logistics*.
- Zio, E., & Peloni, G., (2011). Particle Filtering Prognostic Estimation of the Remaining Useful Life of Nonlinear Components. *Reliability Engineering and System Safety*, vol. 96(3), pp. 403-409.

## BIOGRAPHIES

**Dawn An** received the B.S. degree and M.S. degree of mechanical engineering from Korea Aerospace University in 2008 and 2010, respectively. She is now a joint Ph.D. student at Korea Aerospace University and the University of Florida. Her current research is focused on the Bayesian inference, correlated parameter identification and the methodology for prognostics and health management and structural health monitoring.

**Joo-Ho Choi** received the B.S. degree of mechanical engineering from Hanyang University in 1981, the M.S. degree and Ph.D. degree of mechanical engineering from Korea Advanced Institute of Science and Technology (KAIST) in 1983 and 1987, respectively. During the year 1988, he worked as a Postdoctoral Fellow at the University of Iowa. He joined the School of Aerospace and Mechanical Engineering at Korea Aerospace University, Korea, in 1997 and is now Professor. His current research is focused on the reliability analysis, design for life-time reliability, and prognostics and health management.

**Nam Ho Kim** received the B.S. degree of mechanical engineering from Seoul National University in 1989, the

M.S. degree and Ph.D. degree of mechanical engineering from Korea Advanced Institute of Science and Technology (KAIST) in 1991 and the University of Iowa in 1999, respectively. He worked as a Postdoctoral Associate at the University of Iowa from 1999 to 2001. He joined the Dept. of Mechanical & Aerospace Engineering at the University of Florida, in 2002 and is now Associate Professor. His current research is focused on design under uncertainty, design optimization of automotive NVH problem, shape DSA of transient dynamics (implicit/explicit) and structural health monitoring.

# Structural Health Monitoring on Metallic Aircrafts Using Flexible and Bulk PZT Transducers: Case of Corrosion Detection and Crack Localization

Hamza Boukabache<sup>1</sup>, Christophe Escriba<sup>2</sup>, Sabeha Zedek<sup>3</sup> and Jean-Yves Fourniols<sup>4</sup>

<sup>1,2,4</sup> *C.N.R.S-LAAS French national center for scientific research, Toulouse, 31400, FRANCE*

*hboukaba@laas.fr  
cescriba@laas.fr  
fourniols@laas.fr*

<sup>3</sup> *Université de Toulouse, INSA - Toulouse, 31400, France*

*sfzedek@laas.fr*

## ABSTRACT

This work focus on the structural health monitoring of aircrafts parts specimen structures made of 2024 Aluminum alloys. In this paper we demonstrate the feasibility of a new non destructive control method capable to probe very large structures within a short time. The method we developed is based through a wide piezoelectric sensors network on a smart comparison between two acoustic signatures: the healthy structure response captured before the commissioning of the plane and “an after flight” response. The sensors network exploits the capability of piezoelectric patches to generate/measure specific Lamb wave’s modes. The system is therefore dynamically configured to localize mechanicals flaws using an algorithm that operates using different techniques like pitch-catch and pulse-echo.

An analytic study is performed and tests to prove the proposed method feasibility on corroded structures specimens are provided at the end of this paper.

## 1. INTRODUCTION

Damage detection and structural health monitoring is one of the major challenges that face aircraft manufacturers. With the perpetual increase of structures complexity and the constant integration of composite materials to lighten aircrafts weight, the development of new non destructives tools capable to detect different flaws natures is more necessary than ever before. Based on Eddy current (Henry A. Sodano, 2007), ultrasonic’s radar (Victor Giurgiutiu, 2004) or thermal imaging (McIntosh, Greg, 1996) these widely developed and used techniques are locals, very

accurate but definitely unsuitable to probe a complete aircraft structures in a reasonable laps of time.

Active health monitoring and non destructive evaluation (NDE) systems for aircraft structures using guided waves and more specifically Lamb waves are relatively new and modern. These waves are able to travel from a single excitation point through very large distances in the medium where they propagate within a minimum energy loss. So far, the exploitation of these waves were widely focused on plate-like structures and others simple geometries made of a unique material like oil pipe-line and (Dixon, S et al. 2004). Based on echoes identification some commercial applications have therefore recently emerged (OLYMPUS, 2009) for local inspection. However in more complexes geometries, the multiple reflections due to the various structural features make the distinction between the different echoes quite difficult. Another problem emerges from the fact that Lamb wave are multimodal and dispersive by nature which makes the analysis very difficult in large scale structures. Many studies were focused on a Lamb wave frequency tuning (Victor Giurgiutiu, 2005) to generate a pure mode or just to maximize/minimize a particular mode response. Although, this technique gives very good results in cracks detection for metal plate like structures, it seems to be unsuited for complexes geometries which show a high thickness variation.

To circumvent these different problems, we propose a diagnosis method based on a smart comparison algorithm between two signatures: a healthy and a damaged one. The first signature is captured before the commission of the plane. The saved signals will be used in a database as a baseline for a future control. The signature of the structure is captured through a large sensors/actuators network that uses the capability of piezoelectric materials to generates/measures specific Lamb wave’s modes (Hamza Boukabache et al, 2011). The different interaction of theses waves into the probed structures are therefore exploited to

Hamza Boukabache et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

extract a significant signature. The diagnosis is thus based on a correlation between a baseline and a captured signature. This correlation is performed using a wavelet transform to have a time frequency comparison. The asset of this methodology is to have fast diagnostic and a quick localization of an eventual existing damage even in large scales aircraft structures whatever the nature of the damage. Another advantage of the proposed method is its capability to detect corrosion which represents one of the most critical damage on aerospace metallic structures. The detection of such flaw become a priority issue for the US Air Force (Dustin Thomas et al 2004) that spend 1.2 billion USD a year to repair corrosion on the KC-135 for example.

In this paper, the use of Lamb wave to extract a significant structure baseline is investigated. Therefore, an analytical study is performed and correlated to the experimental results. The generation of these specifics waves using piezoelectric wafer is also studied and the results are confronted to the multiple issues linked to the bonding with the host structure.

## 2. LAMB WAVES THEORY

Guided waves have a major importance in embedded non destructive control applications. In opposite to the reset of elastic waves, they can travel on longue distances and cover large areas. The created acoustic compression field stays confined between the structure's faces if the wavelength of the propagating wave is comparable to the structure thickness.

For isotropic materials the study of these waves begins with the general equation of dynamics applied to elastics materials:

$$\mu \nabla^2 \vec{u} + (\lambda + \mu) \nabla \nabla \cdot \vec{u} = \rho \frac{\partial^2 \vec{u}}{\partial t^2} \quad (1)$$

Where  $\vec{u}$  is the deformation vector,  $\lambda$  and  $\mu$  are the Lamé constants and  $\rho$  is the mass density.

For plate-like isotropic structures, the symmetry of the geometry creates invariance in the orthogonal direction which allows using Helmholtz decomposition to reduce  $\vec{u}$  into a sum of a potential vector and a potential scalar.

$$\vec{u} = \nabla \Phi + \text{curl } \vec{\Psi} \quad (2)$$

By developing the equation (1) using the relation (2) we find the differentials equations of elastic wave propagation.

$$\begin{cases} c_p^2 \Delta^2 \Phi = \frac{\partial^2 \Phi}{\partial t^2 \rho} & \text{with } c_p^2 = \frac{(\lambda + 2\mu)}{\rho} \\ c_s^2 \Delta^2 \vec{\Psi} = \frac{\partial^2 \vec{\Psi}}{\partial t^2 \rho} & \text{with } c_s^2 = \frac{\mu}{\rho} \end{cases} \quad (3)$$

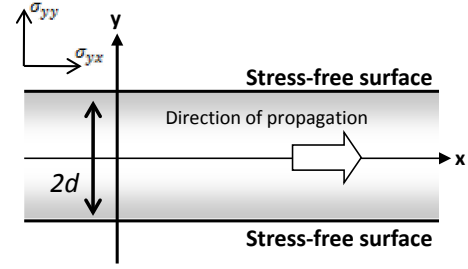


Figure 1. Plate-like structure with a thickness of  $2d$ . The upper and lower surfaces are unconstrained and stress-free.

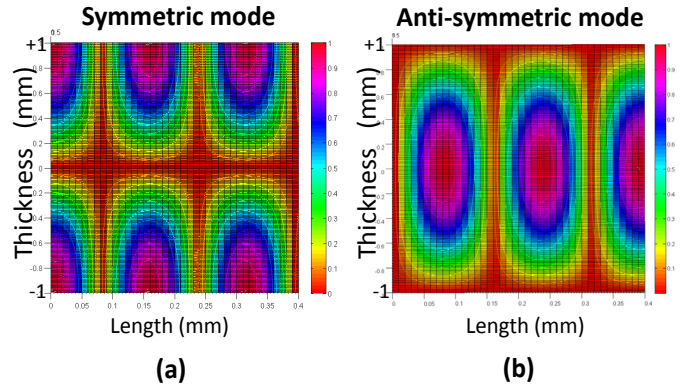


Figure 2. Numerical resolution of the deformation equation  $\vec{u}(t)$ . The excitation frequency is 200 kHz. The acoustic compression field is simulated for a normalized structure thickness. (a) The simulation result for the fundamental symmetric mode  $S_0$  (b) The fundamental anti-symmetric motion mode  $A_0$ .

With:

$$\Phi = \Phi(y) \cdot e^{i(k \cdot x - \omega t)} \quad (4)$$

$$\vec{\Psi} = (\Psi_x(y)\vec{e}_x + \Psi_y(y)\vec{e}_y + \Psi_z(y)\vec{e}_z) \cdot e^{i(k \cdot x - \omega t)} \quad (5)$$

Where,  $k$  is the wave number,  $\omega$  is the frequency;  $\Psi_x, \Psi_y, \Psi_z$  are the scalars components of  $\vec{\Psi}$ . The substitution of the equations (5) and (3) in the system (3) gives four scalars differentials equations that accept solutions in the form:

$$\begin{cases} \Phi = (A \cdot \cos py + H \cdot \sin py) e^{i(k \cdot x - \omega t)} \\ \Psi_x = (B \cdot \cos qy + G \cdot \sin qy) e^{i(k \cdot x - \omega t)} \\ \Psi_y = (E \cdot \cos qy + D \cdot \sin qy) e^{i(k \cdot x - \omega t)} \\ \Psi_z = (C \cdot \cos qy + F \cdot \sin qy) e^{i(k \cdot x - \omega t)} \end{cases} \quad (6)$$

Where,  $p^2 = \omega^2/c_p^2 - k^2$  et  $q^2 = \omega^2/c_s^2 - k^2$  and  $A, B, C, D, E, F, G, H$  are the integration constants. These constants are found only by setting the boundary conditions. We assume that structures thickness is equal to  $2d$  and its upper and lower surfaces are stress-free:

$$\begin{aligned}\sigma_{yx}(x, -d) &= -\sigma_{yx}(x, d) = 0 \\ \sigma_{yy}(x, -d) &= \sigma_{yy}(x, d) = 0\end{aligned}\quad (7)$$

In this case the integrations constants are given by the system of equations (8). Its complete resolution leads to the particle movement equation  $\vec{u}(t)$ .

The calculation of the coefficient matrix determinant yields to the characteristic equation of guided waves. This determinant is constituted by the smaller determinants couples (A, B), (C, D), (E, F), (G, H) where each one specifies a different kind of motion. The two first couples correspond respectively to the symmetric and anti-symmetric Lamb wave's modes and the rest pairs correspond to symmetric and anti-symmetric SH waves.

$$\begin{pmatrix} C_{11} & C_{12} & 0 & 0 & 0 & 0 & 0 & 0 \\ C_{21} & C_{22} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & C_{33} & C_{34} & 0 & 0 & 0 & 0 \\ 0 & 0 & C_{43} & C_{44} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & C_{56} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{65} & C_{66} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & C_{77} & C_{78} \\ 0 & 0 & 0 & 0 & 0 & 0 & C_{87} & C_{88} \end{pmatrix} \begin{pmatrix} A \\ B \\ C \\ D \\ E \\ F \\ G \\ H \end{pmatrix} = 0 \quad (8)$$

With:

$$\begin{aligned}C_{11} &= -2ikp \sin pd & C_{12} &= [k^2 - q^2] \sin qd \\ C_{22} &= 2i\mu kq \cos qd & C_{21} &= [(\lambda + 2\mu)\alpha^2 + \lambda k^2] \cos pd \\ C_{43} &= 2ikp \cos pd & C_{44} &= [k^2 - q^2] \cos qd \\ C_{34} &= -2i\mu kq \sin qd & C_{33} &= [(\lambda + 2\mu)\alpha^2 + \lambda k^2] \sin pd\end{aligned}$$

The calculation of the two determinants (A, B), (C, D) yields for isotropic materials to Rayleigh-Lamb characteristic equation.

$$\frac{\tan pd}{\tan qd} = - \left[ \frac{4k^2 pq}{(k^2 - q^2)^2} \right]^{\pm 1} \quad (9)$$

Where, +1 corresponds to the symmetric mode whereas -1 corresponds to the anti-symmetric mode of motion.

This equation allows the calculation of dispersive curves and thus allows the identification of each propagation mode for a fixed operating frequency and a determined structure thickness. According to our diagnosis methodology that consists on a comparison between a healthy and damaged structure's signature, equation (9) will be useful to minimize the number of generated Lamb mode and thus minimize the complexity of the different interactions that may occurs in the structures thickness.

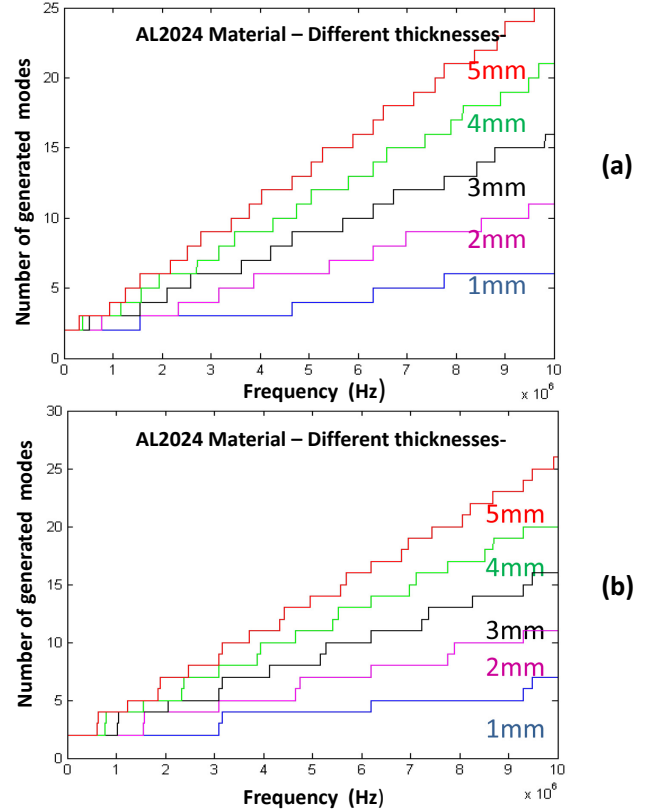


Figure 3. Number of coexisting Lamb wave modes in the symmetric (a) and anti-symmetric case (b) simulated for different operating frequencies and different thicknesses. [Simulation performed for AL2024]

### 3. LAMB WAVE GENERATION USING PIEZOELECTRIC SENSORS/ACTUATORS

#### 3.1. Piezoelectric sensor/actuator development

Many techniques are already described in the literature to generate Lamb waves. The most common effective's ones use ultrasonic transducers with a couplant (D. N. Alleyne et P. Cawley, 1996) to facilitate the transmission of ultrasonic energy from the transducer into the tested structure. These techniques allow a precise detection of mechanical flaws by a post identification of Lamb modes conversion (Victor Giurgiutiu, 2005) due to the interaction of the generated wave with the flaws. However to reach their optimum detection capability, these systems have to generate a pure lamb mode into the structure which is only possible on small areas where the structure's thickness is constant. These techniques are very local and therefore unsuitable for a network integration system intended to aeronautics' structures health monitoring application.

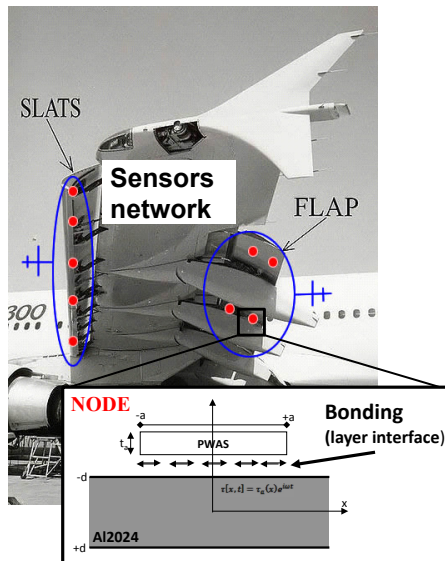


Figure 4 : Airbus A300 left wing. Schematics of the network sensor pasted on specifics mobile parts like the Slats or the kruger flaps. Areas that are often subject to aerodynamics forces and thus subject to damage due to the high cycle of use.

Therefore, the focus of our research turned on the study of small transducers made of piezoelectric material mounted directly onto the surface of the structure within minimal intrusion. The exploitation of the piezoelectric effect allows the generation of a shear stress directly through an interfacing layer into the structure (Victor Giurgiutiu, 2002). To do so, we used two kinds of transducers:

1. PI-DuraAcac P-876.A12 transducer of 61x35x0.5 mm made from a thin layer of PZT materials staked between two layers of kapton which makes is totally flexible (Cf. Figure 5). Because of its capacitive nature and its thin thickness, the transducer needs a high voltage amplifier between [100 V, 400V] to have a significant actuation. To perform this amplification we used the E-835 DuraAct™ Piezo Driver Module that provides 30W of power output with a voltage of 250V for an input signal of 10V. Authors' names are inverted (last name first); give the last name and initials (first and middle) for all authors of a particular work.
2. Custom transducer made from PZT-5H material designed to have a radial oscillation frequency of 200 kHz (Cf. Figure 6). The material properties fixe the sensor diameter to ~10 mm. The thickness was fixed to 1 mm which gives a second oscillation mode frequency equal to 2MHz.

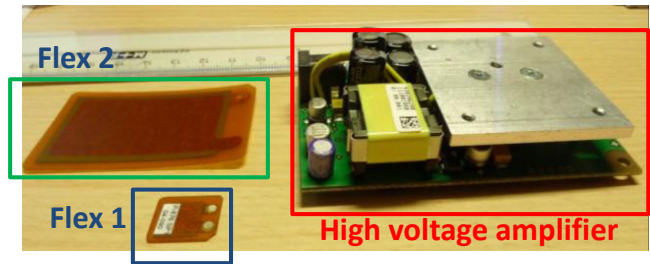


Figure 5 : DuraAct transducer with its power amplifier.

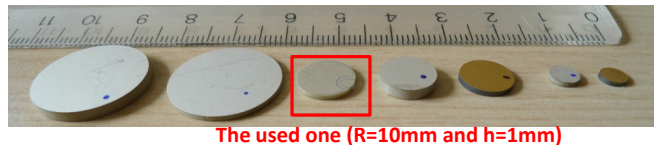


Figure 6 : PZT bulk transducers.

As shown in figure 3, to minimize the number of generated modes into the structure, we should use an excitation frequency that stays below 500 kHz. This condition minimizes the number of Lamb modes (Hamza Boukabache et al, 2011) and thus reduces the interference into the host structure which makes the interpretation of the acquired signals much easier. In opposite to what was already described in the literature (Victor Giurgiutiu, 2005), we have chosen to exploit both symmetrical and anti-symmetrical modes which is easier to generate than pure modes.

### 3.2. Experimental setup

To demonstrate the generation of Lamb waves into plates likes structures we installed a test bench based (Figure 7 and 8) on an National instrument acquisition card that offers up to 8 simultaneous acquisition channel of 14bits resolution and 2.5MSample/s speed. In the other hand the stimulus generation is based on an Agilent 3322 arbitrary waveform generation instrument. The communication with the PC is performed through an USB-VISA protocol which allows us to transfer the stimulus data directly from the data processing software to the instrument.

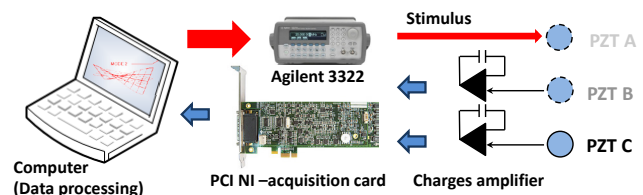


Figure 7: Synoptic of the test bench setup.

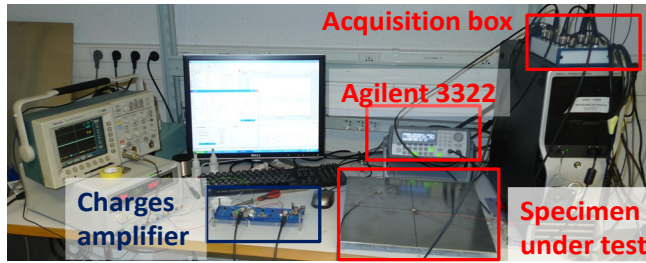


Figure 8: Synoptic of the test bench setup.

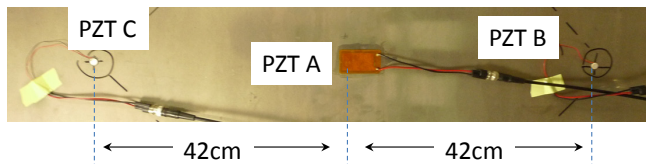


Figure 9: Synoptic of the test bench setup.

The chosen stimulus waveform is the pure tone sine modulated by a Hanning window which is commonly used in the literature (Hamza Boukabache et al, 2011).

In actuation mode, the excitation frequency was fixed to 200 kHz which minimizes the number of generated mode to 3 into an Al2024 plate of 3mm thickness.

When the piezoelectric transducer is used in sensor mode, a charge amplifier based on a capacitive feedback is also necessary to have an exploitable voltage signal. According to piezoelectric effect, when subject to a mechanical stress, the transducer generates electric charges  $Q$  [Coulombs] at its respective electrodes that are not directly exploitable. A capacitance  $C$  [Farads] should be added to perform the voltage conversion  $V=Q/C$ . According to this equation, when an operational amplifier is used, the capacitive feedback fixes the amplification. A step of signal processing is needed to filter the converted voltage. Experimental measurements showed that for our operating frequency a simple bandpass filter based on Sallen-Key analog circuit architecture presenting an attenuation factor of 40dB and a bandwidth of 10 kHz is enough to get an acceptable SNR. Actually this is only true for metallic alloys structures.

From figure 9, notice that the flexible transducer is used as an emitter when the PZT bulk transducers are used in passive sensing mode. As shown in figure 10 the time delay between the emission of PZT A and the reception of PZT B and C gives us the group velocity of the generated wave. This time delay is known as the Time of flight (TOF) of the wave. The 3 transducers are pasted with a distance 420mm from each other. From figure 10 we extract the time of flight which is equal  $TOF = 112\mu s$ . For the traveled distance, the wave speed is equal to 4200ms.

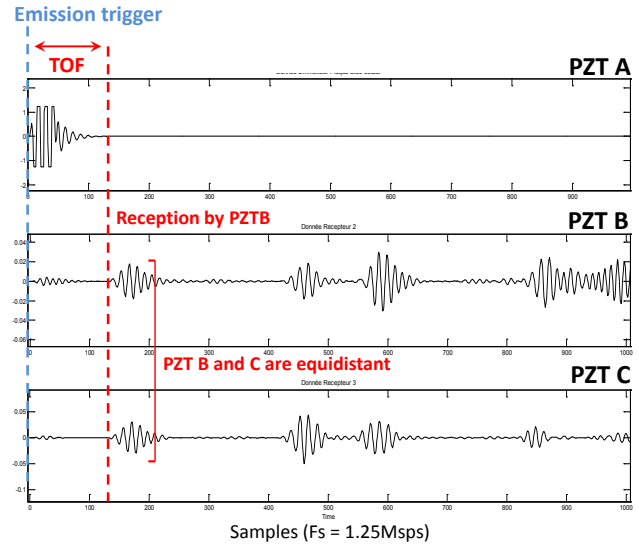


Figure 10 (a) Lamb wave generation into a structure specimen. Note that the time of flight between the emitter and the receptors (TOF) is  $112\mu s$

#### 4. DAMAGES DETECTION :

In order to detect mechanical flaws in complexes structures such as aeronautics flats, wingbox, intra and extra wing panel we developed an algorithm capable to diagnosis and localize damage by comparing the acquired signature to a base line captured before the commissioning of the plane. By minimizing the number of Lamb modes generated into the structure (figure 3) this method circumvents the issues linked to the multiple reflections and the different interactions into the structure.

Globally, there are two methods to detect a defect using guided waves.

- The simplest and the most standard one consist to extract one signature per node using a pulse-echo technique. Using an analogy with sonar world, pulse echo technique consists to emit an impulse and listens to the echoes reflected by the medium where the waves propagate. At final, this configuration gives as many signatures (signals) as the used number of nodes.
- A second more complex technique consists to use an emitter and a receptor. The used principle is based on pitch-catch method. This one consists to detect the deformation and distortions that occurred to the transmitted wave. A baseline is therefore systematically used



#### 4.1. Crack detection using pulse-echo based NDE technique:

Inspired from electromagnetic radar world, pulse-echo technique uses at least one node to probe a local area looking for mechanical flaws. The principle is based on the capability of the piezoelectric patch to act both as actuators and as sensors. The node emits a short pulse noted  $x(t)$  given by Eq.9 then it listens for echoes.

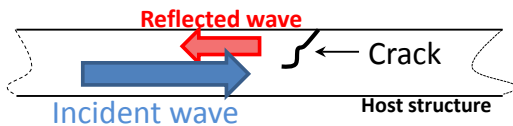


Figure 11. Pulse echo principle

For the simplest case where a pure Lamb mode is generated, the reflected signal could be assumed as a simple image of the emitted one  $x(t)$ . In this case we can suppose that the reflected signal  $r(t)$  from the crack is only delayed and attenuated.

$$r(t) = \alpha x(t - D) + \beta(t) \quad (9)$$

With:

- $\alpha$  : Attenuation factor (it depends on the structure nature)
- D: [s] Time propagation delay or time of flight (TOF)
- $\beta$ : [V] Noise factor

Thus, by knowing the group velocity of the generated pulse  $x(t)$  we can use an estimation of the time delay D and calculate the location of the damage.

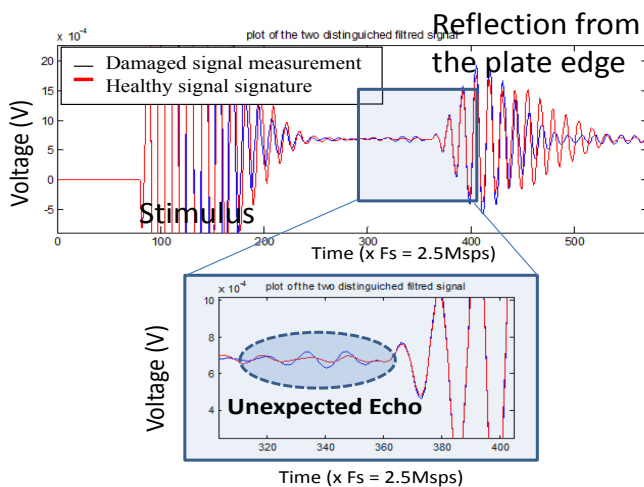


Figure 12. Superposition of the healthy signature and the damaged structure response.

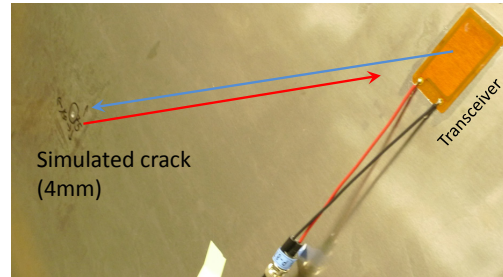


Figure 13. Specimen under test (Crack of 4mm)

From figure 12, the crack echo takes 128 $\mu$ s to perform a go and back from the transducer to the crack then from the crack to the transducer. We know that group celerity is 4200m/s into this material. Therefore, the crack is located at 27cm from the transceiver. Using at least 3 nodes we are capable to locate the damage using a basic triangulation algorithm.

#### 4.2. Corrosion detection using pitch catch embedded NDE:

Corrosion in aircrafts metallic structures is one of the most common problems that we face during planes life time. It generally appears into the hidden side of the structure where the thickness of the chemical surface processing is thinner. Therefore the corroded areas are hardly accessible to human beings which make their detection very difficult. Furthermore, the corrosion in Al2024 implies a thickness loss for the structure which weakens its mechanical proprieties.



Figure 14. A320 wingbox with corrosion flaw inside the wing. (Acknowledgments to Airbus and EADS-IW)

In order to detect corrosion into aeronautics Al2024 structures, we extracted healthy panels of 32cm x 32 cm x 3 mm that do not contain rivets or fasteners. We introduced artificially corrosion using a NaCl attack and thermal processing (Cf. Figure 14). The 5 specimens are listed in the table 1.

Specimen	Corrosion size (depth- area)
N4	Baseline (None- None)
N1	1μm x 230 cm <sup>2</sup>
N3	1μm x 38 cm <sup>2</sup>
N5	500μm x 56cm <sup>2</sup>
N6	1mm x 4cm <sup>2</sup>

Table 1. Specimens characteristics

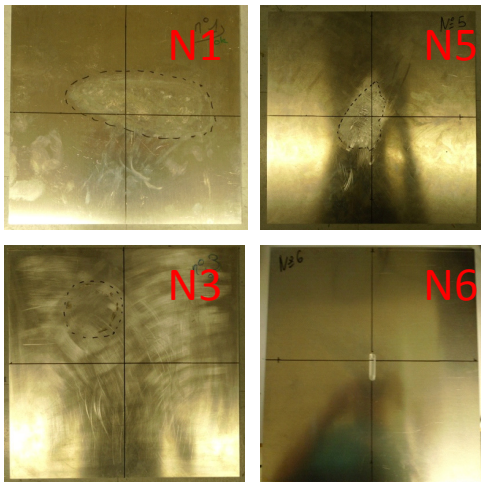


Figure 15. Photos of the corroded specimens (Acknowledgements to CIRMIAT's team who introduced the corrosions onto the specimens)

As shown in figure 16, pitch catch technique uses at least two nodes, one as an emitter and the second one as a simple passive sensor (the receptor). For a simple structure as shown in figure 15 the reflected signal captured by the first node contains the damage's echo  $x(t - D)$  plus the parasitic echoes due to geometry of the structure. However, in opposite to the crack damage case, the produced echoes are very weak in the corrosion case. In fact the corrosion produces a thickness loss that is very small (up to 10% of the global thickness) which make the reflected echo negligible. In the second hand the receptor sensor captures the transmitted signal noted  $T(t)$

$$T(t) = \gamma x(t + \Delta T) + \sum_i (\alpha_i x(t_i)) + \beta(t) \quad (10)$$

With:

- $\alpha_i$  : Attenuation factor
- $D_i$  : [s] Time propagation delay
- $\beta$ : [V] Noise factor
- $\gamma$  : Transmission deformation

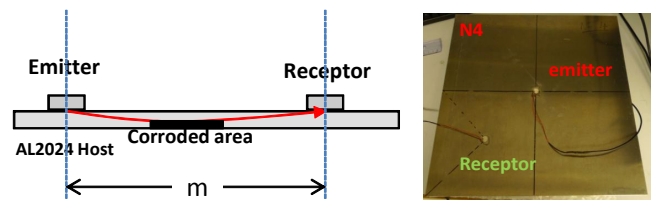


Figure 16. Pitch catch principle

The comparison between the different transmitted signals allows us to detect a corroded zone using a reference or a baseline acquired when the structure was healthy.

To perform the comparison we developped an algorithm based on wavelet transform. It allows us to decompose our transmitted signal into both frequency and time domaine which make the comparison much accurate. To performe the continus wavelet transform Eq .11 we used the morlet wavelet function as the temporal correlation coeficien  $\psi_{s,\tau}(t)$ .

$$g(s, \tau) = \int_{-\infty}^{\infty} f(t) \psi_{s,\tau}(t)^* dt \quad (11)$$

With :  $f(t)$  the transmitted signal.

The decompostision presented in figure 17 was performed on the temporal transmitted signal captured by the receptor for the healthy structure (specimen N4 presented in figure 16). The figure 17.b presents the percentage of energy for each wavelet coeficient in time-frequency domaine. By isolating the 200kHz coeficients that corresponds to our excitation frequency we plot the transmission profile of this frequency during its travel into the host plate.

Figure 18 shows the decomposition of signal N1. Notice the clear difference with the withness specimen Figure 17

For the plate structure N1 with a corroded area of 230 cm<sup>2</sup> with a thickness loss of 1μm , the transmitted wave had during its travel a different intetergaction with the host structure than for N4 (Baseline)

The superposition of the different transmitted profile (Cf. Figure 19) allows us to diagnosis the presence of corrosion into the structure. However, unlike pulse echo technique our methodology is unable to locate the corrosion. We only know that the corroded area is situated between the emitter and the receptor. Therefore the resolution of this method is totally linked to the distance between the transducer and thus to the fixed meshing.

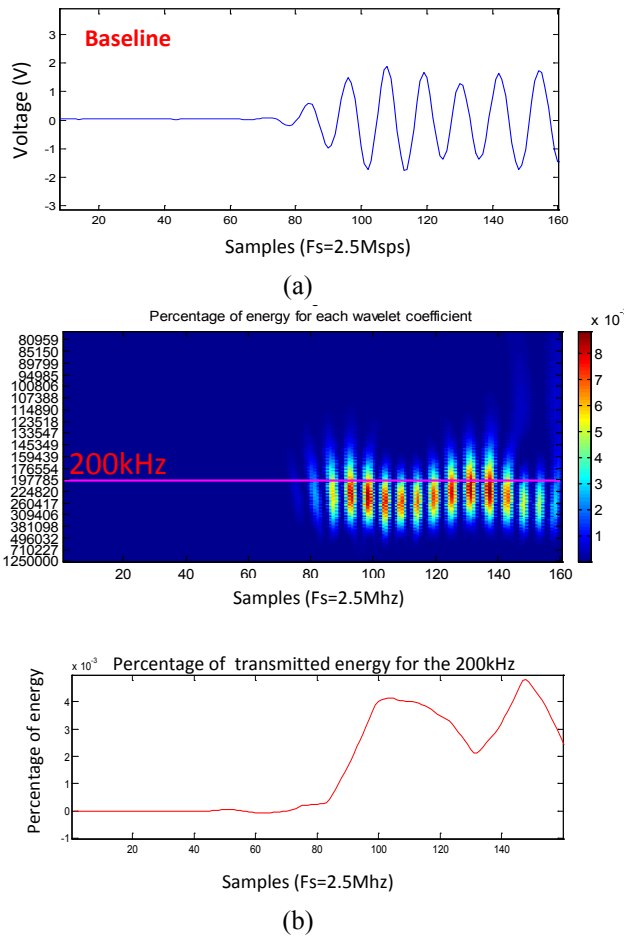


Figure 17. (a) The acquired transmitted signal for the witness specimen N4. (b) Wavelet decomposition of the temporal transmitted signal. We extract the profile of propagation of the 200kHz frequency

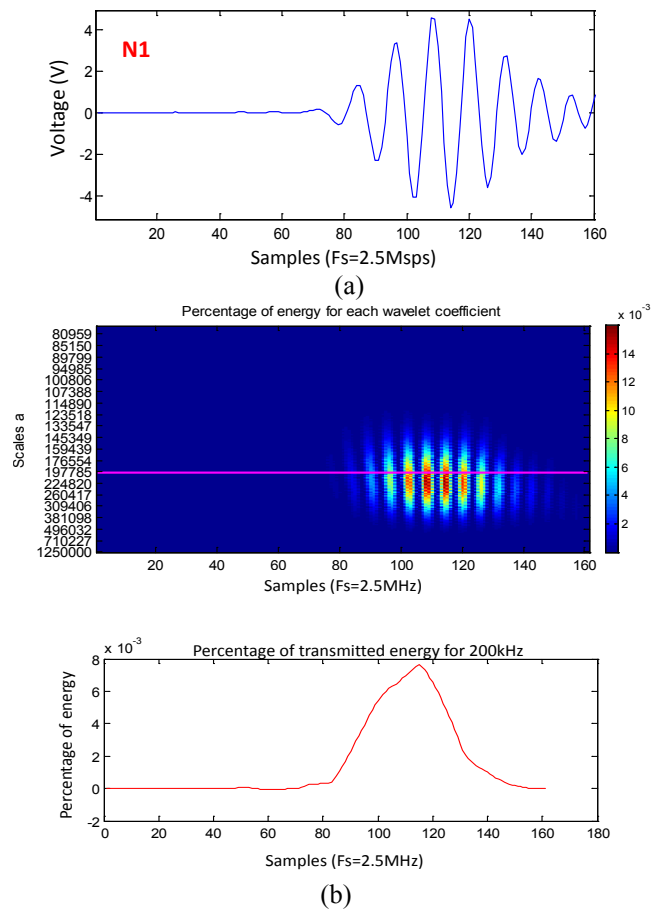


Figure 18. (a) The acquired transmitted signal for the specimen N1. (b) Wavelet decomposition of the temporal transmitted signal. We extract the profile of propagation of the 200 kHz frequency

From figure 19 it is clear that each specimen plate has its own signature. The measurements were performed 3 times for each plate using the same measurements conditions.

Notice that the more the corroded area is large the more is the amplitude of the first peak. The depth of the thickness loss seems to have also a direct link with the amplitude.

For the extreme case where the thickness loss is 1mm the amplitude of the transmitted signal is significantly low in opposite to all the other signals. Due to the high thickness loss (33%), a possible explanation would be the fact that most of the wave energy emitted by the actuator is reflected by the damage and only a low amount of energy is transmitted.

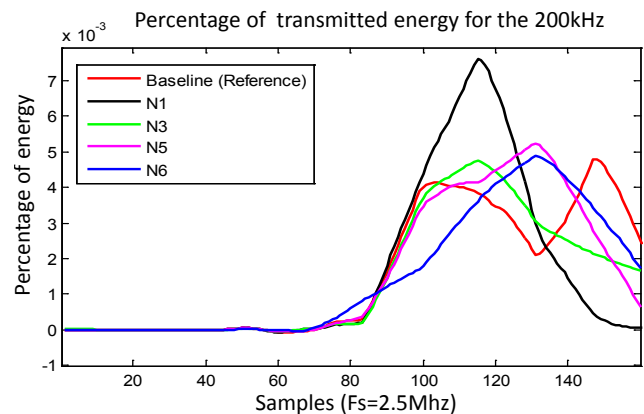


Figure 19. Superposition between the different transmissions signals for the different specimens

## 5. CONCLUSION

In this paper we demonstrate the feasibility of cracks and corrosion detections inside Al2024 structures using a unique piezoelectric technology and with the same sensor network.

The next step of our research is the hardware integration to make the nodes autonomous. An electronic architecture using reconfigurable system on chips are currently developed to support flaw detection, data processing and prognostic. The current technology is also tested for carbon composite structures where delaminating defects appears without any external sign.

The final aim is to develop embedded network system capable to monitor different kinds of aerospace structures whatever is the nature of their material.

## ACKNOWLEDGEMENT

This work is done in the frame of DIAGNOSTAT project which is funded by the Midi Pyrenees council and the Ministry of Industry. Many thanks to EADS-IW and Airbus group who provided the aircrafts specimens.

## REFERENCES

- Henry A. Sodano, (2007). *Development of an Automated Eddy Current Structural Health Monitoring Technique with an Extended Sensing Region for Corrosion Detection*, Structural Health Monitoring vol. 6 no. 2 111-119
- Victor Giurgiutiu, (2004) *Embedded-ultrasonics Structural Radar for In Situ Structural Health Monitoring of Thin-wall Structures*, Structural Health Monitoring June 2004 vol. 3 no. 2 121-140
- McIntosh, Greg, (1996) *Infrared thermography monitors composite consolidation*, Advanced Materials & Processes; Dec96, Vol. 150 Issue 6, p29
- Dixon, S et al. (2004). *Inspection of rail track head surfaces using electromagnetic acoustic transducers (EMATs)*. Insight - Non-Destructive Testing and Condition Monitoring, Volume 46, Number 6,
- OLYMPUS, (2009). *Flaw Detectors, BondMaster*, <http://www.olympus-ms.com/en/bondmaster1000eplus/>
- Victor Giurgiutiu, (2005). *Tuned Lamb Wave Excitation and Detection with Piezoelectric Wafer Active Sensors for*

*Structural Health Monitoring*, Journal of Intelligent Material Systems and Structures vol. 16 no. 4 291-305

- Hamza Boukabache et al, (2011) *Piezoelectric wafer active sensor network for aircraft structures damage localisation: pitch-catch method*. International Workshop on Structural Health Monitoring (IWSHM 2011), Stanford (USA), 13-15 Septembre 2011, pp.555-561
- D. N. Alleyne P. Cawley, (1996), *The excitation of Lamb waves in pipes using dry-coupled piezoelectric transducers.*, Volume 15, Number 1 (1996), 11-20, DOI: 10.1007/BF00733822
- Victor Giurgiutiu, (2002) *Lamb Wave Generation with Piezoelectric Wafer Active Sensors for Structural Health Monitoring*. Annual International Symposium on Smart Structures and Materials and 8th Annual International Symposium on NDE for Health Monitoring and Diagnostics, 2-6 March 2002, San Diego, CA
- Dustin Thomas et al, (2004) *Corrosion Damage Detection with Piezoelectric wafer Active Sensors*. Annual International Symposium on Smart Structures and Materials and 9th Annual International Symposium on NDE for Health Monitoring and Diagnostics, 14-18 March 2004, San Diego, CA

## BIOGRAPHIES



**Hamza BOUKABACHE** received his M.S in embedded aerospace electronics from the French national institute of applied science at Toulouse (INSA) in 2009 and the same year he got a Master research in micro and nano-systems from Paul Sabatier University at Toulouse. He is currently preparing his Ph.D. with the French national center for scientific research and Airbus FRANCE. Current research interests include Aerospace structure health monitoring using wide sensors network, energy harvesting and reconfigurable electronics development.

# A Tutorial for Model-based Prognostics Algorithms based on Matlab Code

Dawn An<sup>1</sup>, Joo-Ho Choi<sup>2</sup>, and Nam Ho Kim<sup>3</sup>

<sup>1,2</sup>*Korea Aerospace University, Goyang-City, Gyeonggi-do, 412-791, Korea*

*skal34@nate.com*

*jhchoi@kau.ac.kr*

<sup>3</sup>*University of Florida, Gainesville, FL, 32611, USA*

*nkim@ufl.edu*

## ABSTRACT

This paper presents a Matlab-based tutorial for model-based prognostics, which combines a physical model with observed data to identify model parameters, from which the remaining useful life (RUL) can be predicted. Among many model-based prognostics algorithms, the particle filter is used in this tutorial for parameter estimation of damage or a degradation model in model-based prognostics. The tutorial is presented using a Matlab script with 62 lines, including detailed explanations. As examples, a battery degradation model and a crack growth model are used to explain the updating process of model parameters, damage progression, and RUL prediction. In order to illustrate the results, the RUL at an arbitrary cycle are predicted in the form of distribution along with the median and 90% prediction interval.

## 1. INTRODUCTION

Although many prognostics methods have been presented in literature (Daigle & Goebel, 2011; DeCastro et al., 2009; Luo et al., 2008), it is still difficult for engineers to use them for their applications. The objective of this paper is to demonstrate how to use a prognostics method using a simple Matlab code as short as 62 lines.

Among different prognostics methods, the model-based approach is considered, which assumes that a physical model describing the behavior of damage or degradation is available. In this approach, the model parameters are often unknown and need to be identified as a part of the prognostic process. The method combines the model with measured data to identify the model parameters and predict

its behavior under future loading conditions. There are several methods to estimate model parameters, such as the Kalman filter (KF) (Kalman, 1960), Particle filter (PF) (Orchard & Vachtsevanos, 2007; Zio & Peloni, 2011; Li et al., 2003), and Bayesian method (BM) (An et al., 2011; An et al., 2012; Payne, 2005). In this paper, PF is employed because it can be used for a nonlinear model with non-Gaussian noise and is the most widely used in the field of prognostics.

The Matlab code is composed of 62 lines including detailed explanations, which is further divided into three parts: (1) problem definition, (2) prognostics using PF, and (3) post-processing. Users are required to modify the first part according to their application. For demonstration purposes, examples of battery degradation and crack growth are presented.

This paper is organized as follows: in Section 2, the overall process of model-based prognostics is explained with the Matlab code; in Section 3, the usage is explained with a battery degradation example; and in Section 4, various cases are described with a crack growth example, followed by conclusions in Section 5.

## 2. MODEL-BASED PROGNOSTICS

The process of model-based prognostics is illustrated in Figure 1, in which the degradation model is expressed as a function of usage conditions  $U$ , elapsed cycle or time  $t$ , and model parameters  $\theta$ . The usage conditions and time are given, while the model parameters characterizing the damage behavior should be identified. Then, the remaining useful life (RUL) which represents the remaining time to failure is calculated based on the estimated model parameters.

The model parameters are estimated using an algorithm such as PF by integrating the damage model with the

---

Dawn An et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

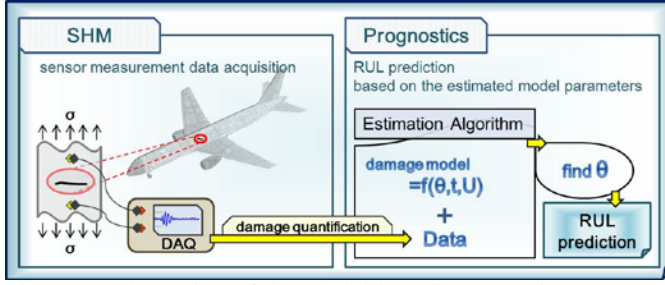


Figure 1. Illustration of the model-based prognostics process

damage data that represent the system’s health state at the time the data are obtained. Since damage cannot be directly measured in most cases, a damage quantification process is required from sensor measurement data, which is called structural health monitoring (SHM). This tutorial assumes that data are available in terms of the level of damage at various times.

### 2.1. Model Definition: Battery Degradation

In the following explanation, ‘line’ or ‘lines’ in a parenthesis indicates the line number of the code in Appendix. In the degradation of a battery (line 2), it is well known that the capacity of a secondary cell such as a Lithium-ion battery degrades over cycles in use, and the failure threshold is defined when the capacity fades by 30% of the rated value (line 7). A simple form of the empirical degradation model is expressed by an exponential growth model as follows (Goebel et al., 2008):

$$\lambda = a \exp(-bt) \quad (1)$$

where  $a, b$  are model parameters,  $t$  is the time or cycles, and  $\lambda$  is internal battery performance, such as electrolyte resistance  $R_E$  or transfer resistance  $R_{CT}$ . The internal battery performance is normally observed instead of capacity. Also, there is a relationship between  $R_E + R_{CT}$  and C/1 capacity (capacity at nominally rated current of 1A);  $R_E + R_{CT}$  is typically inversely proportional to the C/1 capacity. Therefore, in this paper, the observed data are assumed to be given as a form of C/1 capacity for the purpose of demonstrating the prognostics algorithm.

The C/1 capacity data (lines 5-6) measured at every 5 weeks (lines 3-4) are given in Table 1. The data are generated by (a) assuming that the true model parameters

time step, $k$	initial, 1	2	3	4	5
weeks	0	5	10	15	20
C/1 (Ahr)	1.0000	0.9351	0.8512	0.9028	0.7754
time step, $k$	6	7	8	9	10
weeks	25	30	35	40	45
C/1 (Ahr)	0.7114	0.6830	0.6147	0.5628	0.7090

Table 1. Measurement data for battery degradation problem

$a_{\text{true}} = 1$  and  $b_{\text{true}} = 0.012$ ; (b) calculating the true C/1 capacity according to Eq. (1) for the given time steps; and (c) adding Gaussian noise  $\varepsilon \sim N(0, 0.05^2)$  to the true C/1 capacity data. The true values of parameters are only used to generate observed data. Then, the goal of prognosis is to estimate  $b$  using the data (lines 16-39)\*.

### 2.2. Estimation Algorithm: Particle Filter (PF)

PF uses a statistical method called Bayesian inference, in which observations are used to estimate and update unknown parameters as a form of the probability density function (PDF). Bayesian inference is based on the following Bayes’ theorem (Bayes, 1763):

$$p(\Theta | \mathbf{z}) \propto L(\mathbf{z} | \Theta) p(\Theta) \quad (2)$$

where  $\Theta$  is a vector of unknown parameters,  $\mathbf{z}$  is a vector of observed data,  $L(\mathbf{z} | \Theta)$  is the likelihood or the PDF value of  $\mathbf{z}$  conditional on the given  $\Theta$ ,  $p(\Theta)$  is the prior PDF of  $\Theta$ , and  $p(\Theta | \mathbf{z})$  is the posterior PDF of  $\Theta$  conditional on  $\mathbf{z}$ .

In PF, the Bayesian update is processed in a sequential way with particles (or samples) having probability information of unknown parameters; When a new measurement is available, the posterior at the previous step is used as the prior information at the current step, and the parameters are updated by multiplying it with the likelihood. Therefore, PF is also known as the sequential Monte Carlo method (Orchard & Vachtsevanos, 2007; Zio & Peloni, 2011). The general process of PF is based on the state transition function  $f$  and the measurement function  $h$  (Zio & Peloni, 2011; Li et al., 2003):

$$x_k = f(x_{k-1}, \theta_k, v_k) \quad (3)$$

$$z_k = h(x_k, \omega_k) \quad (4)$$

where  $k$  is the time step index,  $x_k$  is the damage state,  $\theta_k$  is a vector of model parameters,  $z_k$  is measurement data.  $v_k$  and  $\omega_k$  are, respectively, process and measurement noise. In this paper, the state transition function  $f$  is referred to as a damage model.

According to the damage model in Eq. (3), the battery degradation model in Eq. (1) can be rewritten in the following form (line 29):

$$x_k = \exp(-b_k \Delta t) x_{k-1} \quad (5)$$

with  $t_k = t_{k-1} + \Delta t$ . In this case, process noise  $v_k$  is ignored because it can be handled through the uncertainty in model

\* For simplicity, it is assumed that  $a = 1$  is given.

parameters. For the measurement function, it is assumed that  $z_k$  is the same as C/1 capacity including measurement noise  $\omega_k$ . Gaussian noise,  $\omega_k \sim N(0, \sigma)$ , is used with unknown standard deviation  $\sigma$ . Therefore, the unknown parameters become  $\Theta = [x, \theta (= [b]), \sigma]^T$ , including the damage state  $x_k$  which is obtained based on the model parameter  $b_k$  (see Eq. (5)) (line 8).

The process of PF is based on the Bayes' theorem illustrated in Figure 2 with one parameter estimation. At the first time step, i.e.,  $k=1$ ,  $n$  samples of the parameters are drawn from the initial (prior) distribution (lines 9, 16-21). Then, the following three steps are employed. In the first prediction step (lines 25-30), the posterior distributions of the model parameters at the previous ( $k-1$ th) step are used for the prior at the current ( $k$ th) step in the form of samples (lines 26-27). Also, the damage state at the current time is transmitted from the samples of the damage model at the previous step based on the model parameters (lines 28-29). The samples in this step correspond to  $p(\Theta_k)$  in Figure 2. Next is the updating step (lines 31-33), which is related to the likelihood of measurement data  $L(z_k | \Theta_k)$  in the figure. Assuming  $\omega_k$  is normally distributed, the likelihood of the measurement can be expressed as (line 33):

$$L(z_k | x_k^i, b_k^i, \sigma_k^i) = \frac{1}{\sqrt{2\pi}\sigma_k^i} \exp\left[-\frac{1}{2}\left(\frac{z_k - x_k^i(b_k^i)}{\sigma_k^i}\right)^2\right], \quad i = 1, \dots, n \quad (6)$$

In Eq. (6), the PDF value of  $z_k$  at the given  $i$ th samples of the unknown parameters  $\Theta = x, b, \sigma$  corresponds to the weight of the  $i$ th samples; the weight is proportional to the magnitude of the PDF value, which is expressed as the length of the vertical bar in Figure 2. Finally, the samples with high or low weight are duplicated or eliminated, respectively, at the resampling step (lines 34-39). Among

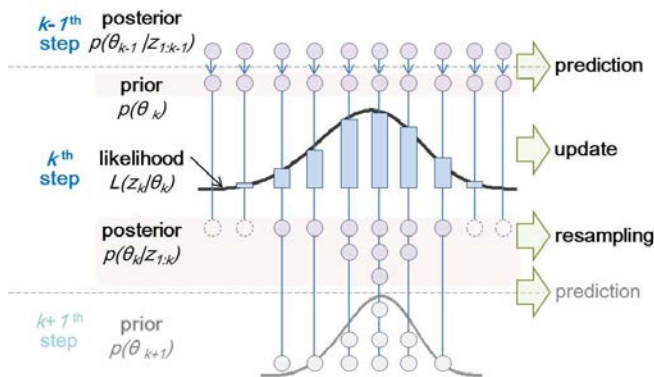


Figure 2. Illustration of the PF process

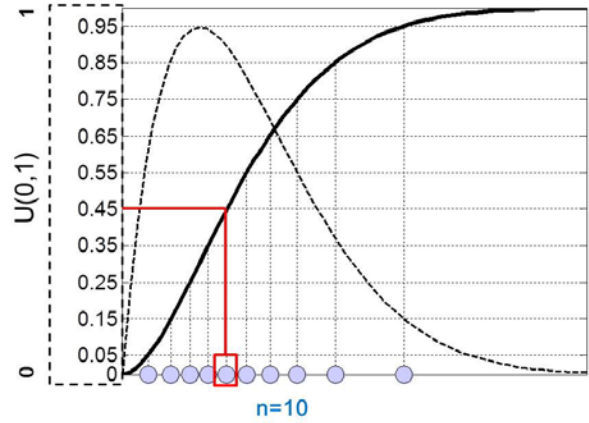


Figure 3. Illustration of resampling method

several methods, the inverse CDF method (Zio & Peloni, 2011) is used, which is illustrated in Figure 3. Firstly, a CDF is constructed from the likelihood function in Eq. (6) (line 35), which is illustrated as solid curve in the figure. Next, a random value is generated from  $U(0,1)$  (line 37), which becomes a CDF value (e.g., 0.45 in the figure). Finally, a sample of the parameter having the CDF value is found (line 38), which is marked by a rectangle in the figure. By repeating this process  $n$  times,  $n$  samples are obtained (line 36). Note that since samples exist in a discrete form, the sample having the closest value to the CDF value is selected. Consequently, the resampled results become the posterior distribution  $p(\Theta_k | z_{1:k})$  in Figure 2, which corresponds to the posterior distribution at the current step (line 38), and is also used as the prior distribution at the next ( $k+1$ th) step (lines 25-30).

### 2.3. Prognosis: Predicting the Damage State and RUL

Once the estimated parameter is obtained (lines 25-39, line 43), the future damage state and RUL can be predicted by progressing the damage state until it reaches the threshold as shown in Figure 4 (lines 24-30, 40-47). In the figure, the two dashed curves and the PDF shape, respectively,

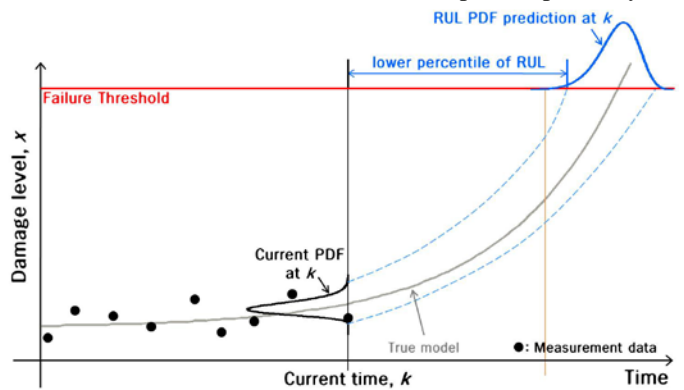


Figure 4. Illustration of RUL prediction

represent the prediction interval of the damage state and the distribution of time when the damage state reaches the threshold. The distribution of RUL can be obtained by subtracting this PDF from the threshold. In the prognosis step, only the damage state is transmitted (lines 25-30) without updating model parameters (lines 40-41). At this time, the measurement error with the updated standard deviation is added to the damage state (lines 44-46).

### 3. MATLAB IMPLEMENTATION

In this section, the usage of the 62-line Matlab code is explained. The code is divided into three parts: (1) problem definition for user-specific applications, (2) prognostics using PF, and (3) post-processing for displaying results. The block diagram of the code is illustrated in Figure 5. Only the problem definition part needs to be modified for different applications, which are further divided into two sections: parameter definition and model definition. In the parameter definition, all known parameters as well as the initial estimate of unknown parameters are defined, such as the name of parameters to be estimated, the probability parameters of initial distributions of the unknown parameters and measured data, etc. (lines 1-14). Next, the damage equation or state transition function is defined in model definition (line 29). Once these two are completed, users can obtain the RUL distribution at the current time and its percentiles, median and 90% prediction interval. Detailed explanations are given in the following subsections with an example of battery degradation, in which italicized bold letters represent the Matlab code in the Appendix.

#### 3.1. Problem Definition (Lines 1-14, 29)

##### 3.1.1. Parameter Definition (Lines 1-14)

For the battery degradation example, **'Battery'** is used for **WorkName**, which is the name of the result file. The capacity is measured at every 5 weeks, so **'weeks'** and the number **5** are, respectively, typed in **TimeUnit** and **dt**. The C/1 capacity data in Table 1 are stored in **measuData**, which

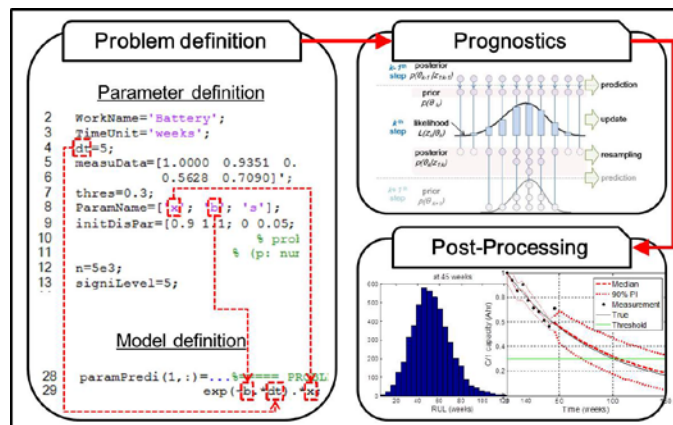


Figure 5. Block diagram of the code

is a  $k1 \times 1$  vector. According to the definition of failure threshold in Section 2.1, **0.3** (30% of C/1 capacity) is used for **thres**. **ParamName** is the name of the unknown parameters to be estimated; damage state **'x'**, model parameter **'b'** and the standard deviation of measurement error **'s'** are included. When determining the parameters' name, there are four cautions: (1) the user can define anything for the parameter's name, but the length of parameters' name should be the same as each other; (2) when assigning a one letter name, be careful not to use **i, j, k, n, p, u** because they are already used in the code; (3) the parameter's name representing the damage state and model parameters should be used as the model equation on line 29; and (4) the parameter's name of the damage state and standard deviation, respectively, should be placed on the first and the last row. **initDisPar** is a  $p \times q$  matrix of probability parameters of initial distributions, where  $p$  and  $q$  are the number of unknown parameters and probability parameters, respectively. Since there are no available prior information, it is assumed that the initial distributions of the three ( $=p$ ) unknown parameters are uniform whose probability parameters are two ( $=q$ ), lower and upper bounds:

$$x_0 \sim U(0.9, 1.1), b_0 \sim U(0, 0.05), s_0 \sim U(0.01, 0.1) \quad (7)$$

Equation (7) can be typed as **[0.9 1.1; 0 0.05; 0.01 0.1]**; in **initDisPar**. The rest of the required parameters are the number of particles (or samples) **n** and significance level **signiLevel** for calculating the confidence interval (C.I.) and prediction interval (P.I.). Usually, 1,000~5,000 particles and a 5, 2.5 or 0.5 significance level for 90%, 95% or 99% intervals are used. In this example, **5000** and **5** are set for **n** and **signiLevel**, respectively. To consider the effect according to the number of samples, users can refer to Ref. (Pitt et al., 2012).

##### 3.1.2. Model Definition (Line 29)

The damage model equation in Eq. (5) is used in line 29. In the equation, the time interval  $\Delta t$  is expressed as **dt** in the script, which was defined in line 4. Also, the model parameter  $b_k$  and the damage state at the previous step  $x_{k-1}$ , respectively, are expressed as **b** and **x**, which were defined in line 8. The algebraic expressions should use component-wise operations (i.e., using **.'**) since damage state is a vector with  $n$  samples.

#### 3.2. Prognostics using PF (Lines 15-47)

The prognostics process is composed of three steps: (1) the initial distributions of the parameters (lines 16-21), (2) estimation process (lines 25-39, 43), and (2) prognosis (lines 24-30, 40-47). In terms of the code usage, there are two issues that can be considered according to users' intention: the initial distribution (line 18) and the likelihood function



(line 33). In the code, the default options for the initial distribution and the likelihood function, respectively, are uniform and normal distribution. The other probability distributions can also be employed, and this will be introduced in Section 4.

### 3.3. Post-processing (Lines 48-62)

Once problem definition is completed and the code is implemented, distribution and its percentiles of RUL at the current time can be displayed. Figure 6 shows RUL results at 45 weeks after the updating process is progressed up to  $k=9$  (see Table 1;  $k=9$  corresponds to  $kI=10$  in the script since the initial,  $k=0$  is stored in  $kI=1$ ). Figure 6(b) shows 5, 50 (median), and 95 percentiles, which are caused by  $signiLevel=5$  (line 13). Also, the solid box in the figure represents that the results are saved as a name of 'WorkName (line 2) at current time.mat'.

The other results, such as the trace of parameters and prediction of the damage state, can be displayed by using sampling results during the updating process. Users can display the sampling results of any variable at each step by entering *ParamResul* into the Matlab command window; *xResul*, *bResul* and *sResul* are forms of adding *ParamName* (line 8) to *Resul*. Therefore, users can draw the percentiles of the damage state prediction by coding `plot(repmat(time,1,3), prctile(xResul,'perceValue'))`, and for the other cases, *xResul* is replaced with *bResul* or *sResul*. If the true values of the model parameters are

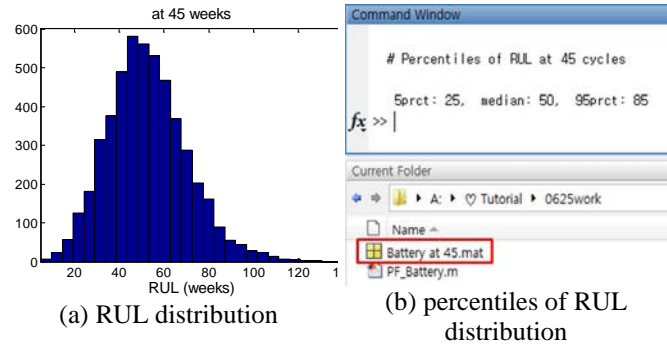


Figure 6. Visual results obtained from the code: battery degradation example

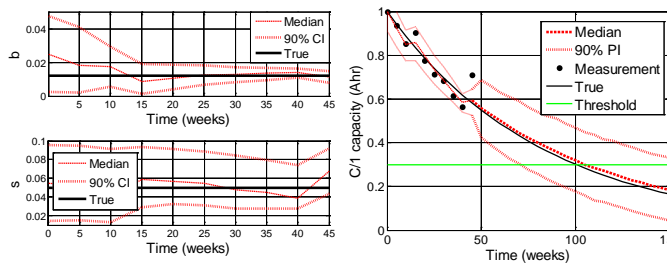


Figure 7. Visual results obtained from the additional code: battery degradation example

time step, $k$	initial, 0	1	2	3	4
time (cycles)	0	50	100	150	200
crack size (m)	0.0119	0.0103	0.0118	0.0095	0.0085
time step, $k$	5	6	7	8	9
time (cycles)	250	300	350	400	450
crack size (m)	0.0122	0.0110	0.0120	0.0113	0.0122
time step, $k$	10	11	12	13	14
time (cycles)	500	550	600	650	700
crack size (m)	0.0110	0.0124	0.0117	0.0138	0.0127
time step, $k$	15	16	17	18	19
time (cycles)	750	800	850	900	950
crack size (m)	0.0115	0.0135	0.0124	0.0141	0.0160
time step, $k$	20	21	22	23	24
time (cycles)	1000	1050	1100	1150	1200
crack size (m)	0.0157	0.0149	0.0156	0.0153	0.0155

Table 2. Measurement data for crack growth problem

known, the results can be compared with the true values. In this problem, the true values of  $b=0.012$  and  $s=0.05$ , and the true damage state are calculated using Eq. (1) or Eq. (5). The additional visual results are shown in Figure 7.

## 4. PRACTICAL USE

The code can be easily adapted by users for more practical use. As an example, the usage algorithm with a crack growth example is considered in the following subsections.

### 4.1. Model Definition: Crack Growth

It is assumed that a through-the-thickness center crack exists in an infinite plate under the mode I loading condition. The rate of damage growth can be expressed using the Paris model (Paris & Erdogan, 1963) as

$$\frac{da}{dN} = C (\Delta K)^m, \quad \Delta K = \Delta \sigma \sqrt{\pi a} \quad (8)$$

where  $a$  is the crack size,  $N$  is the number of cycles,  $m$  and  $C$  are damage model parameters,  $\Delta K$  is the range of the stress intensity factor, and  $\Delta \sigma$  is the stress range. The model can be rewritten in the form of the state transition function:

$$a_k = C_k \left( \Delta \sigma \sqrt{\pi a_{k-1}} \right)^{m_k} dN + a_{k-1} \quad (9)$$

The model parameters  $m_k$  and  $C_k$  as well as the damage state  $a_k$  are estimated using the measured crack size  $z_k$  at every 50 cycles under loading condition  $\Delta \sigma = 78 \text{ MPa}$ , which is listed in Table 2. First the true crack size data are generated using Eq. (9) with  $m_{\text{true}} = 3.8$  and  $C_{\text{true}} = 1.5 \times 10^{-10}$ . The measured crack size data are then generated by multiplying noise, which is lognormally distributed with standard deviation of  $0.001/a_k$  (m). For the

RUL calculation, the critical crack size is determined as 0.0463m. More specific information for crack growth problem is in the literature by An et al. (2012).

It is assumed that the standard deviation of measurement,  $\sigma$ , is known as 0.001m. Also, the initial distribution of the parameters and the likelihood function are, respectively, normal and lognormal distributions, which are as follows:

- initial distribution:

$$\begin{aligned} a_0 &\sim N\left(0.01, (5 \times 10^{-4})^2\right), \\ m_0 &\sim N(4, 0.2^2), \log C_0 \sim N(-22.33, 1.12^2) \end{aligned} \quad (10)$$

- likelihood function:

$$\begin{aligned} L(z_k | a_k^i, m_k^i, C_k^i) \\ = \frac{1}{z_k \sqrt{2\pi} \zeta_k^i} \exp\left[-\frac{1}{2} \left(\frac{\ln z_k - \lambda_k^i}{\zeta_k^i}\right)^2\right], \quad i = 1, \dots, n \end{aligned} \quad (11)$$

where  $\zeta_k^i = \sqrt{\ln\left[1 + \left(\frac{\sigma}{a_k^i(m_k^i, C_k^i)}\right)^2\right]}$  and

$$\lambda_k^i = \ln\left[a_k^i(m_k^i, C_k^i)\right] - \frac{1}{2}(\zeta_k^i)^2.$$

## 4.2. Modifying the Code for the Crack Growth Problem

### 4.2.1. Problem Definition

Based on the above given information, the code in the Appendix is changed as follows:

- line 2: **WorkName='Crack';**
- line 3: **TimeUnit='cycles';**
- line 4: **dt=50;** (or **dN=50**, but should be matched with line 29)
- lines 5-6: **measuData=[0.0119 0.0103 0.0118 0.0095 0.0085 0.0122 0.0110 0.0120 0.0113 0.0122 0.0110 0.0124 0.0117 0.0138 0.0127 0.0115 0.0135 0.0124 0.0141 0.0160 0.0157 0.0149 0.0156 0.0153 0.0155]';**
- line 7: **thres=0.0463;**
- line 8: **ParamName=['a'; 'm'; 'C'; 's'];**
- line 9: **initDisPar=[0.01 5e-4; 4 0.2; -22.33 1.12; 0.001 0];** This corresponds to Eq. (10) except the last two values, **0.001** and **0** for  $\sigma$  ( $=s'$ ). Even if  $\sigma$  is a deterministic value, it should be included in lines 8-9. Therefore, users should make  $\sigma$  become a deterministic value (0.001m) by using **0.001** and **0**, which stand for mean and standard deviation, respectively. In other

words, the probability parameters should be set to make the  $n$  samples become the same as a deterministic value.

- line 13: **signiLevel=2.5;** 95% intervals are calculated.
- insert it next line 13: **delSigma=78;**
- line 29: **exp(C).\*(delSigma.\*sqrt(pi\*a)).^m.\*dt+a;** This corresponds to Eq. (9), but note that  $\log(C)$  is used instead of  $C$  due to a numerical problem ( $C$  is a very small value).

### 4.2.2. Prognostics using PF

The initial distribution of the parameters and the likelihood function are different from those of battery degradation. Therefore, the lines 18, 33 and 45 should be modified as follows:

- line 18: **param(j,:)=normrnd(initDisPar(j,1),initDisPar(j,2),1,n);**
- line 33: **sigl=sqrt(log(1+(paramPredi(end,:)./paramPredi(1,:).^2))); mul=log(paramPredi(1,:))-0.5\*sigl.^2; likel=lognpdf(measuData(k),mul,sigl);**
- line 45: **sigl=sqrt(log(1+(param(end,:)./param(1,:).^2))); mul=log(param(1,:))-0.5\*sigl.^2; eval([ParamResul(1,:)'(k,:)=lognrnd(mul,sigl,1,n)']);**

If the prior information and the distribution type of measurement error are not given, the initial distribution and the likelihood function should be assumed. It would be a

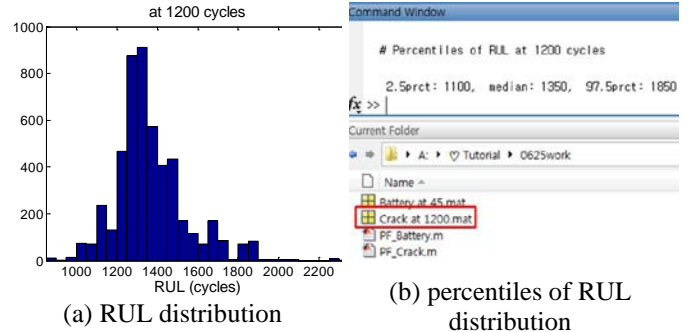


Figure 8. Visual results obtained from the code: crack growth example

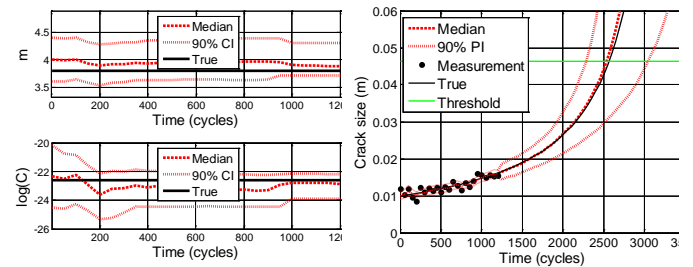


Figure 9. Visual results obtained from the additional code: crack growth example

good exercise to study different distribution types based on the above revision.

#### 4.2.3. Post-Processing

The results obtained from the code are shown in Figure 8. Also, users can obtain Figure 9 using the stored results of the parameters; *aResul*, *mResul*, *CResul* which can be known by typing *ParamResul* in the command window.

#### 5. CONCLUSION

This paper presents a tutorial for model-based prognostics with a Matlab code. The code is simply constructed with 62 lines using an example of battery degradation, and users can easily modify the code for their specific applications. Also, more practical cases are considered with a crack growth example. This will be helpful for the beginners in prognostics to use the prognostics method for their applications.

#### ACKNOWLEDGEMENT

This work was supported by the International Cooperation of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea government Ministry of Knowledge Economy (No. 20118520020010).

#### REFERENCES

- An, D., Choi, J. H., Schmitz, T. L., & Kim, N. H., (2011). In-Situ Monitoring and Prediction of Progressive Joint Wear using Bayesian Statistics, *Wear*, vol. 270(11-12), pp. 828-838.
- An, D., Choi, J. H., & Kim, N. H., (2012). Identification of Correlated Damage Parameters under Noise and Bias Using Bayesian Inference. *Structural Health Monitoring*, vol. 11(3), pp. 292-302.
- Bayes, T., (1763). An Essay towards solving a problem in the doctrine of chances, *Philosophical Transactions of the Royal Society of London*, vol. 53, pp. 370-418.
- Daigle, M., & Goebel, K., (2011). Multiple Damage Progression Paths in Model-based Prognostics. *Aerospace Conference, 2011 IEEE*.
- DeCastro, J. A., Tang, L., Loparo, K. A., Goebel, K., and Vachtsevanos, G., (2009). Exact Nonlinear Filtering and Prediction in Process Model-based Prognostics. *Annual Conference of the Prognostics and Health Management Society*.
- Goebel, K., Saha, B., Saxena, A., Celaya, J. R., & Christophersen, J., (2008). Prognostics in Battery Health Management. *IEEE Instrumentation and Measurements Magazine*, vol. 11(4), pp. 33-40.
- Kalman, R. E., (1960). A New Approach to Linear Filtering and Prediction Problems. *Transaction of the ASME—Journal of Basic Engineering*, vol. 82(1), pp. 35-45.
- Luo, J., Pattipati, K.R., Qiao, L., & Chigusa, S., (2008). Model-based Prognostic Techniques Applied to a

- Suspension System. *IEEE Transactions on System, Man and Cybernetics*, vol. 38(5), pp. 1156-1168.
- Orchard, M. E., & Vachtsevanos, G. J., (2007). A Particle Filtering Approach for On-Line Failure Prognosis in a Planetary Carrier Plate. *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 7(4), pp. 221-227.
- Paris, P. C. & Erdogan, F., (1963). A Critical Analysis of Crack Propagation Laws, *ASME Journal of Basic Engineering*, vol. 85, pp. 528-534.
- Payne, S. J., (2005). A Bayesian Approach for the Estimation of Model Parameters from Noisy Data Sets. *IEEE Signal Processing Letters*, vol. 12(8), pp. 553-556.
- Zio, E., & Peloni, G., (2011). Particle Filtering Prognostic Estimation of the Remaining Useful Life of Nonlinear Components. *Reliability Engineering and System Safety*, vol. 96(3), pp. 403-409.
- Li, P., Goodall, R. & Kadiramanathan, V., (2003). Parameter Estimation of Railway Vehicle Dynamic Model using Rao-Blackwellised Particle Filter, *European Control Conference*.
- Pitt, M. K., Silva, R. S., Giordani, P., & Kohn, R., (2012). On Some Properties of Markov Chain Monte Carlo Simulation Methods based on the Particle Filter. *Journal of Econometrics*, In Press (available online, July 2012).

#### BIOGRAPHIES

**Dawn An** received the B.S. degree and M.S. degree of mechanical engineering from Korea Aerospace University in 2008 and 2010, respectively. She is now a joint Ph.D. student at Korea Aerospace University and the University of Florida. Her current research is focused on the Bayesian inference, correlated parameter identification and the methodology for prognostics and health management and structural health monitoring.

**Joo-Ho Choi** received the B.S. degree of mechanical engineering from Hanyang University in 1981, the M.S. degree and Ph.D. degree of mechanical engineering from Korea Advanced Institute of Science and Technology (KAIST) in 1983 and 1987, respectively. During the year 1988, he worked as a Postdoctoral Fellow at the University of Iowa. He joined the School of Aerospace and Mechanical Engineering at Korea Aerospace University, Korea, in 1997 and is now Professor. His current research is focused on the reliability analysis, design for life-time reliability, and prognostics and health management.

**Nam Ho Kim** received the B.S. degree of mechanical engineering from Seoul National University in 1989, the M.S. degree and Ph.D. degree of mechanical engineering from Korea Advanced Institute of Science and Technology (KAIST) in 1991 and the University of Iowa in 1999, respectively. He worked as a Postdoctoral Associate at the University of Iowa from 1999 to 2001. He joined the Dept. of Mechanical & Aerospace Engineering at the University

of Florida, in 2002 and is now Associate Professor. His current research is focused on design under uncertainty, design optimization of automotive NVH problem, shape

DSA of transient dynamics (implicit/explicit) and structural health monitoring.

#### APPENDIX: MATLAB CODE

```

1 %===== PROBLEM DEFINITION: PARAMETER DEFINITION =====
2 WorkName='Battery'; % work results are saved by WorkName
3 TimeUnit='weeks'; % time unit name
4 dt=5; % time interval (five weeks)
5 measuData=[1.0000 0.9351 0.8512 0.9028 0.7754 0.7114 0.6830 0.6147 ...
6 0.5628 0.7090]'; % measured data at every time intervals (k1 x 1)
7 thres=0.3; % threshold - critical value
8 ParamName=['x'; 'b'; 's']; % model parameters' name to be estimated
9 initDisPar=[0.9 1.1; 0 0.05; 0.01 0.1];
10 % probability parameters of initial distribution, p x q
11 % (p: num. of unknown param, q: num. of probability param)
12 n=5e3; % number of particles
13 signiLevel=5; % significance level for C.I. and P.I.
14 %=====
15 % % % PROGNOSTICS using PARTICLE FILTER
16 p=size(ParamName,1);
17 for j=1:p; % Initial Distribution
18 param(j,:)=unifrnd(initDisPar(j,1),initDisPar(j,2),1,n);
19 ParamResul(j,:)=[ParamName(j,:) 'Resul'];
20 eval([ParamResul(j,:) '=param(j,:);']);
21 end;
22 k1=length(measuData); k=1; % Update Process or Prognosis
23 if measuData(end)-measuData(1)<0; cofec=-1; else cofec =1; end
24 while min(eval([ParamResul(1,:) '(k,:)']))*cofec<thres*cofec; k=k+1;
25 % step1. prediction (prior)
26 paramPredi=param;
27 for j=1:p; eval([ParamName(j,:) '=paramPredi(j,:);']); end
28 paramPredi(1,:)=...%===== PROBLEM DEFINITION: MODEL DEFINITION =====
29 exp(-b.*dt).*x;
30 %=====
31 if k<=k1 % (Update Process)
32 % step2. update (likelihood)
33 likel=normpdf(measuData(k),paramPredi(1,:),paramPredi(end,:));
34 % step3. resampling
35 for i=1:n; cdf(i)=sum(likel(1:i)); end; cdf=cdf./max(cdf);
36 for i=1:n;
37 u=rand;
38 loca=find(cdf >= u); param(:,i)=paramPredi(:,loca(1));
39 end;
40 else % (Prognosis)
41 param=paramPredi;
42 end
43 for j=1:p; eval([ParamResul(j,:) '(k,:)=param(j,:);']); end;
44 if k>k1;
45 eval([ParamResul(1,:) '(k,:)=normrnd(param(1,:),param(end,:));']);
46 end
47 end
48 % % % POST-PROCESSING
49 time=[0:dt:dt*(k-1)]'; % RUL Calculation
50 perceValue=[50 signiLevel 100-signiLevel];

```

```

51 for i=1:n;
52     loca=find(eval([ParamResul(1,:) '(:,i)'])*cofec>=thres*cofec);
53     RUL(i)=time(loca(1))-time(k1);
54 end;
55 RULPerce=prctile(RUL',perceValue);
56 figure; set(gca,'fontsize',14); hist(RUL,30);           %% RUL Results Display
57 xlim([min(RUL) max(RUL)]); xlabel(['RUL' ' (' TimeUnit ')']);
58 titleName=['at ' num2str(time(k1)) ' ' TimeUnit]; title(titleName)
59 fprintf( '\n # Percentiles of RUL at %g cycles \n', time(k1))
60 fprintf('\n %gprct: %g, median: %g, %gprct: %g \n' , perceValue(2), ...
61         RULPerce(2), RULPerce(1), perceValue(3), RULPerce(3))
62 Name=[WorkName ' at ' num2str(time(k1)) '.mat']; save(Name);           %% Work Save

```

# A Design Methodology of Optimized Diagnosis Functions for High Lift Actuation Systems

Christian Modest<sup>1</sup>, Frank Thielecke<sup>2</sup>

<sup>1,2</sup> *Institute of Aircraft Systems Engineering, Hamburg University of Technology, Germany*  
*christian.modest@tuhh.de*  
*frank.thielecke@tuhh.de*

## ABSTRACT

This paper presents a model-based approach to the optimal design of diagnosis system architectures for complex high lift actuation systems. The overall approach consists of two steps. In the first step, safety and reliability related requirements are introduced. These focus on the detectability and isolability of faults. Symptoms are used therefore. These are separated into safety and reliability related symptoms. In the second step, different alternatives to provide the symptoms are drawn and evaluated in order to gain an optimal design solution. A two stage analysis process is used therefore. The first stage focuses on the fulfillment of the safety related requirements whereas the second stage concentrates on the reliability related requirements. All aspects of the analysis are depicted exemplary and formalized theoretically. The results of the application to the high lift actuation system of an Airbus A340-600 aircraft are presented afterwards and discussed in the end.

## 1. INTRODUCTION

The members of the air transport system compete in a global, steadily growing market. Airlines, manufacturers and maintenance, repair, overhaul (MRO) providers are continuously forced to strengthen their competitive edges and improve business in order to stay profitable. One point they all have in common is to provide a safe operation and a high availability of the aircraft at a minimum of total cost. The development and use of efficient health management technologies can thus be seen as an attribute for the diversification and consolidation of the own position inside a global competitive environment.

New technologies tend to a strong increase in complexity in order to meet diversified customer and environmental requirements. One aspect of current research is to raise the function-

ality of high lift systems on commercial airplanes in order to improve the overall aircraft performance. A decoupling of surfaces and the use of decentralized drive units state possible concepts. While developing adequate solutions, it has to be considered that new technologies need not only to improve functional aspects of the aircraft but have also to be operated and maintained in a capable manner in order to reap all benefits. The consideration of efficient health management technologies is thus indispensable.

The focus of this paper is on a model-based approach to the optimal design of diagnosis system architectures for complex high lift actuation systems (HLS). Section 2 gives an overview of current HLS and their essential components. General challenges in developing adequate diagnosis systems (DS) and a systematic, model-based approach for the design and test of DS are depicted in Section 3. The first two steps are focused in this paper. These concern the definition of requirements and the conceptual design phase. The importance of these steps and a concept for a systematic, requirement based design procedure are depicted in Section 4.

The development of a diagnosis model is presented in Section 4.1. The identification and formalization of requirements is depicted afterwards. Section 4.2 demonstrates how safety and reliability related requirements are assigned that the design of the DS has to meet. This concerns the detectability and isolability of faults by means of symptoms. In order to identify optimal symptoms, with respect to various objectives, different alternatives for respective monitoring and sensor devices are drawn and implemented in a diagnosis model. A simulation of fault modes is then carried out and cause-effect relationships are gained. These are evaluated in a two stage process that is depicted in Section 4.3. The first stage describes how minimal architectures, that are adequate to fulfill the safety requirements are gained. Afterwards it is addressed in the second stage how additional monitoring devices are chosen in order to meet also reliability related requirements. Both steps are depicted by means of examples and specified theoretically. The results of the application to

---

Christian Modest et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

the HLS of an Airbus A340-600 aircraft are presented in Section 5. A discussion of related work is given in Section 6. In the end, Section 7 concludes and gives an outlook about open points and future research activities.

## 2. FUNDAMENTALS OF HIGH LIFT SYSTEMS

Commercial airplanes are equipped with high lift devices to augment lift at low speed during takeoff and landing. Today, those systems primarily consist of a mechanical transmission shaft system that transmits mechanical power from a centralized hydraulic power control unit (PCU) to rotary actuators at each wing half. The actuators are located alongside the transmission system and deploy the high lift surfaces synchronously.

Figure 1 depicts a typical high lift actuation system at the trailing edge of a commercial aircraft. This flap system consists of a mechanical shaft system that is powered by a PCU. The mechanical power is transmitted to five down drive stations on each half of the wing via shafts, joints and gearboxes. Each of the drive stations consists of a down-drive gearbox and shaft, an input gear box with torque limiter, a cross shaft and rotary actuator. The inner surface has two drive stations whereas the outer surface is moved by three.

In case of safety-critical failures a wing tip brake (WTB) can hold the system and inhibit movement. In order to monitor for such conditions and control the overall system two Slat-Flap Control Computers (SFCC) are used. The monitoring and control is done using information from discrete and analogue sensors like position pick off units (PPU).

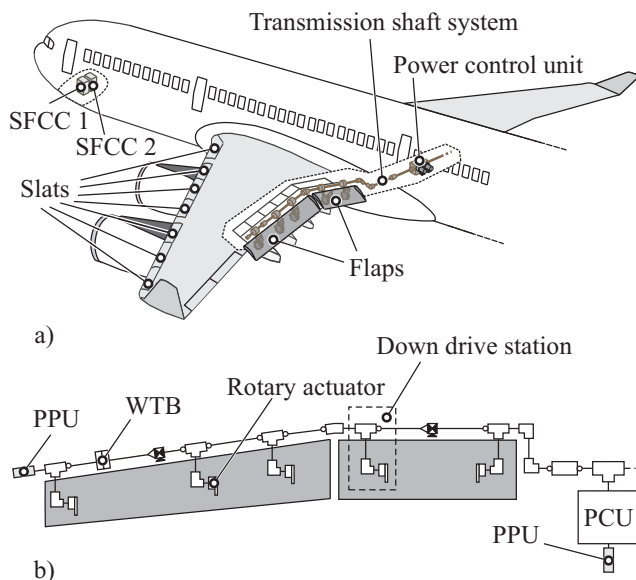


Figure 1. Typical high lift actuation system for a commercial aircraft.

## 3. CHALLENGES AND MODEL-BASED APPROACH

The task of a diagnosis system consists in the detection of abnormal functional conditions (AFC) and the isolation of potential root causes. Based on the criticality of the specific AFC and the underlying fault a decision is made afterwards. This decision can result in an abnormal shut down of the overall system or just in an indication for maintenance. In order to detect and isolate the AFCs and take adequate measures a process chain is used, that consists of different steps. Figure 2 depicts an overview of this chain. In the first step specific features are extracted from measurements on a system level. For the HLS a feature can be a too high position difference between the output of the PCU and one of the PPU at the transmission ends. The logical combination of different features then leads to the detection of the symptom of an AFC. In case of a high criticality an abnormal shut down of the PCU and the setting of the WTBs results as an action, whereas in other cases a degraded operation is still possible.

The development of the previously mentioned diagnosis functions for HLS is done today primarily empirically or as an after-thought due to in-flight incidences. Considering the complexity of new HLS as presented in (Lulla, 2011) and (Recksiek, 2009) the empiric approach results in a costly and laborious process. Furthermore, gaining optimal functions while considering different design objectives is hardly possible. Thus, the development of diagnosis functions has to be dealt with systematically and traceable already during the HLS development process. A model-based approach for the design and verification of a DS for HLS is proposed therefore and introduced in the following.

The proposed approach is embedded into the general aircraft systems development process. This process is typically divided into several steps that are arranged in the common V-model (Haskins, 2006). The left branch of this model designates the overall system design whereas the right branch marks the overall system test.

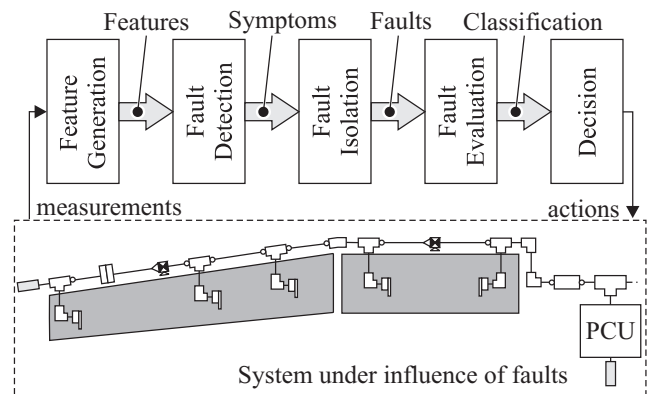


Figure 2. Diagnosis functions for a high lift actuation system.

In the first step of the design phase the system requirements are developed, based on the aircraft level requirements. In order to realize these requirements, the system architecture and all involved components are designed. At the bottom of the model the components are built as hardware. On the right branch tests are performed to verify that the design meets the requirements. This is done first separately for the components and afterwards in an integrated environment for the complete system. If requirements are not met or failures are identified during the test procedures a correction of requirements or re-design could be necessary.

Figure 3 depicts an approach to integrate the development of a diagnosis system into the V-model.

The design and test of the DS should begin early and in parallel to the overall system development process. This should be done in order to avoid failures and unnecessary elements in the functional specifications. The DS should be designed strictly according to requirements. These can be safety, reliability and performance related. For this task models of the different components are used as executable units to validate each design step. At the bottom of the V-model executable code is generated. This can be C-code for monitoring devices and xml-data files for built in test (BIT) specifications.

The model-based test is used for the verification of single and combined applications with respect to the functional requirements. In the first step of the test procedure this is done separately for the components. Examples for this step are the test of stimulated monitoring devices or the test of a knowledge base to verify that BIT requirements are met. In order to test the complete DS in interaction with the physical system a virtual integration approach is used. For this step, models that have already been used for the design and optimization of the system architecture and specific components, are extended in order to be usable for simulations together with the complete software system. This includes the DS and all the control applications.

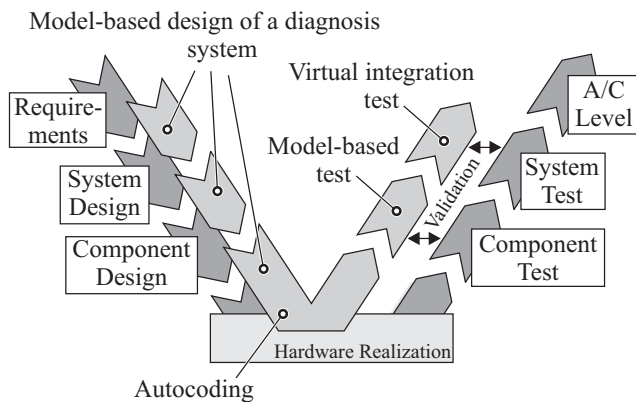


Figure 3. Framework of a model-based development process for diagnosis systems.

The approach of an integrated simulation environment enables the identification of failures, which are caused by the physical interaction of the system components combined with all the software systems.

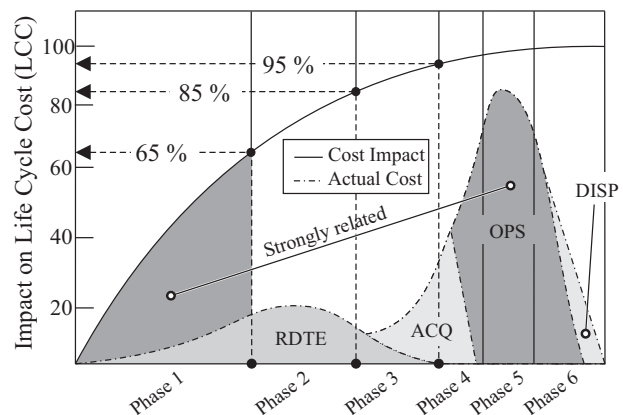
Following the approach for the design and test of the DS a simulation of the complete system is possible before qualified code is generated and implemented on final hardware platforms and further hardware devices are chosen. Thereby an early identification and correction of failures in the functional specifications can be made that should lead to a significant reduction of the overall development costs.

This paper deals with the optimal design of diagnosis system architectures for complex high lift actuation systems. The basic aspects of the integration of all the involved DS instances and the execution of preliminary verification tests are presented in (Modest, Schories, et al., 2011) and (Modest, Grymlas, et al., 2011) for the application to multi functional fuel cell systems.

#### 4. DESIGN OF A DIAGNOSIS SYSTEM ARCHITECTURE

The first phases of the overall system development process have an important impact on the product's total life cycle cost (LCC). The LCC includes cost for research, development, test and evaluation (RDTE), acquisition (ACQ), operation and support (OPS), and the final disposal (DISP).

Figure 4 depicts the impact of the different aircraft program phases on the LCC and illustrates where the actual costs occur (Roskam, 2006).



- Phase 1** : Planning and Conceptual Design
- Phase 2** : Preliminary Design and System Integration
- Phase 3** : Detail Design and Development
- Phase 4** : Manufacturing and Acquisition
- Phase 5** : Operation and Support
- Phase 6** : Disposal

Figure 4. Impact of different aircraft program phases on the life cycle cost.



The biggest part of the LCC occurs during the aircraft operation and support whereas the cost for RDTE are comparatively small. The impact of RDTE in general and the planning and conceptual design phase in particular on the cost for OPS are huge though. The first program phase accounts for an impact of 65% on the LCC. This phase has thus to be dealt with in a careful and systematic way in order to prevent high cost that may result from belated, but necessary changes in the aircraft system and the respective DS due to in-flight incidences. Examples are, that additional sensors are needed in order to detect certain failure conditions or to support the system's troubleshooting for the case that faults lead to high system downtimes. In order to prevent such conditions a systematic and traceable design procedure for a DS is necessary.

The main tasks of the DS are the detection of all abnormal functional conditions, which are here related to their safety impact, and the provision of distinct information for an efficient troubleshooting. A general design procedure for a DS that assures that these requirements can be met during the operation is presented in the following sections. The focus is on the conceptual design of a DS architecture.

Figure 5 depicts a general overview of the proposed design procedure. In the first step, requirements that the design has to meet are analyzed and defined. At the current stage these are related to safety and reliability. The first one focuses on the detection of symptoms of safety critical failure conditions whereas the latter one concentrates on the fusion of symptoms in order to identify root causes. An important aspect is thus the definition of safety critical failure conditions on a system level. As a guideline, the SAE ARP 4761(SAE, 1996) is used.

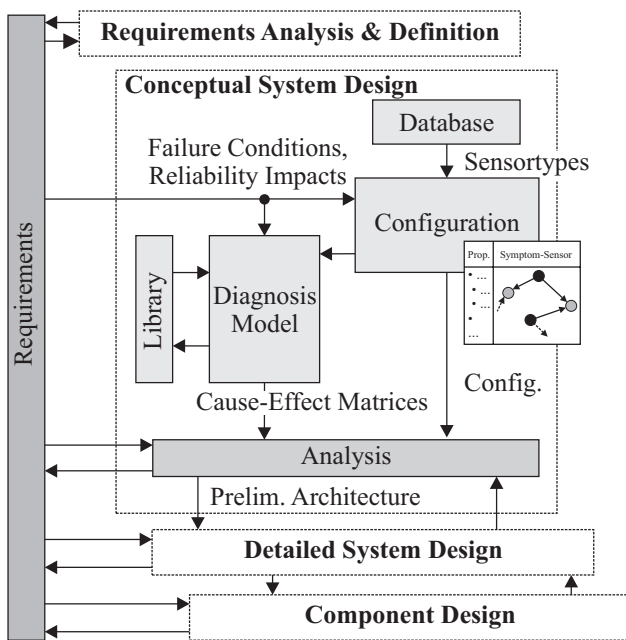


Figure 5. Overview of the design phase of a DS for HLS.

Table 1 depicts three examples of failure conditions on a system level of a typical high lift actuation system. These conditions are numbered and sorted according to their severity. The classification *MIN* stands for minor and *CAT* for catastrophic. In the first case a degraded operation of the system might still be possible whereas in the latter case an abnormal shut down of the overall system has to be commanded. In order to enable the appropriate system reaction these two conditions have to be separated. A prerequisite is, that in the general case the symptoms of all FCs are analyzed properly and adequate monitoring devices for the detection are chosen.

This paper focuses on the optimal design of diagnosis systems. In order to gain these optimal design solutions with respect to various objectives, different alternatives to provide features and detect symptoms for each failure condition have to be considered. In order to clarify this point the FC *Asymmetric Flap Movement* is used as an example. The effect of such a failure condition might be an uncontrollable roll moment on an aircraft level which in turn can lead to a total loss of the aircraft. In order to identify this condition and take adequate measures it has to be analyzed in the first step how respective symptoms look like. One such symptom might be a position difference that exceeds certain limits. This difference can be taken between the left and right hand transmission system or separately between each of the transmission halves and the PCU. A third alternative might consist in taking features from position measurements at each pair of rotary actuators. It is obvious that all the alternatives differ in various objectives. While the first one uses only a minimum of information the other alternatives would need more features and by that cause more effort. However, this simple evaluation holds only for one objective and in general every alternative has to be evaluated in the overall design context. This means with respect to reliability related requirements and other requirements that might be defined in further design steps.

The way, that has been depicted for one failure condition, is now repeated carefully for all the failure conditions that have to be considered. A general overview of the manner in which failure conditions are linked to monitoring devices that again are linked to sensor devices is depicted in Figure 6.

FC Ref.	Title	Class.
001	Flaps operate with reduced rate	MIN
...		
004	Loss of Flap Operation	MIN
...		
011	Asymmetric Flap Movement	CAT
...		

Table 1. Failure condition summary list.

In general, the presented approach leads to design solutions that can effectively fulfill the safety requirement with respect to various criteria. However, the most optimal solution may not be adequate to also fulfill other requirements like the reliability related one. Considering the previous example, this means that only some of the alternatives would allow to isolate potential root causes on different levels of detail. While the first alternative does not allow for any isolation, the second and third one would allow for an isolation on a system or subsystem level. Consequently, the first alternative would have to be extended in order to meet further requirements. Therefore, maintenance conditions (MC) are introduced next.

The MCs are defined to be conditions that are not directly linked to specific FCs. Examples are the conditions "high position difference on right hand system side" or "low load at left hand PCU side". Both the conditions state possible extensions of the first alternative of the previous example and might be used to meet certain reliability related requirements, like an isolation between left and right hand system side. In general, features that are needed to detect symptoms of MCs can be gained from measurements that are also linked to FCs.

According to the way proposed before, a set of alternatives for the detection of symptoms and the respective sensors for FCs and MCs are defined. This set is then implemented in a diagnosis model. A simulation of all component fault modes is carried out and the respective symptoms in form of monitor flags are stored and transferred into cause-effect matrices. An example is depicted in Figure 7.

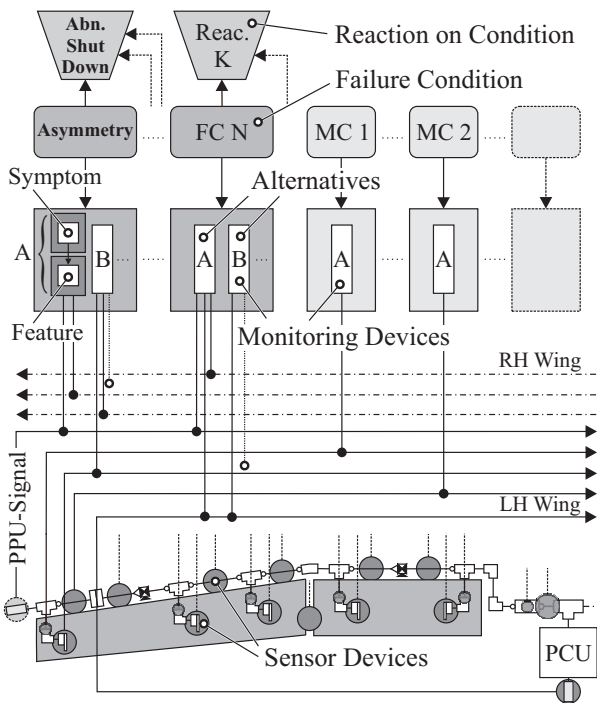


Figure 6. Placement of sensors and concepts for monitoring.

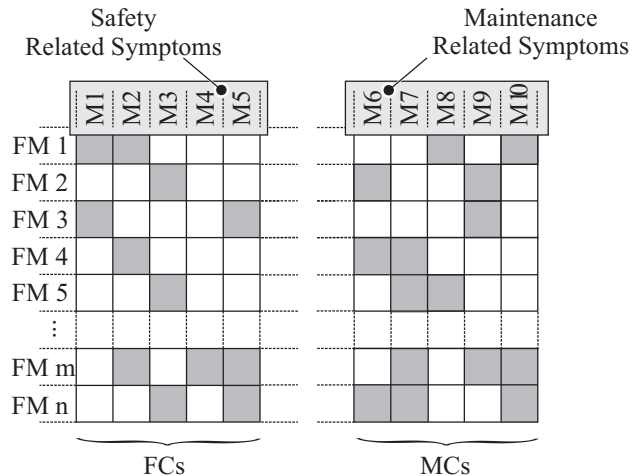


Figure 7. Result of the simulation of dedicated fault modes.

The cause-effect matrices are used as an input for a two stage analysis process. In the first stage, the safety related symptoms are evaluated in order to gain solutions for the fulfillment of a safety requirement. In the second stage, the fulfillment of additional reliability related requirements is checked and if necessary complementary maintenance related symptoms are identified for an extended design solution. In order to identify the overall optimal solution, cost factors for the placement of sensors are considered. The relation between symptoms and sensors is defined as a directed graph, where each sensor node has a cost property. In the end of the overall design procedure a globally optimal solution is gained, that consists in the definition of symptoms, features and measurements.

The entire proceeding is explained in the following. In the first part the development of a diagnosis model is shown. Afterwards, a theoretical definition of the requirements is given and in the third part the analysis process is depicted.

#### 4.1. Development of a Diagnosis Model

The proposed approach uses on an a-causal, component-based, quasi-static model of the high lift actuation system. As a simulation environment the tool RODON is used (Bunus et al., 2009). The overall model is built-up of different layers where each of the layers states one level of hierarchy. This approach is used to effectively handle changes that will naturally occur during the overall system development process and especially during the early phases.

The model's top-layer presents the system level where the interaction between all involved subsystems and the interfaces to other systems, e.g. the electrical power supply, are defined. Subsystems of a typical HLS are the PCU, the right and left hand transmission system and two SFCCs. Each of these subsystems is built-up of components that are defined at

the component level. In case of the SFCC these components are a control and a monitoring part whereas for the transmission system these are shafts, gears and actuators amongst others. Figure 8 depicts an excerpt of the component level of the left-hand transmission system. It is shown the inner flap surface and dedicated mechanical components. The surface is deployed by means of two rotary actuators that in turn are operated by several shafts and gears.

The lowest layer of the model is the constraint level where the specific behavior is defined. A multi-step modeling approach is used therefore. Aspects of that approach have been introduced in (Modest, Schories, et al., 2011). In order to illustrate this approach the example of a gear is used. This gear is located on the lowest component level of the left hand transmission system that has been depicted previously. Figure 9 states the exact position in the second down drive of the inner flap surface. The purpose of the gear is to reduce the transmission's rotational speed. In order to build a model of this component all the interfaces are defined in the first step. A specific domain is used for that. For mechanical components the domain consists of the rotational speed as a potential quantity and the torque as a flow quantity. The gear has one input and one output so that two interfaces are needed. As a result there is a model-shell.

In the second step all parameters are added to the model-shell. In case of the gear these can be the gear ratio, the efficiency factor and the diameters of all cog wheels, depending on the level of detail. Afterwards in the third step all the variables are defined. These can be the rotational speed difference or the power dissipation amongst others. All the variables and the parameters are linked in the fourth step. As it can be seen from Figure 9 the behavioral section is divided into two main parts with two sets of constraints. The first part contains the set of constraints that are supposed to be fix.

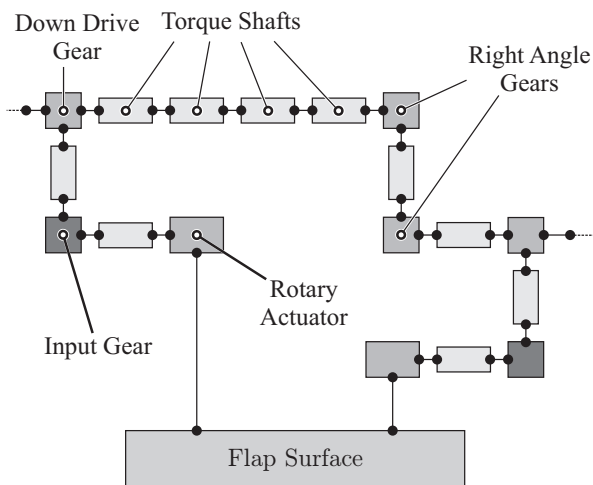


Figure 8. Excerpt from the component level of the transmission model.

The second part contains the set of constraints that are defined to be switchable. An example for a constraint that is supposed to be fix is the rotational speed difference. This is defined as the speed difference  $\Delta n$  between both the speeds  $n_1$  and  $n_2$  at the interfaces under consideration of the gear ratio  $i$ :  $\Delta n = n_1 - n_2 \cdot i$ . The value for the speed difference though is defined as a switchable constraint. In the example there are two of these constraints which relate to two specific fault modes. In general these include the nominal mode. Following the example the speed difference is defined to be zero,  $\Delta n = 0$ , in the nominal mode whereas in the second mode it is undefined which relates to a rupture. For the torque  $M_1$  and  $M_2$  it holds  $M_1 \cdot i + M_2 = 0$  in the nominal and the second mode, whereas in the second mode  $M_1$  is set as a parameter,  $M_1 = 0$ , so that there is zero torque at both the interfaces. A third mode that is not part of the example is a jam of the gear. The way in which this mode is defined is analog to the example.

According to the way, that has been depicted, all the components of a typical high lift actuation system are modeled. This includes mechanical, hydraulic and electrical components as well as the controller and monitoring part of the software systems. The focus of modeling fault modes though lies on the non software systems. All components are then grouped in a library and used to form a model of the specific system under consideration. In this paper it is the Airbus A340-600 flap actuation system. The architecture is similar to the one that has been depicted in Figure 1.

The validation of the behavior in normal operation according to functional requirements has been done but is not the scope of this paper.

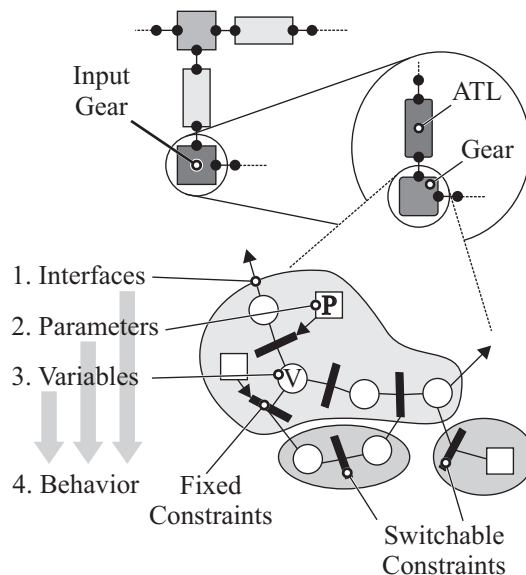


Figure 9. A multi-step modeling approach for a component model applied to a gear.

## 4.2. Definition of requirements

The planning and conceptual design phase for the development of a DS focuses on the analysis of optimal system architectures with respect to the detection of symptoms of safety critical failure conditions (FC) and the isolation of potential root causes. In the following, two requirements for both aspects are introduced in detail. The first one is related to the symptom-detection of FCs and has previously been mentioned as safety related whereas the second one focuses on the isolation task and has previously been mentioned as reliability related. According to the general proceeding of detection followed by isolation both requirements are introduced consecutively in the following.

In the first step the diagnosis system is in charge of the detection of symptoms of FCs. These are defined in accordance with (Isermann, 2006). Related to that, a safety requirement is defined here, such that all fault modes that map to symptoms of FCs shall be detected using a minimal amount of symptoms with a minimal overlap. This is formalized in the following definition.

**Definition 4.1 (Safety Requirement)** *A safety requirement (SR) is defined such that a set of fault modes that map to symptoms of FCs are detectable with a minimal amount of symptoms that have a minimal overlap:*

$$SR := \{fm \mid \exists m \in M : fm \mapsto m \wedge \min(\cap m_i) \forall m_i \in M^*\} .$$

The following holds in the definition:

- $m$  is a symptom that is sensitive to a specific fault mode ,
- $M$  is a set of symptoms that are sensitive to any fault mode ,
- $M^*$  is a minimal set of symptoms that have a minimal overlap .

The safety requirement states that all fault modes  $fm$  that are related to  $M$  shall be detected using  $M^*$ . An example for that is the consideration of two different fault modes. An analysis showed that fault mode one maps to the symptoms A and B whereas fault mode two maps to the symptoms B and C. The safety requirement is fulfilled by two solutions. The first one consists of the symptom B whereas the second one consists of the symptoms A and C. Both solutions use a minimal amount of symptoms to detect all relevant fault modes. The capability to infer the root cause is different though.

In the second step the diagnosis system is in charge of fusing different symptoms to isolate potential root causes. Compared to the definition of the SR it is not required here to isolate between every considered fault mode but according to specified isolability requirements. In order to clearly define this concept some definitions are made in advance. The first one is the definition of isolability items. These can focus on specific fault modes, components and system parts.

**Definition 4.2 (Isolability Item)** *An isolability item ( $ii$ ) is a triple ( $sp, comp, fm$ ) where:*

1.  $sp$ , a system part, is an expression that indicates a location ,
2.  $comp$ , a component, is an expression that indicates a component ,
3.  $fm$ , a fault mode, is an expression that indicates a fault mode .

The set of all isolability items  $\cup \{ii_j\}$  is defined to be  $II$ . An example for an isolability item is the triple  $LH \wedge Shaft001 \wedge Rupture$ . In this expression  $LH$  stands for the flap system's left hand side,  $Shaft001$  stands for a mechanical component of the transmission system and  $Rupture$  is a specific fault mode. If it is unambiguous, parts of the formal  $ii$  can be omitted if necessary in order to define larger items that cover complete locations or components. An example is the item  $PCU \wedge * \wedge *$  that is shortened by  $PCU$ . This covers all components and respective fault modes of the system part PCU.

The isolability items are next used to form isolability clusters where each cluster contains one or more items  $ii$ .

**Definition 4.3 (Isolability Cluster)** *An isolability cluster ( $ic$ ) is a set of isolability items:*

$$ic := \{\cup \{ii_j\}, ii_j \in II, j \leq |II|\} .$$

The set of all isolability clusters  $\cup \{ic_j\}$  is defined to be  $IC$ . Extending the previous example of the isolability item, an isolability cluster  $ic$  can be the set  $\{LH \wedge Shaft001 \wedge Rupture, PCU\}$ . In combination with other clusters this means that every  $ii$  of the  $ic_j$  has to be isolated from all other  $ii \in ic_k$ . The set of clusters is thus used to finally define the isolability requirement.

**Definition 4.4 (Isolability Requirement)** *An isolability requirement ( $IR$ ) is a set of disjunct isolability clusters  $ic$  such that:*

$$IR := \{ic_1, \dots, ic_n\}, ic_n \in IC, 1 \leq n \leq |IC| , \\ ic_k \cap ic_m = \emptyset, \forall ic_k, ic_m \in IR, k \neq m .$$

An example for an isolability requirement is the set:

$$\{ \{LH \wedge Shaft001 \wedge Rupture, PCU\} , \\ \{RH \wedge DownDriveShaft003 \wedge Jam\} \} .$$

The set illustrates the requirement, that all  $ii$ s of the first cluster have to be isolated from the  $ii$ s of the second cluster and vice versa in any failure condition. The requirement is fulfilled if this can be shown by analysis. Considering the example of the two fault modes that was introduced previously, the IR can here be defined such that both fault modes are to be isolated from each other. Regarding the solutions for the fulfillment of the SR, under the assumption that only single faults are considered, only the second solution, which are

the symptoms A and C, can fulfill the IR. The first solution, which is symptom B, has to be extended in order to fulfill both the SR and IR. The general procedure to gain optimal solutions that meet both the requirements is presented next.

### 4.3. Analysis

#### Design for the fulfillment of safety requirements

The result of the simulation of fault modes of the diagnosis model and the observation of the respective symptoms are two cause-effect matrices. In this section it is presented how the safety related matrix is used to gain solutions for a preliminary system design that meets the safety requirement. In order to illustrate the proceeding an example is used. The theoretical approach is defined afterwards.

Figure 10 depicts an example of a safety related cause-effect matrix. In this example two solutions are highlighted. These lead to sets of symptoms and respective monitors that are adequate to detect all safety critical component faults. The non-detectable faults are analyzed before. If they show a safety impact or have to be considered due to other reasons, more and different alternatives have to be defined and the analysis has to be repeated. Otherwise, these fault modes can be cleared from the list of all the considered fault modes.

The rows of the cause-effect matrix are used to form a *Symptom Set* (S). This is defined as the conjunction of symptoms  $m$  that are sensitive to a specific fault mode  $fm_i$ :

$$S(fm_i) := \{m \mid fm_i \mapsto m \wedge m \in M\} .$$

The set  $S(fm_i)$  is thus the set of symptoms such that observing any symptom of the set gives detectability of the fault mode  $fm_i$ . Figure 10 depicts an example for that. Transferring the first row of the matrix into a symptom set gives  $S(fm_1) = \{M1, M2\}$ . This says that observing the symptom  $M1$  or the symptom  $M2$  gives detectability of  $fm_1$ .

In order to provide detectability of all relevant fault modes intersections have to be built that hit each symptom set S at least once. Figure 10 depicts two possible solutions for that.

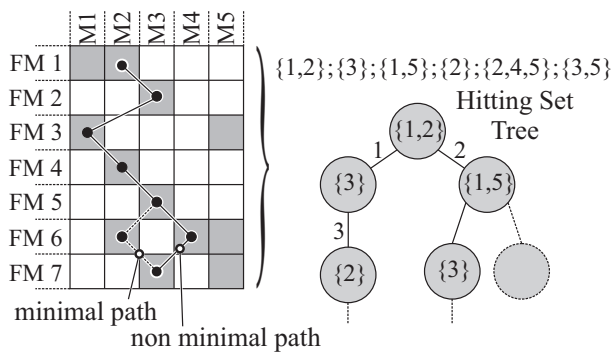


Figure 10. Calculation of minimal hitting sets based on a safety-related cause-effect matrix.

The solutions are different in the way that one of them is minimal whereas the other one includes more symptoms than those that are actually needed for the detectability of all fault modes. Comparing both the solutions the minimal one fulfills the safety requirement whereas the other one does not. In this simple example both solutions were found by manually traversing the rows and columns of the cause-effect matrix. In the general case though this is laborious and hardly possible. In order to find all minimal solutions in a systematic and efficient way the theory of minimal hitting sets (Reiter, 1987)(Greiner et al., 1989) is used. According to that, a minimal hitting set for a multitude of sets is a set that has a non-empty intersection with every set of the multitude of sets. It is thus exactly the set of symptoms  $M^*$  that fulfills the safety requirement.

In the depicted example there are six different symptom sets that are needed for the detectability of the specific fault modes. These are  $\{M1, M2\}$ ,  $\{M3\}$ ,  $\{M1, M5\}$ ,  $\{M2\}$ ,  $\{M2, M4, M5\}$  and  $\{M3, M5\}$ . In Figure 10 these are shortened by using their indices. The theory of minimal hitting sets now gives two solutions that fulfill the safety requirement. These solutions are  $M_1^* = \{M1, M2, M3\}$  and  $M_2^* = \{M2, M3, M5\}$ . In this context minimal means, that when removing one symptom there are specific fault modes that are no longer detectable.

The application to the high lift actuation system of finding minimal sets of symptoms  $M^*$  that fulfill the safety requirement gives 94 solutions. Details on that are depicted in the following sections. In the next step of this section the example is formalized.

The proposed approach of gaining solutions for the fulfillment of the safety requirement is defined in Algorithm 1. As an input the safety related cause-effect matrix  $CEM_{FC}$  is used that states the relation between specific fault modes and their safety critical symptoms. The respective symptom sets  $S$  are conjunct in  $\mathcal{Z}$  that in turn is used as an input for a minimal hitting set algorithm. As a result all solutions for the safety problem are computed.

#### Algorithm 1 All minimal Solutions for Safety Problem

```

1: function DETECTION( $CEM_{FC}$ )
2:    $\mathcal{Z} \leftarrow \emptyset$ 
3:   for  $i = 1 \rightarrow |CEM_{FC}|$  do
4:      $s_i = S(CEM_{FC}(i))$      $\triangleright CEM_{FC}(i) = fm_i$ 
5:      $\mathcal{Z} \leftarrow \mathcal{Z} \cup \{s_i\}$      $\triangleright$  add  $s_i$  to  $\mathcal{Z}$ 
6:   end for
7:    $M^* \leftarrow \text{MinHittingSets}(\mathcal{Z})$ 
8:   return  $M^*$ 
9: end function
    
```

In order to get the *best* solution out off all solutions, criteria for the evaluation have to be defined. In the first step a cost criteria is considered. Cost can be applied to different aspects of the DS architecture. These can be cost for computation of

symptoms, cost for mounting and weight of wires or cost for hardware. The latter is used at the current stage. At this point hardware is referred to sensors. Every symptom is related to features that are gained using information from sensors. An example for that is the symptom *Asymmetry*. Features that are related to this symptom can be gained using information from two sensors, one at each half of the transmission system. In the simplest case this symptom would thus lead to cost of two unit cost. In the general case though, when calculating cost for symptoms, it has to be considered that in the physical system architecture there are already sensors that are needed for the system control. Using these sensors also for the system diagnosis task thus does not lead to additional cost. An example for that is a symptom that is related to information of only one such sensor. This symptom thus does not lead to any unit cost at the current stage of the proposed approach. Therefore, an important aspect is the relation between symptom and sensor. This relation is configured using a database of available sensors and is formed here as a graph. Figure 11 depicts an excerpt of such a graph that shows the relation between symptoms in form of specific monitors and the information they need in form of sensors.

Circles in the graph mark sensors that lead to a specific additional cost whereas the rhomb marks a sensor that leads to no additional costs. In the example the symptom set  $M^* = \{M2, M3\}$  is active. Performing now a reachability analysis on the graph leads to a total cost of:  $C_{tot.} = C_{S3} + C_{S10}$ . At the current stage this will be two unit costs.

Applying the proposed approach of determining cost to all the sets that fulfill the safety requirement, enables to find the most cost effective one. However, it has to be considered that so far cost only includes cost for sensors. In the general case though, the most cost effective symptom set  $M^*$  that fulfills the safety requirement will not also fulfill the isolability requirement. How this is checked and how a design is gained that also meets the isolability requirement is depicted in the next section.

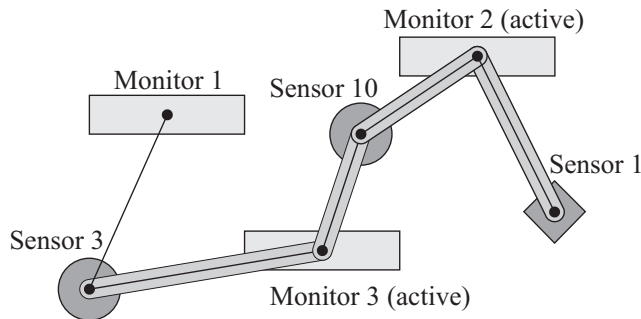


Figure 11. Example of a graph that shows the relation between symptom and sensor.

### Extension of the Design for the fulfillment of isolability requirements

In the case that a symptom of a failure condition is detected, possible root causes shall be determined in accordance with specific isolability requirements. These define how strong possible root causes shall be isolated in order to improve the troubleshooting. Examples are system locations, specific components or specific fault modes. Details on this topic have been explained in Section 4.2. In this section an approach is depicted that can be used to gain all minimal solutions for the isolability problem.

In order to make the isolability items  $ii$  of different isolability clusters  $ic \in IR$  isolable from each other, a prerequisite is, that their respective symptom sets  $S$  have to be different in at least one symptom  $m$ . Referring to the example of Figure 10 the fault modes  $fm_1$  and  $fm_2$  are isolable as they have non overlapping symptom sets. All their symptoms can be used as *candidates* for the fulfillment of a requirement. On the contrary symptoms that are included in all symptom sets can not be used to meet a specific isolability requirement. This assumption is used to define sets of isolability candidates  $C$ :

$$C(a,b) := \{m \mid m \in (S(a) \cup S(b)) \setminus (S(a) \cap S(b))\}, \\
 a, b \in ic_j, ic_k, ic \in IR, j \neq k, |a| \leq |ic_j|, |b| \leq |ic_k| \}.$$

The candidate set  $C(a, b)$  is the set of symptoms  $m$  of the conjunction of symptom sets  $S(a), S(b)$  that are not in the intersection of all symptom sets  $S(a), S(b)$ . The determination of  $C$  is done problem specific. This means that  $a, b$  can be isolability items  $ii$  of a specific isolability sub-problem, but can also cover complete isolability cluster  $ic$ . A necessary prerequisite for the possible fulfillment of a specific requirement is thus that  $C(a, b) \neq \emptyset$  for all  $a = ic_j, b = ic_k, ic \in IR$ . If this is fulfilled, specific candidates for all sub-problems are computed in the next step. In order to find the candidates that lead to an optimal fulfillment of the overall  $IR$  problem a solution is presented in the following. The proceeding is illustrated by examples.

In the previous design phase, solutions were gained that are adequate for the fulfillment of the safety requirement. The next step in the current design phase is thus to check if these solutions already provide candidates for the fulfillment of the isolability requirement. In order to illustrate this point the example from Figure 10 is used. The solutions for the SR that were found, are  $M_1^* = \{M1, M2, M3\}$  and  $M_2^* = \{M2, M3, M5\}$ . An isolability requirement can be that  $IR = \{FM4, FM5, FM6, FM7\}$ . This means that all cluster  $ic_j \in IR$ , which are in this case the isolability items  $FM4, FM5, FM6$  and  $FM7$ , shall be isolable from each other under any condition. Figure 12 depicts the case that  $M_1^*$  is used to check for the fulfillment.

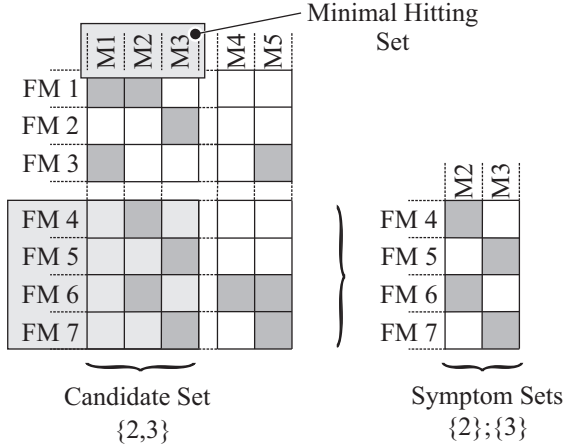


Figure 12. Candidate and symptom sets to check for fulfillment of the isolability requirement  $IR$  using  $M_1^*$ .

Referring to the previous example, the symptom sets for all isolability items are  $S(FM4) = S(FM6) = \{2\}$ , and  $S(FM5) = S(FM7) = \{3\}$ . This illustrates that there are no candidates such that  $FM4$  and  $FM6$ , and  $FM5$  and  $FM7$  can be isolated from each other. The requirement is thus not fulfilled using the solution  $M_1^*$ . The exemplified proceeding to check for the fulfillment of the IR is generalized in Algorithm 2.

**Algorithm 2** Check for the fulfillment of the isolability req.

```

1: function FULFILLMENT( $IR, CEM_{FC}, M_i^*$ )
2:    $\mathcal{A} \leftarrow \emptyset$ 
3:   for  $ic_j, ic_k \in IR, j \neq k$  do
4:      $(a, b) \leftarrow \mathcal{P}(ic_j, ic_k)$   $\triangleright \forall a \in ic_j, b \in ic_k$ 
5:      $C_{a,b} \leftarrow C(a, b)$   $\triangleright$  on  $CEM_{FC}$  and  $M = M_i^*$ 
6:     if  $C_{a,b} = \emptyset$  then  $\triangleright$  Req. not fulfilled!
7:        $\mathcal{A} \leftarrow \mathcal{A} \cup \{(a, b)\}$   $\triangleright$  add  $(a, b)$  to  $\mathcal{A}$ 
8:     end if
9:   end for
10:  return  $\mathcal{A}$ 
11: end function
    
```

The function FULFILLMENT uses the requirement  $IR$  and the safety-related cause-effect matrix  $CEM_{FC}$  as an input and checks if each sub-problem, meaning the power set  $(a, b)$  of elements  $ii$  of the clusters  $ic_j, ic_k \in IR$ , has a non empty candidate set  $C_{a,b} = C(a, b)$ . An output of the function is the set  $\mathcal{A}$ . This set is empty in the case that the requirement is fulfilled for the overall problem, otherwise  $\mathcal{A}$  includes the sub-problems for which further candidates have to be found.

In the current example the set  $\mathcal{A}$  is not empty. This means that the isolability requirement is not fulfilled and further candidates have to be found. Therefore, the maintenance related symptoms are analyzed in the next step. In Figure 13 it is shown that there are four maintenance related symptoms that may lead to a fulfillment of the IR. These are  $\{6, 8, 9, 10\}$  and collected in the candidate set  $C_M$ . The

requirements that were not met by using the candidate set  $C_S$  are  $(FM4, FM6)$  and  $(FM5, FM7)$ . Applying now the candidate set  $C_M$ , local candidate sets for both the requirements are  $C_{FM4, FM6} = \{6, 9, 10\}$  and  $C_{FM5, FM7} = \{6, 8, 10\}$ . All elements of these local sets lead to a fulfillment of the respective sub-problem. In order to gain solutions that fulfill the overall isolability requirement, combinations of symptoms from both the candidate sets have to be built. For the aim not to find all solutions but the minimal ones, a minimal hitting set algorithm is used. This provides all minimal solutions  $\hat{M}_{i,k}$  that lead to a fulfillment of the IR for a specific  $M_i^*$ . For the example these solutions are  $\hat{M}_{1,1} = \{6\}$ ,  $\hat{M}_{1,2} = \{10\}$  and  $\hat{M}_{1,3} = \{8, 9\}$ .

The set  $\hat{M}$  includes all solutions  $\hat{M}_{i,k}$  for the particular isolability problem. The general proceeding to gain  $\hat{M}$  is defined in Algorithm 3. There the function ISOLATION uses the  $IR$ , the maintenance-related cause-effect matrix  $CEM_{MC}$  and the set  $\mathcal{A}$  of unmet sub-problems as an input. The candidates for the fulfillment of the IR are then computed for all elements of  $\mathcal{A}$  and collected in the set  $\mathcal{B}$ . Afterwards a minimal hitting set algorithm uses  $\mathcal{B}$  to calculate all minimal solutions  $\hat{M}$  that fulfill the IR for the specific  $M_i^*$ . The calculation of the candidate set  $C_M$  is omitted as it was used only for demonstration.

In order to gain solutions  $\tilde{M}_i$  that hold for the overall problem, that is the fulfillment of safety and isolability requirements, the conjunction of both the single solutions  $M_i^*$  and  $\hat{M}_{i,k}$  has to be built:

$$\tilde{M}_{i,k} = M_i^* \cup \hat{M}_{i,k}.$$

For the example, solutions to the overall problem are  $\tilde{M}_{1,1} = \{1, 2, 3, 6\}$ ,  $\tilde{M}_{1,2} = \{1, 2, 3, 10\}$  and  $\tilde{M}_{1,3} = \{1, 2, 3, 8, 9\}$ .

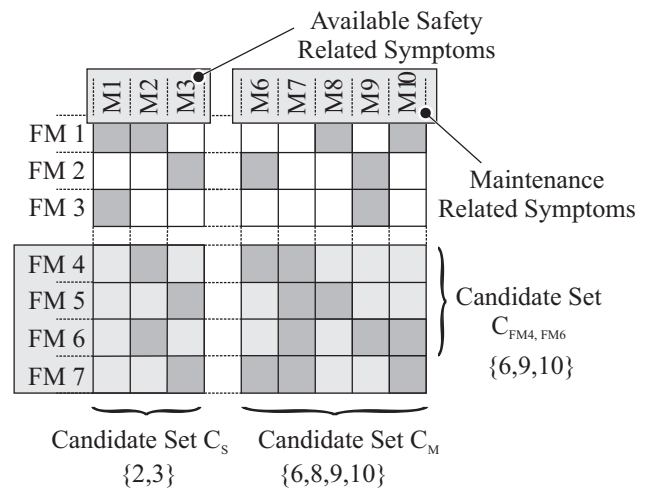


Figure 13. Safety and maintenance related candidate sets for fulfillment of isolability requirement  $IR$  using  $M_1^*$ .

---

**Algorithm 3** All minimal solutions for the isolability requirement

---

```

1: function ISOLATION( $IR, CEM_{MC}, \mathcal{A}$ )
2:    $\hat{M} \leftarrow \emptyset$ 
3:    $\mathcal{B} \leftarrow \emptyset$ 
4:   for  $(a, b) \in \mathcal{A}$  do
5:      $C_{a,b} \leftarrow C(a, b)$  ▷ on  $CEM_{MC}$ 
6:     if  $C_{a,b} = \emptyset$  then
7:       break ▷ Req. can not be fulfilled
8:     end if
9:      $\mathcal{B} \leftarrow \mathcal{B} \cup \{C_{a,b}\}$  ▷ add  $C_{a,b}$  to  $\mathcal{B}$ 
10:  end for
11:   $\hat{M} \leftarrow \text{MinHittingSets}(\mathcal{B})$ 
12:  return  $\hat{M}$ 
13: end function
    
```

---

The proposed approach has now to be repeated and to be applied to all the solutions  $M_i^*$  in order to gain the global set of all the solutions  $\tilde{M}$  that fulfill both the requirements.

In case of the previous example two solutions  $M_1^*$  and  $M_2^*$  for the fulfillment of the safety requirement were found. A detailed description of all the proposed steps of the analysis process for  $M_2^*$  is omitted at this point. However, checking  $M_2^*$  for the fulfillment of the IR, four distinct symptom sets and six respective non-empty candidate sets are found. The minimal hitting set  $M_2^*$  is thus adequate for the fulfillment of both the safety and the isolability requirement. For this particular case it thus holds that  $\tilde{M}_2 = M_2^*$ .

In the end of the depicted process four solutions to the overall problem are gained. These are  $\tilde{M}_{1,1} = \{1, 2, 3, 6\}$ ,  $\tilde{M}_{1,2} = \{1, 2, 3, 10\}$ ,  $\tilde{M}_{1,3} = \{1, 2, 3, 8, 9\}$  and  $\tilde{M}_2 = \{2, 3, 5\}$ .

The complete proceeding of determining solutions to both the safety and the isolability problem is summed up in the Algorithm 4. There the function SOLUTIONS combines the three functions that have been presented in the previous sections.

In order to find the overall optimal solution the relation from Figure 11 combined with a reachability analysis can now be used. This is omitted for the previous example but shown exemplary in the following section for the application of the proposed approach to the high lift actuation system of an Airbus A340-600 aircraft.

## 5. RESULTS FOR THE AIRBUS A340-600 FLAP SYSTEM

The previous sections gave a general overview of the proceeding and the theoretical backgrounds of a model-based approach to the optimal design of a DS architecture. In this section the results of the application of the approach to the flap system of an Airbus A340-600 aircraft are presented. In the first step, in Subsection 5.1, the solution spaces for the fulfillment of both the requirements and the corresponding efforts are shown. In the second step, in Subsection 5.2, two examples of solutions are depicted in detail. There, an overview of the resulting elements of the DS architectures is given.

---

**Algorithm 4** All minimal solutions that meet both the safety and isolability requirements

---

```

1: function SOLUTIONS( $IR, CEM_{FC}, CEM_{MC}$ )
2:    $\tilde{M} \leftarrow \emptyset$ 
3:    $M^* \leftarrow \text{DETECTION}(IR, CEM_{FC})$ 
4:   for  $M_i^* \in M^*$  do
5:      $\hat{M} \leftarrow \emptyset$ 
6:      $\mathcal{A} \leftarrow \text{FULFILLMENT}(IR, CEM_{FC}, M_i^*)$ 
7:     if  $\mathcal{A} \neq \emptyset$  then
8:        $\hat{M} \leftarrow \text{ISOLATION}(IR, CEM_{MC}, \mathcal{A})$ 
9:       for  $\hat{M}_k \in \hat{M}$  do
10:         $\tilde{M}_{i,k} \leftarrow M_i^* \cup \hat{M}_k$ 
11:         $\tilde{M} \leftarrow \tilde{M} \cup \{\tilde{M}_{i,k}\}$  ▷ add  $\tilde{M}_{i,k}$  to  $\tilde{M}$ 
12:       end for
13:     else
14:        $\tilde{M} \leftarrow \tilde{M} \cup \{M_i^*\}$  ▷ add  $M_i^*$  to  $\tilde{M}$ 
15:     end if
16:   end for
17:   return  $\tilde{M}$ 
18: end function
    
```

---

### 5.1. Analysis

The first stage of the overall design procedure consists in the definition of safety and isolability requirements. Two consecutive steps are then executed to design a DS architecture according to these requirements. A diagnosis model is used to provide safety and maintenance related symptoms. In the first step the safety related symptoms are evaluated in order to fulfill the related requirement. As a result 94 different solutions  $M_i^*$  are gained. Strictly speaking  $M_i^*$  is a symptom set, as introduced in Subsection 4.2, but in this context it should be seen as a solution, that always consists of a set of specific sensors, a set of features, a set of symptoms and the diagnostic knowledge, that is stored in the cause-effect matrices. Details about the different elements are given in Section 5.2 by means of two specific examples.

In order to evaluate every single solution  $M_i^*$  the effort for placement of sensors was introduced as a criteria in Section 4.3. This simple cost function should by no means considered to be complete, but as a first step in a multi criteria decision making process. The effort that is induced by every solution  $M_i^*$  is depicted in Figure 14. The solutions are marked there with black dots and are arranged according to their specific index  $i$ . In total, there are 11 solutions that cause the same minimal effort of five unit cost whereas two solutions lead to the highest effort which is 15 unit cost. The two solutions 9 and 19 are depicted in more detail in the end of this section.

In the second step of the approach the fulfillment of specific isolability requirements is checked and the design is extended if necessary. As an example, the requirement  $IR = \{\text{LH-System}, \text{RH-System}\}$  is chosen. This states that under any condition the root cause of a detected failure shall be isolated at least between the left and right hand system half. The



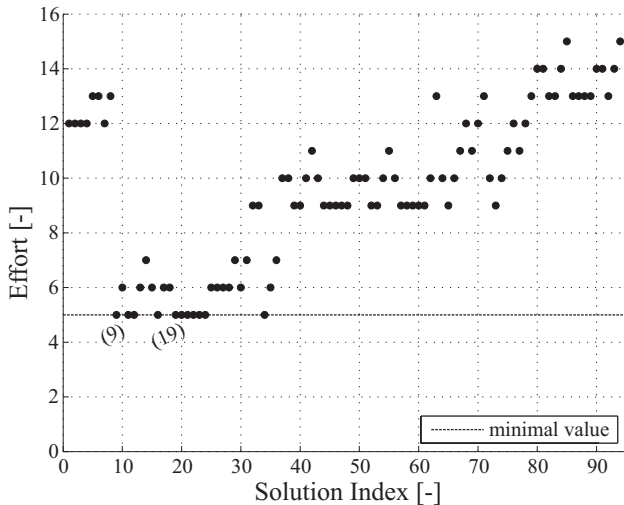


Figure 14. Effort for the fulfillment of the safety requirement.

result of the analysis is, that all of the solutions  $M_i^*$  have to be extended by means of additional symptoms  $\hat{M}_{i,k}$  in order to meet the IR. In detail this means, that some of the previous solutions can be extended by additional symptoms, that are gained using features from signals that are linked to sensors, which are already part of the architecture due to the safety requirement. This case induces no additional effort therefore. In other cases new, maintenance related only symptoms and sensors are needed, which accordingly rises the effort for the fulfillment of the IR.

In total, there are 480 solutions  $\tilde{M}_{i,k} = M_i^* \cup \hat{M}_{i,k}$  that meet both the requirements in a local optimal way with respect to minimality. An overview of the effort for every single solution  $\tilde{M}_{i,k}$  is given in Figure 15. Each tuple  $(i, k)$  is there represented by a new sequential index  $i^*$ . The solutions 9 and 19 from Figure 14 are highlighted for two respective tuples  $(i, k)$  using the old index  $i$ . Both the solutions had to be extended using an additional sensor in order to meet the IR.

A result of the analysis is, that all of the optimal solutions found in the first step remain optimal also in the second step. In addition there are five more solutions that cause the same minimal effort of six unit cost. A reason for that is, as mentioned previously, that for some solutions from step one there was no need for additional sensors in order to meet the IR, but only for additional symptoms gained from data from already available sensors. Due to that, some solutions from step one did not cause a rise in effort during the analysis in step two. Summarizing the results, in the end of the analysis process, there are 16 solutions that meet both the requirements in a global optimal way with respect to minimality and the single cost function that has been considered.

In order to choose one solution that has to be detailed in further steps, more criteria for the evaluation have to be intro-

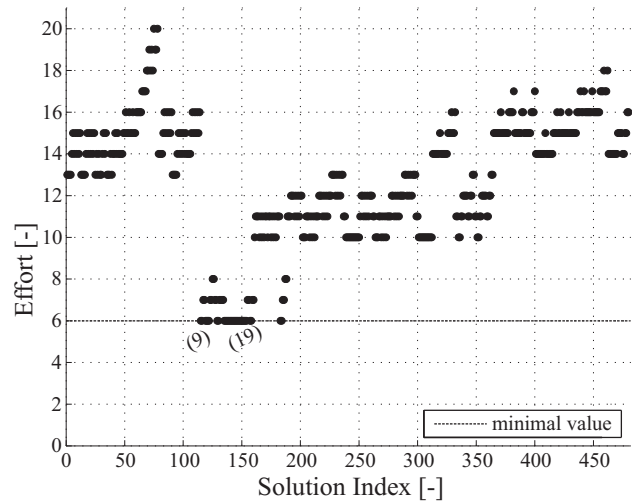


Figure 15. Effort for the fulfillment of both the safety and the isolability requirement.

duced. Possible extensions are the consideration of an isolability performance metric and the use of more realistic cost factors amongst others.

Apart from detailing of the cost function, a further important aspect is, that only one design case, that means one operating condition, under a certain side condition has been considered so far. In the current case the analysis was based on an extension of the system against high air loads. Further cases that have to be analyzed are different operating speeds and other load conditions, so that in the end an overall optimal solution is gained that holds for all conditions considered.

## 5.2. Discussion

The solutions 9 and 19 that were marked in both the previous figures belong to the set of global optimal solutions. In the following, both of them are depicted in detail and differences are discussed.

An overview of the resulting preliminary design of a DS architecture according to solution 9 is given in Figure 16. On the lowest layer of this architecture there are seven sensors. These are three position sensors, three proxy sensors and one load sensor. The sensors  $a$  and  $c$  are placed on the left and right hand system side, whereas all the other sensors are only placed once. At this point of the design procedure these sensors are basically reduced to the physical quantity they measure, so that aspects of redundancy have not been considered yet which may demand additional sensors in future. Although there are currently seven sensors, the effort for the architecture consists of six unit cost. This is based on the fact, that the position sensor located at the PCU is also used for the system control and by that comes for free for the diagnostic tasks.

The sensor signals are evaluated by six monitoring devices.

Five of them are related to the safety task and one is only needed for the fulfillment of the isolability requirement. This device is marked as maintenance related only. All of the devices are built up like it is shown for the device *A*. In the first step different signals are used to generate features, that in the second step are compared to a threshold in order to detect symptoms. The example of device *A* is related to the failure condition *Asymmetry*. In this case, position measurements from both ends of the mechanical transmission system are used to calculate the absolute difference between both the position signals. If this feature exceeds a certain threshold the symptom of the failure condition is detected. In order to take adequate measures and to generate maintenance messages all of the detected symptoms are correlated afterwards. In order to determine potential root causes the diagnostic knowledge is used. At this point of the preliminary design phase it is stored in the cause-effect matrices. Due to the isolability requirement, the root cause can then be isolated at least between the left and right hand system side by using the knowledge.

The resulting architecture of the DS according to solution 9 is very close to the Airbus A340-600 flap system's current DS's architecture. The focus of the current architecture is on the detection of FCs though. Due to confidentiality reasons no details can be stated about that at this point. In the proposed

new concept though, less monitoring devices are needed in order to meet the stated safety requirement. Furthermore one additional, maintenance related only, device is added in order to fulfill the isolability requirement. Due to the non-safety critical functionality of this device a low development assurance level can be applied in the next design steps. It has to be considered though, that a loss of the respective function could lead to more cost and effort for the troubleshooting. The consideration of this aspect and the detailing of all the elements of the architecture will come in further design steps. Next, a second alternative of a preliminary architecture of a DS is presented.

Figure 17 depicts the resulting architecture according to solution 19. In this case, there are again used information provided by seven sensors. These are position and proxy sensors and one load sensor. The effort consists of six unit cost, due to the fact, that again the position sensor, that is located at the PCU, comes for free. All of the sensor signals are evaluated by six safety related monitoring devices and again one additional, maintenance related only, device.

Comparing both the architectures they show clear similarities. Both are using only information that are provided by position, proxy and load sensors. They differ in the way, the sensors are located and features are extracted from the measurements.

The architecture according to solution 9 uses information that are provided by position sensors *a*, which are located at the outermost ends of the mechanical transmission system.

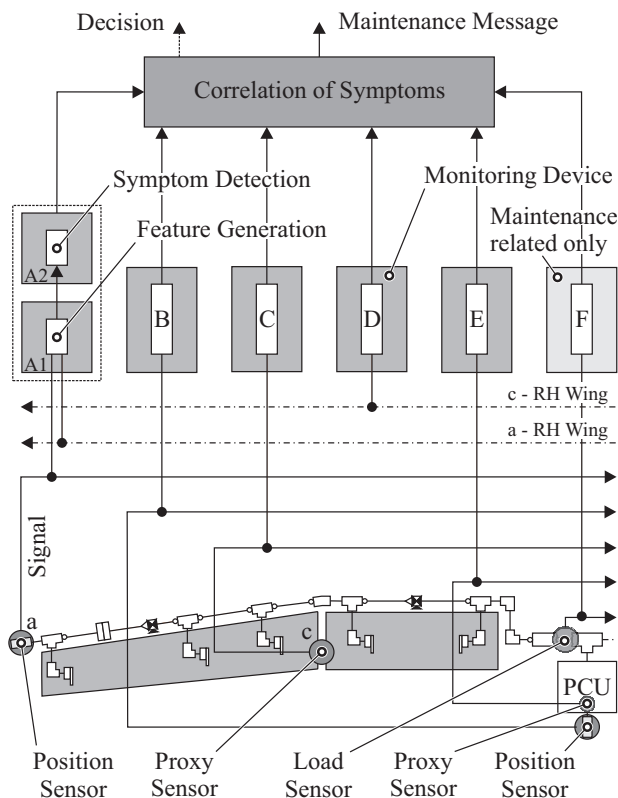


Figure 16. Detailed overview of the resulting architecture according to solution 9.

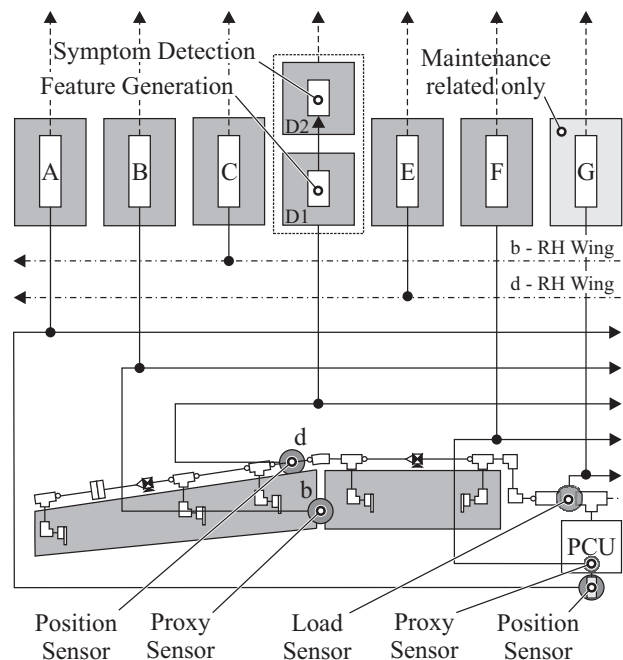


Figure 17. Detailed overview of the resulting architecture for solution 19.

The monitoring device *A* is in charge of extracting a feature from these information in order to detect the symptom of an *Asymmetry* failure condition. In the architecture according to solution 19 the sensors *a* are missing. Instead, two sensors *d* are used, that are located on the transmission system between the in and outboard flap of each system side. In detail, the sensors *d* provide position measurements. The monitoring devices *D* and *E* use these measurements to calculate a speed signal that is used as a feature in order to detect a symptom of an *Overspeed* failure condition.

The differences in both the architectures result from the fact, that the failure conditions *Asymmetry* and *Overspeed* overlap. The safety requirement was defined such that the overlapping should always be as minimal as possible, so that the two alternatives result as solutions. If this holds for all the conditions that have to be considered, has to be checked in the extended design procedure, as mentioned previously.

In the current case only unit cost factors for the placement of sensors have been considered in order to evaluate all the solutions. Extending this approach by taking the weight for wiring into account, solution 19 seems to be a little advantageous. On the other hand though, the computational effort rises, due to the fact that more monitoring devices are needed. Therefore, it is obvious that further criteria have to be identified and considered in order to determine the overall optimal solution.

## 6. RELATED WORK

The development of diagnostic systems and in general of prognostic and health management systems has been studied by various researchers under different aspects. In the following, selected examples of works are presented that are related to the topics of this paper.

In the work of (Kurtoglu, Johnson, Barszcz, Johnson, & Robinson, 2008) a design methodology for the development of system health management is introduced. This is called the Functional Fault Analysis (FFA) and is based on a functional model of the system to be analyzed. Outputs of the FFA are timing analyses for fault-effect propagation, ambiguity statistics for fault isolation and the model itself for online integration. The definition of effect nodes and test points shows similarities to failure and maintenance conditions from our paper, but no clear policies about the optimal selection of the nodes and no integrated requirement based process are shown.

Another approach that combines an extended tabular FMECA and a functional block diagram to a functional graphical health management model is presented in (Kacprzyński, Roemer, Hess, & Bladen, 2001) and (Kacprzyński, Roemer, & Hess, 2002). The model and its elements contain various attributes, that are used as input to a fitness function for a genetic optimization procedure. Output of the analy-

sis is a health management configuration consisting of sensors and algorithms that has the highest system reliability to cost/benefit ratio. Not only diagnostic but also prognostic aspects are addressed. Temporal information, that are manually inserted in the process of (Kurtoglu et al., 2008), are not considered completely, but by means of propagation probabilities and response models (Kacprzyński et al., 2002). While in our approach, for each task of the design procedure, a requirement is introduced, in order to keep the process traceable, and to extend the model complexity only if needed, the design procedure of (Kacprzyński et al., 2001) is basically done using only one model and one single iteration step.

The papers mentioned and other similar ones focus on the development of functional, qualitative models that provide information about fault propagation and serve as basis for different analyses. Input to most of them is a manually created FMECA. Matters of optimality are only dealt with in a few and integrated processes for the design and test are addressed only marginally. The focus of our paper was to define the framework of a traceable process, based on physical models, where the design and test of a diagnosis system is done in consecutive steps according to posed requirements. A set of two requirements was introduced. This however should not be considered to be complete but to be the basic starting point. The further discussion on related work will therefore focus on papers, that have problem formulations with similarities to our paper.

In the work of (Scandura, 2005) an overview of a general framework for the development of integrated vehicle health management systems is introduced and the importance of the combined consideration of a philosophy, a methodology and a continuous process is emphasized. Policies like a fault detection and isolation philosophy, the optimal sensor quantity and placement guidelines are mentioned, but no approaches to design a system accordingly are presented.

A detailed description of an approach that is used by Boeing for the development of model-driven integrated support architectures is given in (Ofsthun & Wilmering, 2004). There, a process is defined, that is centered around requirement based design and test. The framework of the process is close to our approach, whereas the realization is different. While our approach uses a performance model, that tries to capture the physics of operation under all normal and failure conditions, their starting point is a qualitative model of the system. The result of their process is a directed timed failure propagation graph that is formalized as platform executable code. In contrast, our approach focuses on the configuration of a generic diagnostic engine and the generation of diagnostic rules as presented in (Modest, Schories, et al., 2011).

## 7. CONCLUSION

This paper addressed a model-based development approach for diagnosis systems of high lift actuation systems. The focus was on the definition of requirements and the optimal conceptual design of a diagnosis system architecture. A diagnosis model was developed therefore in the first step. This provided safety and reliability related cause-effect matrices. These were used as an input for a two step design process. The first step was about the design for fulfillment of safety-related requirements, whereas the second step focused on the extended design to meet reliability-related requirements. A set of minimal solutions that met both the requirements was found in the end. In order to determine the optimal solution a cost criteria was introduced, so that optimality was defined in terms of minimality and a single criteria for effort. As a result of the overall proceeding a preliminary specification of an optimal design for a diagnosis system architecture was gained. This specification showed the type and location of sensors and respective monitoring devices by means of signals, features and symptoms. All steps of the process were illustrated by examples and defined afterwards. The results of the application to the high lift actuation system of an Airbus A340-600 aircraft were presented and discussed in the end. In the following, an outlook about open points and future research activities is given.

The current approach considered only one design case for the HLS operation and only unit costs as criteria for effort. Future work focuses therefore on the consideration of different design cases and more advanced evaluation criteria in order to find the overall optimal solution. A performance criteria of isolability might be one point for extension as well as the consideration of weight factors for wiring. Furthermore the other parts of the proposed development approach have to be worked out. The next steps are the detailed system design as well as the component design. This includes further studies on monitoring, sensor devices and BIT functionalities. Aspects that have to be addressed are the definition of the kind of sensor, meaning hardware or a virtual sensor, and the evaluation of different strategies to the BIT, amongst others. Strongly related to that is the diagnosis model. Up to now, a quasi static model has been used. The reason to start with such a model was based on the concept to increase the model's complexity according to the posed requirements from stage to stage and only if needed. The current model can thus be seen as a first stage model. It has now to be investigated what the additional value of second stage models, i.e. dynamic models, offers and how to link both the stages. Important aspects that have to be answered by that are the support for the detailed design phases and the consideration of temporal aspects of the symptoms.

A complexity analysis for the minimal hitting set algorithm was not in the scope of this paper, but is an important aspect

that has already been dealt with and will be discussed in a further paper together with the extended design procedure.

## ACKNOWLEDGMENT

The authors thank the Airbus Operations GmbH for sponsoring and supporting their work.

## NOMENCLATURE

<i>ACQ</i>	Acquisition
<i>BIT</i>	Built In Test
<i>CEM</i>	Cause Effect Matrix
<i>DISP</i>	Disposal
<i>DOC</i>	Direct Operating Cost
<i>FC</i>	Failure Condition
<i>IR</i>	Isolability Requirement
<i>LCC</i>	Life Cycle Cost
<i>HLS</i>	High Lift Actuation System
<i>HMS</i>	Health Management System
<i>MC</i>	Maintenance Condition
<i>MRO</i>	Maintenance Repair Overhaul
<i>OPS</i>	Operation and Support
<i>PCU</i>	Power Control Unit
<i>RDTE</i>	Research Development Test Evaluation
<i>SFCC</i>	Slat Flap Control Computer
<i>SR</i>	Safety Requirement
<i>WTB</i>	Wing Tip Brake
<i>fm</i>	Fault Mode
<i>ic</i>	Isolability Cluster
<i>ii</i>	Isolability Item
<i>m</i>	Symptom
<i>M</i>	Set of all Symptoms
<i>S</i>	Set of Symptoms for specific Fault Mode

## REFERENCES

- Bunus, P., Isaksson, O., Frey, B., & Munker, B. (2009). Model-Based Diagnostics Techniques for Avionics Applications with Rodon. In O. von Estorff & F. Thielecke (Eds.), *Proceedings of the 2nd International Workshop on Aircraft System Technologies*. Shaker.
- Greiner, R., Smith, B. A., & Wilkerson, R. W. (1989). A Correction to the Algorithm in Reiter's Theory of Diagnosis. In Elsevier Science Publishers Ltd. (Ed.), *Artificial Intelligence* (Vol. 41, pp. 79–88). Essex, UK: Elsevier Science Publishers Ltd.
- Haskins, C. (Ed.). (2006). *Systems Engineering Handbook - A Guide For System Life Cycle Processes And Activities* (3rd ed.). INCOSE - International Council On Systems Engineering.
- Isermann, R. (2006). *Fault-Diagnosis Systems*. Springer.
- Kacprzyński, G. J., Roemer, M. J., & Hess, A. J. (2002). Health Management System Design: Development,

- Simulation and Cost/Benefit Optimization. In *2002 IEEE Aerospace Conference Proceedings* (Vol. 6, p. 3065-3072).
- Kacprzynski, G. J., Roemer, M. J., Hess, A. J., & Bladen, K. R. (2001). Extending FMECA-Health Management Design Optimization for Aerospace Applications. In *2001 IEEE Aerospace Conference Proceedings* (Vol. 6, p. 3105-3112).
- Kurtoglu, T., Johnson, S. B., Barszcz, E., Johnson, J. R., & Robinson, P. I. (2008, october). Integrating System Health Management into the Early Design of Aerospace Systems using Functional Fault Analysis. In *2008 International Conference on Prognostics and Health Management Proceedings* (p. 1-11).
- Lulla, C. (2011). Functional Flexibility of the A350XWB High Lift System. In Deutsche Gesellschaft für Luft- und Raumfahrt (DGLR) (Ed.), *Tagungsband DLRK 2011* (pp. 385-392).
- Modest, C., Grymlas, J., Schories, K., Lüdders, H. P., & Thielecke, F. (2011). Model-Based Development of Control and Diagnosis Concepts for Multifunctional Fuel Cell Systems. In *9th European Workshop on Advanced Control and Diagnosis, ACD 2011*. Budapest.
- Modest, C., Schories, K., Lüdders, H. P., & Thielecke, F. (2011). A Model-Based Development Approach for a Diagnostic System for a Multifunctional Fuel Cell System. In Society of Automotive Engineers (Ed.), *SAE International Journal Of Aerospace* (Vol. 4, p. 1324-1333). Warrendale, PA: SAE International.
- Ofsthun, S. C., & Wilmering, T. J. (2004, March). Model-Driven Development of Integrated Health Management Architectures. In *2004 IEEE Aerospace Conference Proceedings* (Vol. 6, p. 3692-3705).
- Recksiek, M. (2009). Advanced High Lift System Architecture With Distributed Electrical Flap Actuation. In O. v. Estorff & F. Thielecke (Eds.), *Proceedings of the 2nd International Workshop on Aircraft System Technologies* (pp. 49-59). Shaker.
- Reiter, R. (1987). A Theory of Diagnosis from First Principles. In Elsevier Science Publishers Ltd. (Ed.), *Artificial Intelligence* (Vol. 32, pp. 57-95). Essex, UK: Elsevier Science Publishers Ltd.
- Roskam, J. (2006). *Airplane Cost Estimation : Design, Development, Manufacturing and Operating* (3rd ed.). Lawrence, Canada: DARcorporation.
- SAE (Ed.). (1996). *Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment* (No. 4761).
- Scandura, J., Philip A. (2005, October). Integrated Vehicle Health Management as a System Engineering Discipline. In *Digital Avionics Systems Conference, 2005. DASC 2005. the 24th* (Vol. 2).

#### BIOGRAPHIES

**Christian Modest** is a research assistant at the Institute of Aircraft Systems Engineering of the Hamburg University of Technology. He received the German academic title Dipl.-Ing. in mechanical engineering from the Hamburg University of Technology in 2010. His research interests include diagnostics, health management systems and aircraft systems engineering.

**Frank Thielecke** is the head of the Institute of Aircraft Systems Engineering of the Hamburg University of Technology. For 10 years he was in charge of the Systems Automation Department at the Institute of Flight Systems of the German Center for Aeronautics and Space (DLR). He has a long-standing experience in fields of model-driven systems engineering, flight systems design, reliability analysis as well as diagnostics and testing.

# Fault Diagnostics Using Network Motif Signature

Tsai-Ching Lu<sup>1</sup>, David L. Allen<sup>1</sup>, Yilu Zhang<sup>2</sup>, and Mutasim A. Salman<sup>2</sup>

<sup>1</sup>*HRL Laboratories LLC, Malibu, California, 90265, U.S.A*

*tlu@hrl.com  
dlallen@hrl.com*

<sup>2</sup>*General Motors, Warren, Michigan, 48090, U.S.A*

*yilu.zhang@gm.com  
mutasim.a.salman@gm.com*

## ABSTRACT

In modern vehicles, controls are distributed over multiple Electronic Control Units (ECUs) that are connected through in-vehicle communication networks. Fault diagnostics for such a distributed control system is very challenging, which has resulted in many no-trouble-found (NTF) cases during warranty repairs. To address this problem, we propose a novel network-theoretic approach that detects, identifies, and localizes faults using both the structure of the communication network (topological information) and message flow information. The proposed method not only enables the characterization of normal operation and a-priori known faults across communication networks, which is already beyond the current practice of individual ECU centric diagnostics, but also the diagnostics of unknown or cascading failures emerging from unexpected operational environments.

## 1. INTRODUCTION

In-vehicle electrical and electronic systems (EES) are embedded systems that implement advanced fuel-economy, emission control, safety, convenience, and comfort features. Although EES systems are largely deterministic with designed subsystem and component interactions to accomplish desired functionality, it may become stochastic when vehicles operate in extreme and unexpected conditions with fault propagations that were not anticipated during the design, testing, or validation stages. It is the emerging faults and cascading failures that pose ever growing challenges to diagnostics and prognostics in complex electronics, especially in complex systems such as in-vehicle EES.

The state-of-the-art diagnostic techniques for in-vehicle ESS

are generally based on Diagnostic Trouble Codes (DTC). Each individual ECU records the DTCs when internal, local, error-checking routines detect fault conditions. Technicians then inspect DTC archives, and perform diagnostic tests to determine the root causes. In practice, one fault may be detected by multiple ECUs and trigger multiple DTCs. One single fault may also cascade from one part of the system to other parts of the system, and trigger multiple DTCs. In either case, it has been a significant challenge to identify the root cause of multiple observed DTCs. This situation is particularly challenging for communication related DTCs. Moreover, the error checking routines in individual ECUs usually consider only faults anticipated during the design phase, which don't include unknown failures or cascading failures that emerge from unexpected operational conditions.

In this paper, we propose a novel network-theory approach to identify and localize faults based on topological information of the in-vehicle communication network, network motif fault signatures, and message flow information over the network. The proposed methodology is an enabler to

1. Characterize normal operation and a-priori known faults across communication networks which is beyond the current myopic view of observability employed by individual ECUs;
2. Identify and localize unknown or cascading failures emerging from unexpected operational conditions.

The rest of the paper is organized as follows. After a brief survey of related research in Section 2, we present the proposed approach in Section 3, and follow up with technical description for each module of the approach in Sections 4-7. We then report our initial empirical study of the proposed methodology on a multi-ECU system simulated in Vector CANoe (vector.com) in Section 8.

---

Lu et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

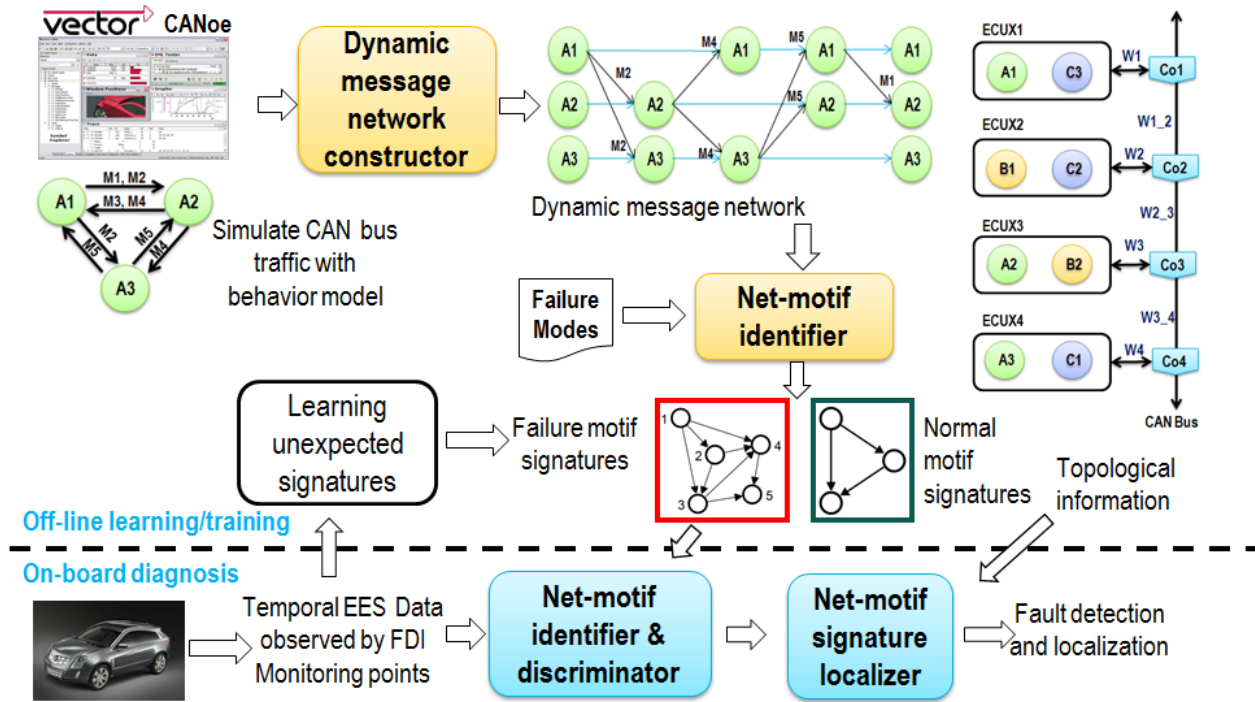


Figure 1. Normal motif and fault motif signatures are trained during the off-line learning/training stage to create a detector and localizer that can be deployed into diagnostic software tools or real-time diagnosis assistance on vehicles. On-board fault detection and isolation (FDI) monitoring point is selected based on methodology in Lu et al. (2011).

## 2. RELATED RESEARCH

Net-motifs are sub-graphs of fixed size or length which appear more commonly than others in a network. In this paper, net-motifs represent the observation of message transmission patterns on the communication bus.

Milo et al. (2002) first introduced the concept of network motifs and demonstrated their abundance in biochemistry (transcriptional gene regulation), ecology (food webs), neurobiology (neuron connectivity), and engineering (electronic circuits, World Wide Web). The net-motif concept has not been applied to in-vehicle communication networks of EES. Rather than focusing on the existences of network motifs in engineering field, we are taking the net-motifs, combining with topological information, to perform fault identification and localization.

Wang et al. (2009) described a network management system to diagnose network traffic faults for TCP/IP communication networks. Their model is trained to derive fault signatures from temporal patterns in historical network traffic data, in conjunction with network topological information. Topological information is demonstrated for selfing/neighbors, containing/contained, down/upstreaming, and tunneling. The key concept is utilizing the event bursts for steps of fault signature learning and

indexing. The term motif is mentioned, but not specifically utilized. Only pair-wise event interactions are considered. There is no mentioning of in-vehicle networks and behavior models.

Dijev et al. (2011) introduced graph based statistical analysis on network traffic of internet for intrusion activity or malicious behaviors. The telescoping graph is proposed to capture the decomposition of a protocol graph which describes activity between hosts in the network for a given protocol, e.g. SSH protocol graph constructed from packet header. A discrete hazard model is learned to detect anomalies from decomposed TSG graphs. Although statistical graph-based methods are used, there is no direct application to in-vehicle networks, and there is no consideration of topological information.

## 3. SYSTEM AND METHODOLOGY

We develop a system and methodology that assists with fault identification and localization for in-vehicle embedded electrical and electronic systems (EES) using network motif signatures. The system takes behavior models, topological information, and communication traffic as input, and outputs net-motif-based fault signatures and the candidate fault sets, if emerging or a-priori known failures occurs in vehicles. The system is first trained off-board to characterize

net-motif signatures under normal operations and known failure-modes in design phase, and later deployed on-board or embedded in a troubleshooting tool to assist failure net-motif signature detection and fault localization (Figure 1). The details of each module in Figure 1 are described in the following sections: Dynamic Message Network Constructor (Section 4), Net-motif Identifier (Section 5), Net-motif Signature Detector (Section 6), and Net-motif Signature Localizer (Section 7). Figure 2 shows the data flow of the proposed fault detection and localization using net-motif signatures.

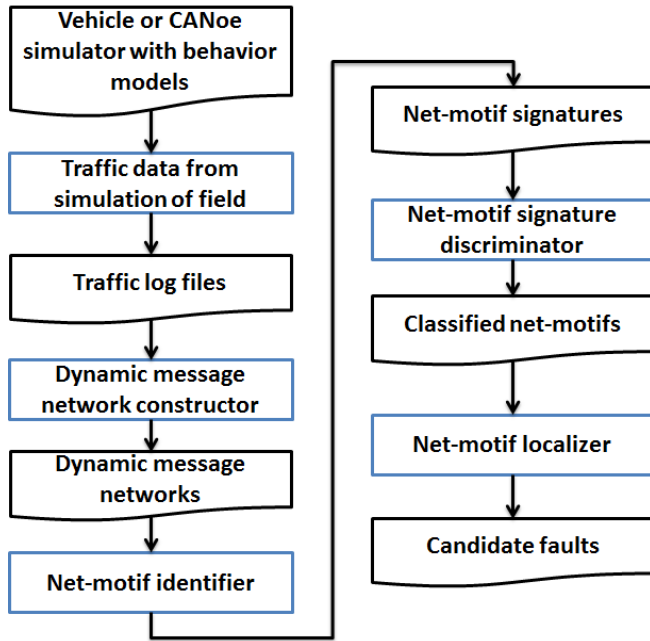


Figure 2. *Dynamic message network constructor* takes traffic log files to generate dynamic message networks; *Net-motif identifier* finds network motifs in constructed dynamic message networks; *network-motif signature detector* compares identified motif profiles against normal operation motif profiles to classify fault motifs; *net-motif signature localizer* employs topological information and behavior models (designed message flows) to generate candidate fault sets.

#### 4. DYNAMIC MESSAGE NETWORK CONSTRUCTION

We develop the dynamic message network constructor to take network traffic as input, and output a dynamic message network where nodes represent ECUs and edges represent messages flows between ECUs. The pseudo code for one instantiation of constructing a dynamic message network is described as follows.

1. Initialize discrete counter  $T=1$ ;
2. For each message  $[ECU_i \rightarrow ECU_{j, k, \dots}]$ 
  - a. Let  $t_{tx} = T$ ;

- b. If the sender of the current message is found as a receive-only node (with only incoming edges) within the previous  $W$  seconds, counted from the current message timestamp (simulation or real time values)
  - i. Set  $t_{tx}$  equal to the counter value associated with  $ECU_i$ ; change  $ECU_i$  to transmit node from receive-only node;
- c. Else
  - i. create a new  $ECU_i$  transmit node and associated it with the counter value  $t_{tx}$
- d. Create receive nodes  $ECU_{j, k, \dots}$  if they do not already exist at the counter value  $t_{tx}+1$  (note this may not be  $T+1$  if used previous receive node);
- e. Add edges from associated  $ECU_i \rightarrow ECU_{j, k, \dots}$  node
- f. Set  $T = t_{tx}+1$  (the receive node's counter value of the current message)

An example dynamic message network generation is shown in Figure 3.

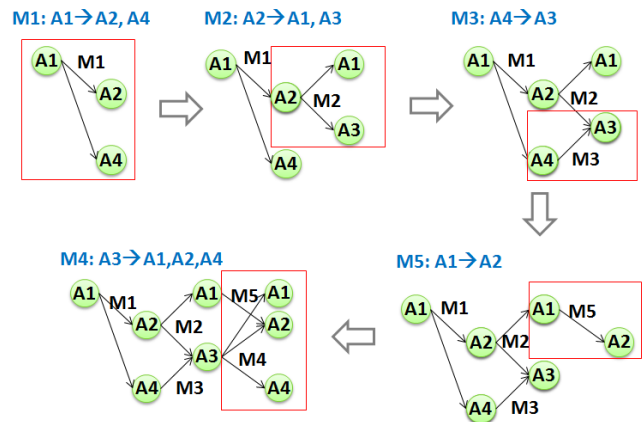


Figure 3. Dynamic message network generation: Messages M1, M2, M3, M5 and M4 are detected and corresponding networks are generated after processing each message. Due to the sliding window parameter, the message M3 from A4 to A3 can have exchangeable temporal sequence with message M2 from A2 to A1 and A3, i.e., two directed edges into A3 before A3 can respond to send out M4.

The dynamic message network constructor will be used first in off-line processing to establish message networks for identifying normal operational net-motifs and net-motif fault signatures. The constructor will later be utilized in on-board diagnosis. The parameter  $W$  second is a tunable such that dynamic message networks could be adapted to different communication architectures in vehicles.

#### 5. NET-MOTIF SIGNATURE IDENTIFICATION

Taking a dynamic message network as input, the net-motif identifier detects net-motifs by extracting and grouping sub-



graph patterns in the network. The output of the net-motif identifier are detected net-motifs with counts providing a distribution of motifs. We build our net-motif identifier based on the efficient motif discovery algorithm in (Wenick, 2006) and efficient sub-graph isomorphism algorithms in (Cordela et al., 2004).

In the extraction step, we will enumerate size-k sub-graphs by growing a recursion tree (RT) as follows:

1. Assign nodes of input graphs with ordered indices
2. Each node in a RT will be associated with two sets sub-graph sets ( $V_{sub}$ ) and exclusive neighbors ( $V_{ecn}$ ) except the root node.
3. Nodes in  $V_{sub}$  are called spawning nodes and nodes in  $V_{ecn}$  are nodes with indices greater than the associated spawning nodes in  $V_{sub}$ .
4. Recursively grow the RT to the k-level which would be the sub graph with size k.

In the grouping stage, we implement graph isomorphism algorithms in (Cordela et al., 2004) to group extracted sub-graphs into canonical patterns of net-motifs. Figure 4 (a) and (b) illustrate the steps in net-motif extraction with recursive tree and identified motifs for an example message network.

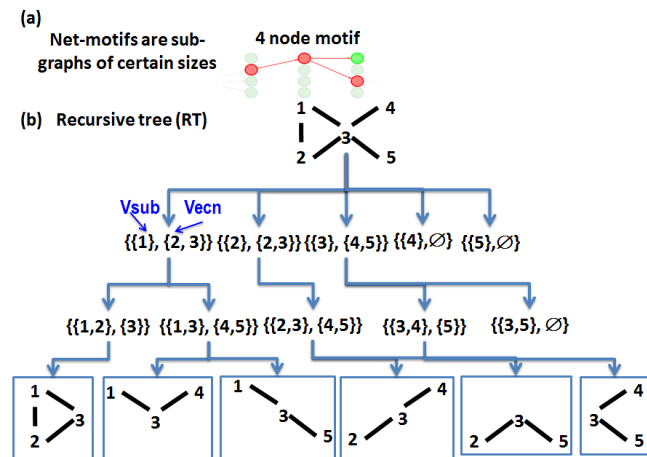


Figure 4. Illustration of Net-motif identification. (a) shows the example 4-node net-motifs in a fragment of dynamic message network; (b) shows the recursive tree constructed to identify 3-node net-motifs in the example 5 node network;

## 6. FAULT DETECTION USING NET-MOTIF SIGNATURE DISCRIMINATION

Given net-motif distributions, we will learn a net-motif discriminator to classify motifs into different modes of operations, such as normal and failures, and detect net-motif of failure operations by comparing against motifs of normal operations to produce fault motif signatures.

We train net-motif discriminator as follows:

1. Simulate or collect network traffics under normal operational conditions
2. Inject failures in network traffic simulation tools, such as CANoe, or inject failures onto real vehicles to collect fault traffic for known failure modes
3. Perform dynamic message network construction to build network for collected traffic data
4. Perform net-motif identification to derive net-motif distributions from normal and failure dynamic message networks.
5. Learn net-motif discriminator on net-motif distributions using machine learning algorithms, such as probabilistic graphical models or other classification algorithm such as k-nearest neighbor or Support Vector Machines.
6. Perform cross-validation to validate the performance of trained net-motif discriminator.

The output of net-motif discriminator is categories of net-motif signatures under normal or failure conditions.

## 7. FAULT LOCALIZATION USING NET-MOTIF SIGNATURES

Although the trained net-motif discriminator could achieve high detection and classification accuracy for known failure modes, it is impossible to simulate all combinations of all known failure modes in various failing sequences. Moreover, unknown failures may occur when vehicle operating in extreme or unexpected conditions. In this section, we describe the net-motif failure signature (NMFS) localizer that detects and localizes unknown faults via net-motifs and topological information.

The topological information is provided as an input to NMFS in the form of expected message flows from behavior models. Each message is associated with a sequence of flow paths starting from a sender ECU, going through sequences of wires, connectors, and gateways to finally reach the receiver ECU. For example, ECU1 sends message M1 on the bus and ECU2 and ECU3 are in the same virtual networks which are supposed to listen and receive message M1. We will write topological information of message flow as  $\{ECU1, wire1, connector1, \dots, wireN, ECU2\}$  and  $\{ECU1, wire1, connector1, \dots, wireI, ECU3\}$ .

Given such topological information and profiles of normal operational motifs, we perform the following steps to do fault localization for unknown faults:

1. Perform dynamic message network construction on traffic data
2. Perform net-motif identification to extract net-motifs from the dynamic message network

3. Apply sub-graph matching to identify the deviation of monitored net-motifs with those in net-motif profiles of normal operations.
4. Trace deviations (observed or missing messages) over topological information of message flow to discern the

ambiguity sets of possible faults using set intersection and differentiation operations.

The output of NMFS is an ambiguity set of likely faults for net-motif failure signatures that deviate from normal operations. Figure 5 illustrate the concept.

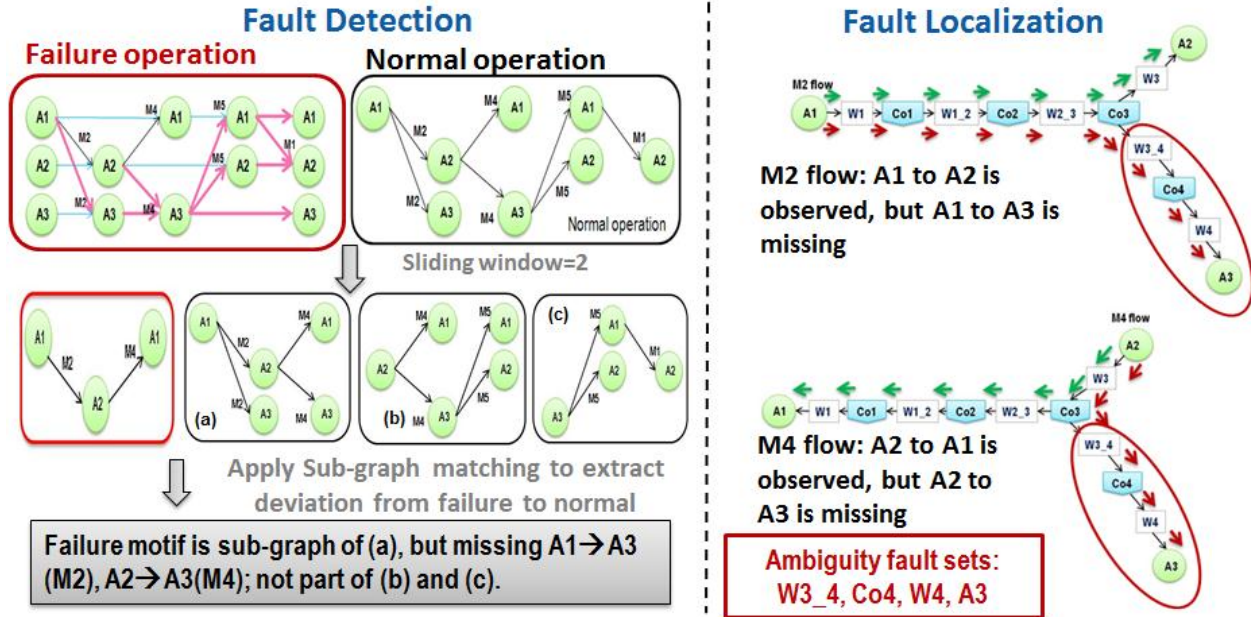


Figure 5. Fault detection localization using net-motif fault signatures. The left-panel shows the steps of fault detection which compare net-motif identified from dynamic message network of failure operation against net-motifs of normal operation. The step of fault localization applies sub-graph matching to identify the observed net-motif fault signature, which is a sub graph of a net-motif in normal operation. This leads to perform the operation of set differentiation over the message flows on topological structure (CAN bus) for observed (green) and unobserved (red) message flows, and recommend ambiguity fault sets of components such as wires , connectors, or ECUs.

### 8. AN EXAMPLE STUDY

We show an example communication network with 4 ECUs and two behavior features A and B. The topology is shown in Figure 6 (a) with wires (annotated with W\*) and connectors (annotated with Co\*). The behavior models are shown in Figure 6 (b). The canonical dynamic message networks for normal operations for the two features are shown in Figure 6 (c). We simulate 2 hardware wire failure and 4 software failure for software components in two features as in Figure 6 (d).

We simulate normal and failure operations with noise, and generate 21 runs of varying durations corresponding to 7 modes of operation with 3 runs for each mode. We perform net-motif identification over each log file and derive its net-

motif distribution. We then perform net-motif discrimination to derive a similarity score by comparing pair-wise differences in the identified distributions. We use the following cosine similarity function:

$$sim(NM, FM) = \frac{\sum_i NM_i \times FM_i}{\sqrt{\sum_i (NM_i)^2} \sqrt{\sum_i (FM_i)^2}}$$

Where  $NM_i$  is the counts of one motif distribution, and  $FM_i$  is the counts of the second motif distribution. The dis/similarity of net-motifs for different modes of operations is shown in Figure 7.

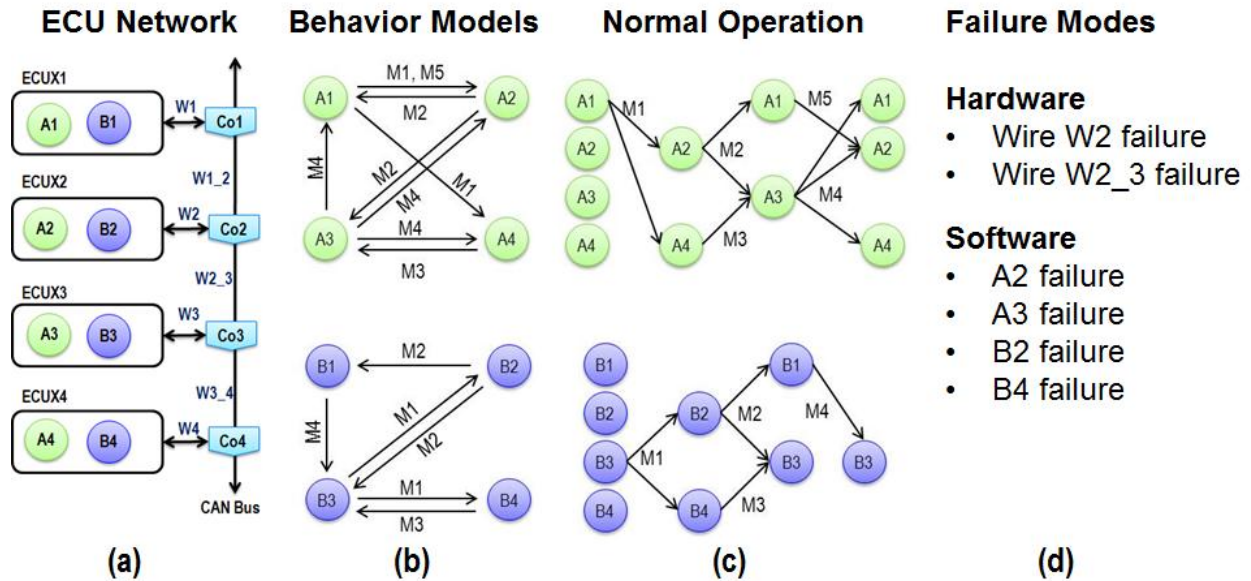


Figure 6. Simulation of communication network traffics: (a) shows 4 ECUs, wires and connectors with their topological connections; (b) shows behavior models of two features implemented on 4 ECUs; (c) shows normal operational of canonical dynamic message network for two features; (d) shows simulated hardware and software failure modes.

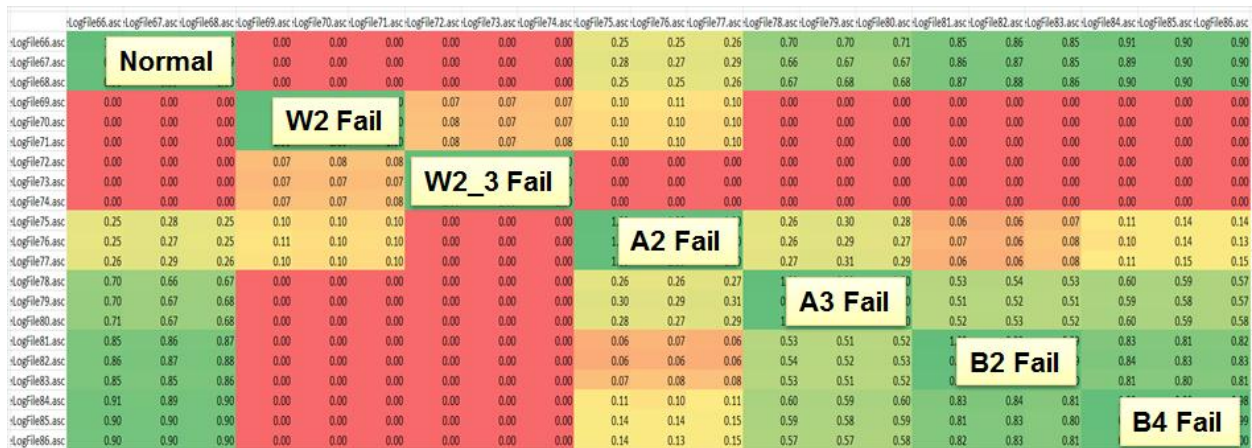


Figure 7. The resulting similarity matrix for net-motif distributions from the dynamic message constructed from simulated network traffic in Figure 6. Each row and each column represent one run of the simulation, and each mode of operation was simulated with 3 runs. The matrix is color-coded based on similarity score (green: similar, red: different). Some net-motif fault signatures (e.g. B2 Fail and B4 Fail) are fairly similar to those in normal operation, and some are quite different (e.g. W2 Fail and W2\_3 fail). Despite this mixed of similarity, our net-motif discriminator based on k-nearest neighbor can discriminate different mode of operations with 100% accuracy on new log files which it was not trained on.

9. CONCLUSION

The increasing complexity of in-vehicle electrical and electronic systems poses ever-growing challenges to diagnostics and prognostics. In 2011, we leveraged the latest advancements in Network Science to develop a system and method based on betweenness centrality to find good monitoring points for fault detection (Lu et al. 2011). In 2012, we address the challenge of emerging or cascading

faults in EES that are difficult to be addressed by traditional component-level diagnostics. We present the novel network-theoretic approach to detect, identify, and localize faults using both the structure of communication network (topological information) and message flow information over the communication network. The proposed method enables the characterization of normal operation and a-priori known faults across communication networks, which is beyond the current practice of individual ECUs point of

view. It also differs from conventional signature-based approach in that the mode of operations is classified at the level of modular interactions (motifs) rather than individual observation. It also enables the diagnostics of unknown or cascading failures emerging from unexpected operational environments. The system has been demonstrated on a simulated multi-ECU system in the CANoe environment, and shows promising results in fault diagnostics.

#### ACKNOWLEDGEMENT

The authors would like to thank Shengbing Jiang and Mark Howell for many valuable technical discussions. Special thanks also go to many of our colleagues in Engineering, including Doug Duddles, Sandeep Menon, Ken Orlando, Bob Schwabel, Lars Soderlund, Mike Sowa, and Natalie Wienckowski, for sharing the domain knowledge and shaping the research direction.

#### REFERENCES

- Bastian, M., Heymann, S., and Jacomy, M. (2009). Gephi: An Open Source Software for Exploring and Manipulating Networks, International AAAI Conference on Weblogs and Social Media.
- Cordella, L.P., Pasquale, F., Sansone, and C., Vento, M. (2004). A (Sub)Graph Isomorphism Algorithm for Matching Large Graphs, , IEEE Transaction on Pattern Analysis and Machine Intelligence, 26(10), 1367-1372.
- Djidjev, H., Sandine. G., and Storlie, C. (2011). Graph Based Statistical Analysis of Network Traffic, MLG.
- Lu, T.-C., Zhang, Y., Allen, D. L. and Salman, S. M. (2011). Design for Fault Analysis Using Multi-Partite, Multi-Attribute Between Centrality Measures, PHM.
- Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., and Alon, U. (2002), Network-motifs: Simple Building Blocks of Complex Networks, Science, vol. 298, no. 5594, pp. 824–827.
- Wang, T., Srivatsa, M., Agrawal, D., and Liu, L. (2009), Learning, Indexing, and Diagnosing Network Faults, KDD.
- Wernicke, S. (2006), Efficient Detection of Network Motifs, IEEE/ACM Transaction on Computational Biology and Bioinformatics, 3(4), 347-359.

#### BIOGRAPHIES

**Tsai-Ching Lu** received his M.S. degree in Computer Sciences from NYU in 1996, and his M.S. and Ph.D. degrees in Intelligent Systems from University of Pittsburgh in 2000 and 2003 respectively. He joined HRL Laboratories,

LLC, in 2003 and currently holds the position of Senior Research Staff Scientist. His research interests include probabilistic causal reasoning, complex systems, and network sciences with applications to behavior modeling of socio-technological systems. He received HRL's Distinguished Inventor Award in 2010 and GM contribution Award in 2011.

**Yilu Zhang** received his B.S., and M.S. degrees in Electrical Engineering from Zhejiang University, China, in 1994, and 1997, respectively; and his Ph.D. degree in Computer Science from Michigan State University in 2002. He joined the R&D center of General Motors Co. at Warren, Michigan in 2002, and currently holds a position of Staff Researcher. Dr. Zhang's research interests include statistical pattern recognition, machine learning, signal processing, and their applications such as integrated system health management and human machine interactions. Dr. Zhang is a Senior Member of IEEE. He is a winner of 2008 "Boss" Kettering Award – the highest technology award in GM – for his contribution to Connected Vehicle Battery Monitor, a remote vehicle diagnostics technology.

**David Allen** received his M.S. and Ph.D. degrees in Computer Science from the University of California, Los Angeles (UCLA) in 2001 and 2005 respectively. He joined HRL Laboratories, LLC, in 2006 and currently holds the position of Research Staff Scientist. His research interests include artificial intelligence, probabilistic reasoning, decision making under uncertainty, complex systems analysis, and network science; he has applied them to many domains including integrated system health management, cybersecurity, social network analysis, and behavior modeling. In 2010 he received HRL's Distinguished Inventor Award and in 2011 received a GM Contribution Award.

**Mutasim Salman** is a Lab. Group manager and a Technical Fellow in the Electrical, Controls and Integration Lab. of GM Research and Development Center. He has the responsibility of development and validation of algorithms for state of health monitoring, diagnosis, prognosis and fault tolerant control of vehicle critical systems. He pioneered the work on integrated chassis control in the late eighties that led to the production of GM "Industry First" Stabilitrak1 and then to Stabilitrak3. He had an extensive experience in hybrid vehicle, modeling, control and energy management strategies. He has several GM awards that includes 3 GM prestigious Boss Kettering, 3 McCuen, and 2 President and Chairman Awards. Mutasim received his bachelor's degree in Electrical Engineering from University of Texas at Austin; M.S. and PhD in Electrical Engineering with specialization in Systems and control from University of Illinois at Urbana- Champaign. He also has an Executive MBA. He is IEEE senior member. He holds 21 patents and has coauthored more than 42 refereed technical publications and a book. He joined the GM R&D staff in 1984.

# Condition-Based Maintenance with both Perfect and Imperfect Maintenance Actions

Phuc Do Van, Alexandre Voisin, Eric Levrat, and Benoit Iung

*Lorraine University, CRAN, CNRS UMR 7039*

*Campus Sciences BP 70239, 54506 Vandoeuvre, France*

*van-phuc.do@univ-lorraine.fr, alexandre.voisin@univ-lorraine.fr, eric.levrat@univ-lorraine.fr, benoit.iung@univ-lorraine.fr*

## ABSTRACT

This paper deals with a condition-based maintenance (CBM) model considering both perfect and imperfect maintenance actions for a deteriorating system whose condition is aperiodically monitored according to a remaining useful life (RUL) based-inspection policy. Perfect maintenance actions restore completely the system to the 'as good as new' state. Their related cost are however often high. Imperfect preventive maintenance restores partially the system with reduced maintenance cost. Nevertheless, it may however make the system more susceptible to future deterioration. The aim of the paper is to propose a CBM model which can help to construct optimal maintenance policies when both perfect and imperfect maintenance actions are possible. To illustrate the use of the proposed CBM model, a numerical example finally is introduced.

## 1. INTRODUCTION

Maintenance involves preventive and corrective actions carried out to retain a system in or restore it to an operating condition. Optimal maintenance policies aim to provide optimum system reliability/availability and safety performance at lowest possible maintenance costs, (Pham & Wang, 1996). In the literature, perfect maintenance actions (or replacement actions) which can restore the system operating condition to as good as new have been considered in various maintenance models. The implementation of perfect maintenance policies seems quite simple, however, perfect maintenance actions are often expensive. Imperfect maintenance implying that the system condition after maintenance is somewhere between the condition before maintenance and as good as new has grown recently as a popular issue to researchers as well as industrial engineering, see for example (Castro, 2009; Kijima, Morimura, & Suzuki, 1988; Labeau & Segovia, 2010; Levitin & Lisnianski, 2000; Nakagawa & Yasui, 1987; Liu & Huang,

2010). From a practical point of view, imperfect maintenance can describe a large kinds of realistic maintenance actions (Pham & Wang, 1996). Furthermore, imperfect maintenance is usually cheaper than perfect maintenance. Various methods and optimal policies for imperfect maintenance are summarized and discussed in (Pham & Wang, 1996; Wu & Zuo, 2010). In such maintenance models, preventive maintenance decision is however based on the system age and on the knowledge of the statistical informations on the system lifetime. As a consequence, the realistic operating conditions of the system over time can not be taken into account.

To face this issue, condition-based maintenance (CBM), for which preventive maintenance decision is based on the observed system condition, is recently introduced. Thank to rapid development of monitoring equipments which can provide accurately information about the system condition over time, CBM becomes nowadays an interesting approach for maintenance optimization. Various CBM policies have been proposed and applied for many industrial systems, see for example (Ghasemi, Yahcoub, & Ouali, 2007; Grall, Dieulle, Bérenguer, & Roussignol, 2002; Neves, Santiago, & Maia, 2011; Noortwijk, 2009; Tan, Cheng, Guo, & Gong, 2010). It is recently shown in (Meier-Hirmer, Riboulet, Sourget, & Roussignol, 2008; Nicolai, Frenk, & Dekker, 2009; Ponchet, Fouladirad, & Grall, 2011) that CBM is specially suited to imperfect maintenance since according to the observed condition of the system, an optimal maintenance action represented by an optimal intervention gain is preventively carried out. However, in such maintenance policies, only imperfect preventive or imperfect repair actions are considered and the system is assumed to be imperfectly maintained an infinite number of times. From a practical point of view, this assumption may not always be relevant since, in variety of engineering and service applications, systems can be maintained only a limited number of times due to technical or economical reasons (Kurt & Kharoufeh, 2010). Furthermore, as mentioned in (Nicolai et al., 2009), each imperfect maintenance action may make the system more susceptible to future deterioration. To this end, a fixed number of allowable imperfect

Phuc Do Van et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

maintenance actions is introduced in maintenance models in (Kurt & Kharoufeh, 2010; Do Van & Berenguer, 2012) and considered as a decision parameter. However, the value of variable is arbitrary chosen and they do not describe how the imperfect repair actions affect the deterioration evolution of the system.

The aim of this paper is to propose a CBM model considering both perfect and imperfect maintenance actions for a deteriorating system. Imperfect maintenance is investigated with both positive and negative sides. Positive impact means that it can reduce the deterioration level of the system with reduced maintenance cost. Negative impact implies that each imperfect preventive action may accelerate the speed of the system's deterioration process. Moreover, in CBM practice, inspections are usually performed at regular intervals. However, it may not be always profitable to systematically inspect the system, especially when the inspection procedure is costly. The present paper proposes to use an aperiodic inspection policy which is based on the residual useful life (RUL) of the system, see (Cui, Xie, & Loh, 2004; Gebraeel, Lawley, Li, & Ryan, 2005; Yang & Klutke, 2001).

This paper is organized as follows. Section 2 is devoted to the description of the system characteristics and related assumptions. Imperfect maintenance actions and their related cost model are also described and discussed. Section 3 focuses on the proposed imperfect maintenance policy. To illustrate the proposed maintenance policy, a simple numerical example is introduced in Section 4. Some numerical results are in addition discussed here. Finally, the last section presents the conclusions drawn from this work.

## 2. MODEL DESCRIPTION AND ASSUMPTIONS

### 2.1. General assumptions

Consider a system in which the condition (deterioration level) at time  $t$  can be summarized by an observable random scalar variable  $X_t$ . This random variable can represent for example a crack length or defect products. In the absence of repair or replacement actions, the evolution of the system deterioration is assumed to be strictly increasing. The process  $(X_t)_{t \geq 0}$  is then an increasing stochastic process. Moreover, we suppose that the following assumptions are verified.

- The initial state  $X_0$  is 0;
- The system is failed if its deterioration level is greater than a level  $L$ . The threshold  $L$  can be seen as a deterioration level which must not be exceeded for economical or security reasons. We assume that (i) inspection and maintenance actions can be only performed at discrete times and (ii) the system failure is self-announcing.

The system degradation behavior and corresponding states are illustrated in Fig. 1.

To avoid failure occurrence of the system, preventive maintenance is considered. It is assumed that both imperfect and

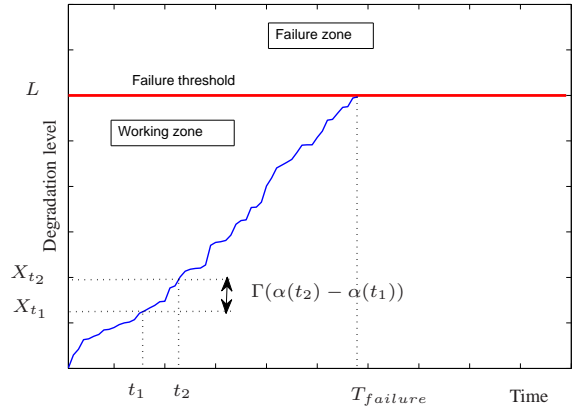


Figure 1. Illustration of the system degradation evolution and its state.

perfect preventive maintenance actions, which incur respectively perfect maintenance cost  $C_p$  and imperfect maintenance cost  $C_p^k$ , are possible.

### 2.2. Deterioration modelling

Gamma processes have been widely used to describe the degradation of systems (Noortwijk, 2009; Grall et al., 2002; Do Van & Berenguer, 2012). A characteristic of this process is that it is clearly monotone increasing which is the behavior observed in most physical deterioration processes. Moreover, its paths are discontinuous and it can be thought as the accumulation of an infinite number of small shocks. Following this spirit, it is assumed that the deterioration of the system between the  $k$ th and the  $(k+1)$ th maintenance actions evolves like a Gamma stochastic process  $(\tilde{X}_t)_{t \geq 0}$ , with the following characteristics:

- $\tilde{X}_0 = X^k$ , ( $X^k$  represents the deterioration level of the system after the  $k$ th maintenance action);
- $(\tilde{X}_t)_{t \geq 0}$  has independent increments;
- for all  $0 \leq l < t$ , the random increment  $\tilde{X}_t - \tilde{X}_l$  follows a Gamma probability density (pdf) with shape parameter  $\alpha_k(t-l)$  and scale parameter  $\beta$ :

$$f_{\alpha_k(t-l), \beta}(x) = \frac{1}{\Gamma(\alpha_k(t-l))} \beta^{\alpha_k(t-l)} x^{\alpha_k(t-l)-1} e^{-\beta x} \mathcal{I}_{\{x \geq 0\}},$$

where:

- $\mathcal{I}_{\{x \geq 0\}}$  is an indicator function  $\mathcal{I}_{\{x \geq 0\}} = 1$  if  $x \geq 0$ ,  $\mathcal{I}_{\{x \geq 0\}} = 0$  otherwise;
- $\alpha_k = v_k/\beta$  with  $v_k$  being the mean deterioration speed of the system between the  $k$ th and the  $(k+1)$ th maintenance actions.

After a corrective or perfect preventive maintenance action, the system becomes as good as new (the deterioration level is set to 0 and the deterioration behavior evolves with time ac-

cording to the nominal speed  $v_0 = \alpha_0/\beta$ ). Imperfect maintenance actions can reduce the system's deterioration level with reduced maintenance costs. However, as mentioned in (Nicolai et al., 2009; Kurt & Kharoufeh, 2010), imperfect maintenance actions may affect the evolution of the system's deterioration process. The impacts of imperfect maintenance actions will be described in the next section.

### 2.3. Imperfect maintenance actions and related costs

#### 2.3.1. Impact of imperfect actions on the deterioration level

It is shown in the literature that maintenance gains, defined as the reductions of the deterioration level of the system due to imperfection maintenance actions, could be random, see for instance (Castro, 2009; Meier-Hirmer et al., 2008; Do Van & Berenguer, 2012).

In this way, if the  $k$ th imperfect maintenance action is performed at inspection time  $T_i$ , the intervention gain is then assumed to be described by a continuous random variable  $Z^k$ .  $Z^k$  is restricted,  $0 \leq Z^k \leq X_{T_i}$  where  $X_{T_i}$  is the deterioration level of the system at  $T_i$ . In fact, it is shown in (Do Van & Berenguer, 2012) that  $Z^k$  can be distributed according to a truncated normal distribution with density:

$$g_{\mu,\sigma,a,b}(x) = \frac{\frac{1}{\sigma}\phi\left(\frac{x-\mu}{\sigma}\right)}{\Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)}\mathcal{I}_{[a,b]}(x), \quad (1)$$

where:

- $\mathcal{I}_{[a,b]}(x) = 1$  if  $a \leq x \leq b$  and  $\mathcal{I}_{[a,b]}(x) = 0$  otherwise;
- $\phi(\xi) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}\xi^2)$  is the probability density function of the standard normal distribution and  $\Phi(\cdot)$  is its cumulative distribution function;
- $\mu = X_{T_i}/2$  and  $\sigma = X_{T_i}/6$ ;
- $a = \mu - 3\sigma = 0$  and  $b = \mu + 3\sigma = X_{T_i}$ ,

According to this distribution, it is clear that  $a \leq Z_{T_i} \leq b$ , i.e.  $0 \leq Z^k \leq X_{T_i}$  is satisfied. The mean intervention gains is  $\mathbb{E}(Z^k) = \mu$  and the variance is  $\mathbb{V}\mathbb{A}\mathbb{R}(Z^k) = 0.973\sigma^2$ , see (Ross, 1996). Thank to the imperfect preventive action, the deterioration level of the system after maintenance is set to  $X^k = X_{T_i} - Z^k$ . The illustration of the system deterioration evolution and random maintenance gain are shown in Fig. 2.

#### 2.3.2. Impact of imperfect actions on the deterioration speed

To model the impact of imperfect actions on the evolution deterioration of the system, it is assumed in this work that each imperfect preventive action affects the speed of the system deterioration process. This can be found in variety of business sectors, e.g. removing several components for maintenance actions may accelerate the deterioration evolution of other components; spare parts may be reusable components

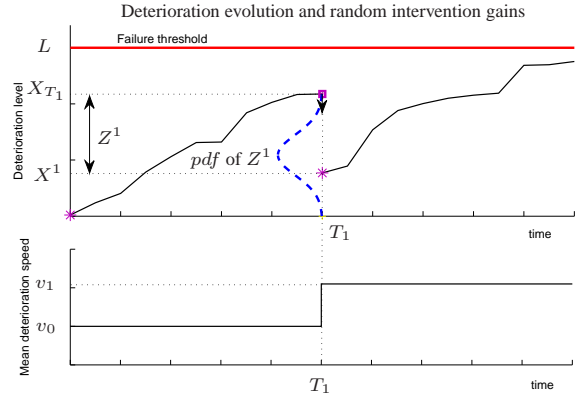


Figure 2. Illustration of deterioration evolution and impacts of imperfect maintenance.

or low quality components, as a consequence, after maintenance the deterioration level of the system can be reduced however the deterioration speed may be increased. The impact of an imperfect maintenance action on the system deterioration speed can be described by non-negative continuous random variable  $\epsilon$  which follows an exponential distribution with density probability:

$$h(x) = \gamma e^{-\gamma x} \mathcal{I}_{\{x \geq 0\}},$$

where  $\gamma$  is a non negative real number. The mean value of  $\epsilon$  is  $\mathbb{E}[\epsilon] = \frac{1}{\gamma}$ .

By this modelling, if the  $k$ th maintenance action is a corrective or perfect preventive maintenance, the mean deterioration speed of the system after maintenance is reset to  $v_k = v_0 = \alpha_0/\beta$ . If the  $k$ th maintenance action is an imperfect preventive one, the mean deterioration speed of the system after maintenance is:

$$v_k = v_{k-1} + \epsilon. \quad (2)$$

An example of increasing of the deterioration speed due to an imperfect maintenance action is illustrated in Fig. 2. A case study on the sensitivity to the effect of imperfect maintenance actions will be discussed in Section 4.

#### 2.3.3. Imperfect preventive maintenance cost

In general, each maintenance action incurs a cost and an imperfect maintenance action often incurs a reduced maintenance cost, namely imperfect maintenance cost, which may be independent of maintenance gain and bounded by perfect maintenance cost, see for instance (Castro, 2009; Labeau & Segovia, 2010; Meier-Hirmer et al., 2008). From a practical point of view, in most cases, the quality of the maintenance action increases with the level of resources allocated to it, and hence with its cost, see (Lie & Chun, 1986; Liu & Huang, 2010; Mettas, 2000). To model imperfect maintenance ac-

tions in the context of deteriorating systems, the degradation improvement factor, defined as the ratio of the improvement gain divided by the deterioration level of the system before maintenance, has been recently introduced, see (Do Van & Berenguer, 2012). Based on the improvement factor, imperfect maintenance costs can be evaluated and considered as a function of the improvement factor. In this way, it is assumed in this work that when the  $k$ th imperfect preventive action is performed at inspection time  $T_i$ , we have to pay a maintenance cost which is defined as:

$$C_p^k = C_p^0 \cdot u(T_i)^\eta, \quad (3)$$

where:

- $u(T_i) = \frac{Z^k}{X_{T_i}}$  is the degradation improvement factor;
- $C_p^0$  is imperfect preventive cost incurred when the deterioration level of the system is reduced to 0 with imperfect maintenance action. This cost is usually lower than a perfect preventive cost ( $C_p^0 \leq C_p$ );
- $\eta$  is a non-negative real number.

According to this cost model, different kinds of maintenance cost function can be found depending on the value of  $\eta$ . More precisely:

- when  $\eta = 0$ , imperfect maintenance cost is constant ( $C_p^k = C_p^0$ );
- when  $0 < \eta < 1$ , imperfect maintenance cost  $C_p^k$  is a concave function: the maintenance cost increases more than the improvement gain when performing the maintenance;
- when  $\eta = 1$ ,  $C_p^k$  is a linear function which implies that the maintenance cost is proportional to the improvement level gain;
- when  $\eta > 1$ ,  $C_p^k$  is a convex function: the maintenance cost increases less than the improvement gain.

Figure 3 illustrates these three different shapes of the imperfect maintenance cost function. A case study of the proposed imperfect maintenance policy with different kinds of imperfect maintenance cost functions will be presented in Section 4.

### 3. MAINTENANCE POLICY

In the framework of CBM optimization, a maintenance policy relies essentially on two main decisions: when to take (preventively/correctively) maintenance actions and when to inspect. The system degradation can be used to make the decision on the inspection time and on the maintenance action to be performed, (Grall et al., 2002; Noortwijk, 2009; Do Van & Berenguer, 2010).

The maintenance decision is herein based on both the system deterioration level at inspection time and the potential evo-

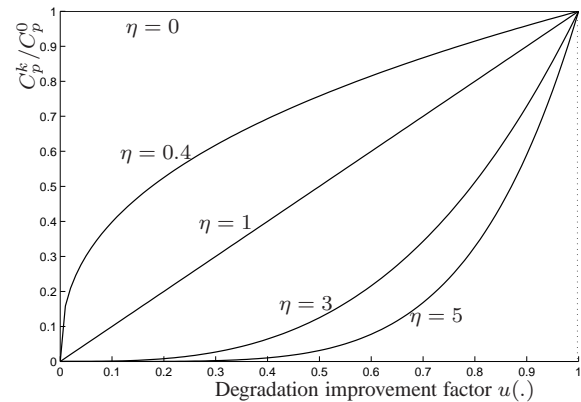


Figure 3. Illustration of imperfect maintenance cost function.

lution of the system's deterioration process. More precisely, according to the degradation level  $X_{T_i}$  at inspection time  $T_i$ , the maintenance decision is the following:

- if  $X_{T_i} < M$ , the system is in a working state, no maintenance action is performed.  $M$  is called the preventive maintenance threshold and it is a decision variable to be optimized.
- if  $L > X_{T_i} \geq M$ , the system is still functioning, however its deterioration level is considered as "high". A preventive maintenance action is immediately carried out. Without loss of generality, it is assumed that this preventive maintenance action is the  $k$ th preventive maintenance action from the last perfect maintenance of the system. If  $k = K$  ( $K$  is called the imperfect threshold and it is a decision variable to be optimized), the  $k$ th preventive maintenance action is a perfect one. Contrarily, if  $k < K$  the  $k$ th preventive maintenance action is an imperfect one. This imperfect preventive maintenance action may not restore completely the system but it leads the system to be a better state at which the degradation level is lower or equal to the current deterioration level of the system. The impacts of imperfect preventive maintenance actions and their related cost have been described in Section 2.3.
- if  $X_{T_i} \geq L$ , the system is failed, then a corrective replacement action is performed and a cost  $C_c$  is incurred. An additional cost is incurred by the time  $d^j$  elapsed in the failed state at a unavailability cost rate  $C_d$  which may correspond to, for example, production loss per unit of time. After a corrective maintenance action, the system is considered as good as new.

We assume finally that maintenance durations are neglected and all the necessary maintenance resources to execute preventive or corrective maintenance actions are always available. The decision process of the maintenance policy is illustrated in Fig. 4.



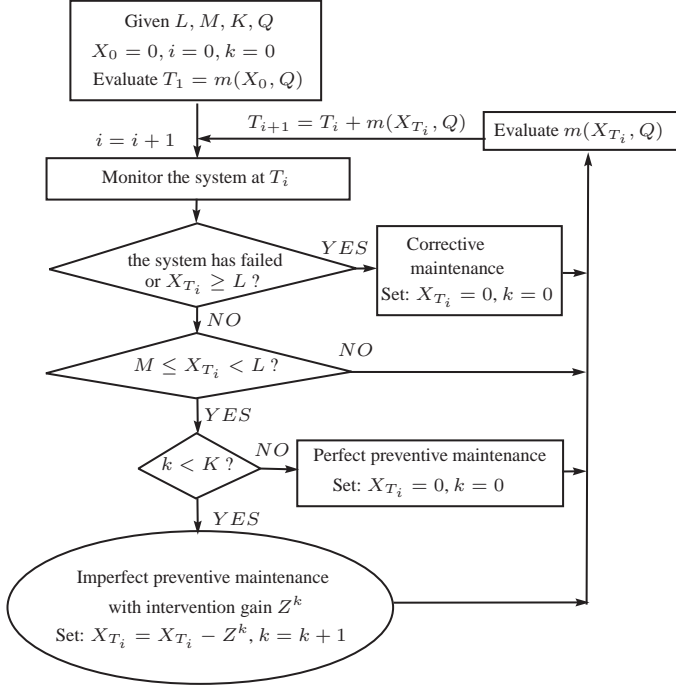


Figure 4. Decision process of the maintenance policy.

### 3.1. RUL based inspection

Different inspection policies, which aim to optimize the inter-inspection interval length, have been introduced in literature. In fact, the inter-inspection interval length can be fixed regardless of the degradation level, e.g. (Ponchet et al., 2011), or aperiodic and deteriorating-dependant via an inspection scheduling linear (Grall et al., 2002), or non-linear (Barker & Newby, 2009) function with respect to the deterioration level. Residual Useful Life (RUL) based inspection has been recently introduced, see (Cui et al., 2004; Gebraeel et al., 2005; Yang & Klutke, 2001). The latter seems very promising especially in the context of condition-based maintenance.

The main idea of the RUL based inspection is that the next inspection time is chosen such that the probability of the failure of the system before the next inspection remains lower than a limit  $Q$  ( $0 < Q < 1$  and it is a decision variable to be optimized). If we let  $T_i$  denote the time at which the system is inspected, the corresponding degradation level of the system is  $X_{T_i}$ , the next inspection time is then determined by:

$$T_{i+1} = T_i + m(X_{T_i}, Q), \quad (4)$$

with,

$$m(X_{T_i}, Q) = \sup\{\Delta T : \mathbb{P}(X_{T_i+\Delta T} \geq L | X_{T_i}) \leq Q\}, \quad (5)$$

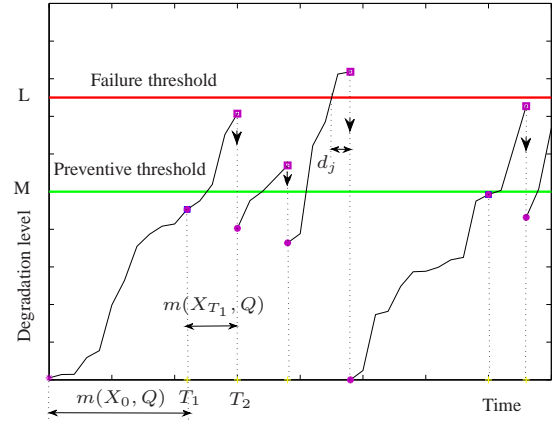


Figure 5. Illustration of degradation behavior and maintenance policy.

where:

$$\begin{aligned} \mathbb{P}(X_{T_i+\Delta T} \geq L | X_{T_i}) &= \mathbb{P}(X_{T_i+\Delta T} - X_{T_i} \geq L - X_{T_i}) \\ &= \int_{L-X_{T_i}}^{\infty} f_{\alpha_k \Delta T, \beta}(x) dx \\ &= 1 - \int_0^{L-X_{T_i}} f_{\alpha_k \Delta T, \beta}(x) dx. \end{aligned} \quad (6)$$

It is clear that  $m(X_{T_i}, Q)$  depend on the current degradation level of the system, the failure threshold  $L$  and the parameter  $Q$  (Gebraeel et al., 2005). The illustration of inter-inspection interval is shown in Fig. 5. The integration of  $m(X_{T_i}, Q)$  in the maintenance decision process is illustrated in Fig. 4.

Finally, the inspections are assumed to be instantaneous, perfect and non-destructive. When an inspection is performed, a cost  $C_i$  is incurred.

According to this inspection policy, the reliability of the system between two inspection times interval remains higher or equal to  $(1 - Q)$ . This means that the proposed maintenance policy can provide an optimal maintenance planning with a given reliability level. From a practical point of view, this result seems to be very interesting since in many industrial systems, the reliability of the system may be an important constraint due to technical and/or economical reasons, see for example (Do Van, Vu, Barros, & Berenguer, 2012).

### 3.2. Optimization of the maintenance policy

To evaluate the performance of the maintenance policy, the long-run expected maintenance cost rate including the unavailability cost is used herein as the main criterion.

According to the proposed model, the cumulative mainte-

nance cost at time  $t$  is:

$$C^t(M, K, Q) = C_i \cdot N_i(t) + \sum_{k=1}^{N_{ip}(t)} C_p^k + N_p(t)C_p + \sum_{j=1}^{N_c(t)} C_c + C_d \cdot d(t), \quad (7)$$

where:  $N_i(t)$ ,  $N_p(t)$ ,  $N_{ip}(t)$ ,  $N_c(t)$  are respectively the number of inspections, of perfect preventive maintenance, of imperfect maintenance and of corrective replacement in  $[0, t]$ ;  $d(t)$  is the the total time passed in a failed state in  $[0, t]$ .

By using the renewal theory (Ross, 1996), the long run expected maintenance cost per unit of time is:

$$C^\infty(M, K, Q) = \lim_{t \rightarrow \infty} \frac{C^t(M, K, Q)}{t}. \quad (8)$$

Stochastic Monte Carlo simulation is used to evaluate this cost criterion. The optimal values of the decision parameters  $(M, K, Q)$  are obtained by minimizing the expected maintenance cost rate, e.i.,

$$C^\infty(M^*, K^*, Q^*) = \min_{M, K, Q} \{C^\infty(M, K, Q), 0 \leq M < L, 0 \leq K, 0 < Q < 1\}. \quad (9)$$

(10)

According the maintenance policy, the interest of imperfect maintenance policy is represented by  $K^*$ . When  $K^* = 0$ , no imperfect action is considered, the proposed maintenance policy becomes a perfect one whose performance is investigated and proved in (Grall et al., 2002; Do Van & Berenguer, 2010). Besides, the higher  $K^*$  is, the more the interests of imperfect actions are.

#### 4. NUMERICAL EXAMPLE

The purpose of this section is to show how the proposed maintenance policy can be used in maintenance optimisation through a simple example whose characteristics are described in Section 2.

Consider a deteriorating system in which its degradation behavior, when no maintenance is carried out, is assumed to be described by a Gamma process with scale parameter  $\alpha_0 = 1$  and shape parameter  $\beta = 1$ . If the degradation of the system exceeds the failure threshold  $L = 20$ , the system is failed. Both corrective and perfect maintenances can restore completely the system to the 'as good as new' state. Besides, the deterioration level of the system can be improved by imperfect maintenance actions which however may affect the deterioration speed, see again Section 2.2. Table 1 reports the data related to inspection, maintenance costs, unavailability cost rate (all costs are given in arbitrary units) and the impact of imperfect maintenance actions on the deterioration speed.

$C_i$	$C_c$	$C_p$	$C_p^0$	$C_d$	$\eta$	$\gamma$
10	100	90	70	20	3	0.2

Table 1. Data of costs and impact of imperfect actions

To evaluate the mean maintenance cost per unit of time, the simulations are done on a very large interval of time so that  $t$  is assumed to tend to infinity. In order to find the optimal decision parameters  $(M, K, Q)$ , the average of maintenance cost per unit of time  $C^\infty(M, K, Q)$  is evaluated with different values of  $M$  ( $0 \leq M < L$ ),  $K$  ( $K \geq 0$ ) and  $Q$  ( $0 < Q < 1$ ) by using Equation (8). A numerical optimization scheme is used, the optimum values of the decision parameters are  $M^* = 14$ ,  $K^* = 4$  and  $Q^* = 0.10$  for an optimal cost rate  $C^\infty(M^*, K^*, Q^*) = 5.15$ . To compare with a perfect maintenance policy (only the maintenance cost criterion is herein used),  $K$  is set to be 0 (as mentionned above, when  $K = 0$ , the proposed imperfect maintenance policy becomes the perfect RUL based maintenance policy which seems to be an efficient policy in the framework of perfect condition-based maintenance, (Grall et al., 2002; Do Van & Berenguer, 2010)), the minimum average maintenance cost rate is then 6.23 which is much more higher than the result obtained by the imperfect policy. Moreover, the sensitivity with respect to the number of imperfect actions within a life cycle is sketched in Fig.6. The results show that when  $K \leq 4$ , the maintenance cost rate increases quickly if  $K$  is close to 0 and when  $K \geq 4$  the maintenance cost rate increases slowly with respect to the increasing of  $K$ . According to these results, it is clear that imperfect maintenance actions seem to be more appropriate than perfect ones.

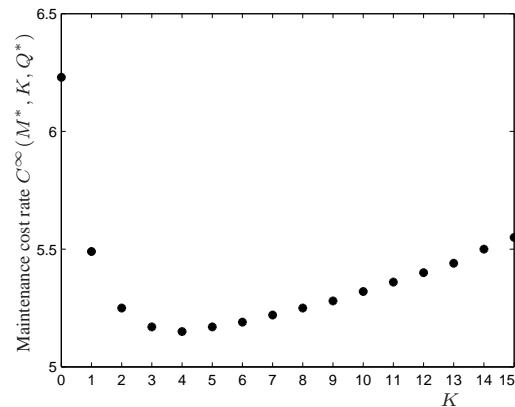


Figure 6. Mean maintenance cost rate as a function of  $K$ .

#### 4.1. Sensitivity analysis to the imperfect maintenance cost

The performance of imperfect maintenance actions may depend on their related cost which are herein by characterized by  $\eta$ , see Equation (3). Table 2 reports the optimum values of  $M$ ,  $K$  and  $Q$  and the minimum value of  $C^\infty(M, K, Q)$  for different values of  $\eta$ .

$\eta$	Optimal decision parameters			$C^\infty(M^*, K^*, Q^*)$
	$M^*$	$K^*$	$Q^*$	
0	16	0	0.15	6.23
0.4	16	0	0.15	6.23
1	16	0	0.15	6.23
2	14	2	0.13	5.72
3	14	4	0.10	5.15
5	12	6	0.08	4.44

Table 2. Optimal maintenance policy with a given  $\eta$

The results show that when  $\eta \leq 1$ , the maintenance cost rate remains unchanged, the optimal maintenance policies corresponds to a perfect maintenance policy ( $K = 0$ ). However, the maintenance cost rate decreases dramatically when  $\eta > 1$ . This means that, in this case study, the imperfect maintenance cost has a significant influence on the performance of the imperfect maintenance policy.

#### 4.2. Sensitivity analysis to the impact of imperfect maintenance on the deterioration speed

To analyze the impact of imperfect maintenance actions, different values of  $\gamma$  are considered. For each value of  $\gamma$ , a maintenance policy characterized by the decision parameters ( $M, K, Q$ ) is optimally found using Equation (9) and the obtained results are reported in Table 3.

$\gamma$	Optimal decision parameters			$C^\infty(M^*, K^*, Q^*)$
	$M^*$	$K^*$	$Q^*$	
0.05	12	13	0.06	4.31
0.1	13	5	0.07	4.78
0.2	14	4	0.10	5.15
0.5	15	2	0.11	5.69
1	15	1	0.12	5.99
2	16	0	0.15	6.23
3	16	0	0.15	6.23

Table 3. Optimal maintenance policy with a given  $\gamma$

The results show that when the impact of imperfect actions on the deterioration speed of the system is small, the maintenance cost is relatively low. Oppositely, when the impact of imperfect actions on the deterioration speed is large,  $\gamma \geq 2$ , the maintenance cost is high and the proposed imperfect maintenance becomes a perfect policy ( $K = 0$ ). This can be explained by the fact that imperfect maintenance actions are cheaper than perfect ones however they are indirectly penalized by their negative influence on the deterioration speed of the system. As a consequence, when their effect on the system's deterioration process is small, they become more appropriate and when this effect is large, perfect maintenance actions seems to be a better choice.

#### 5. CONCLUSIONS

In this work, a condition-based maintenance (CBM) model considering both perfect and imperfect maintenances for a deteriorating system is proposed. Imperfect maintenance ac-

tions characterized by random intervention gains are studied and discussed with different types of their related cost which may be a concave, linear or convex function with respect to intervention gain. The impact of imperfect actions on the deterioration speed of the system is also investigated. Moreover, an adaptive maintenance policy with the residual useful life (RUL) based inspections is proposed. The proposed imperfect maintenance policy may optimally become a perfect policy (Grall et al., 2002; Do Van & Berenguer, 2010) for several cases, e.g., when the imperfect cost is high or/and imperfect actions largely affect the deterioration speed of the system. Finally, the performance of the proposed policy is illustrated and discussed through an example of deteriorating system. Different sensitivity analysis are investigated to show the interest of the proposed CBM model.

Our future research work will focus on the detailed development of the impacts of imperfect maintenance actions in the framework of CBM. Furthermore, the application of the proposed CBM model for realistic industrial systems will be investigated.

#### NOMENCLATURE

$C_i$	inspection cost
$C_p$	perfect (replacement) preventive maintenance cost
$C_p^k$	cost of the $k$ th imperfect maintenance action
$C_c$	corrective (replacement) maintenance cost
$C_d$	unavailability cost rate of the system
$d(t)$	total time passed in failed state in $[0, t]$
$C^\infty(.)$	long-run expected maintenance cost rate
$L$	failure threshold
$K$	imperfect maintenance threshold
$M$	preventive maintenance threshold
$m(.)$	inter-inspection length
$N_i(t)$	number of inspection in $[0, t]$
$N_p(t)$	number of perfect preventive maintenance in $[0, t]$
$N_{ip}(t)$	number of imperfect preventive maintenance in $[0, t]$
$N_c(t)$	number of corrective maintenance in $[0, t]$
$Q$	failure probability between two inspection times
$T_i$	$i$ th inspection time
$X_t$	system deterioration level at time $t$
$Z^k$	$k$ th intervention gain
$u(.)$	degradation improvement factor
$v_k$	mean deterioration speed after the $k$ th maintenance action
$\alpha_0, \beta$	scale and shape parameters of the deterioration process when the system is as good as new
$\alpha_k$	scale parameter of deterioration process after the $k$ th imperfect maintenance action
$\eta$	a non-negative real number
$\gamma$	non-negative real number and represents the impact of imperfect maintenance actions on the deterioration speed of the system

## REFERENCES

- Barker, C., & Newby, M. (2009). Optimal non-periodic inspection for a multivariate degradation model. *Reliability Engineering and System Safety*, 94, 33–43.
- Castro, I. T. (2009). A model of imperfect preventive maintenance with dependent failure modes. *European Journal of Operational Research*, 196(1), 217 – 224.
- Cui, L., Xie, M., & Loh, H.-T. (2004). Inspection schemes for general systems. *IIE Transactions*, 39(9), 817–825.
- Do Van, P., & Berenguer, C. (2010). Condition based maintenance model for a production deteriorating system. In *Conference on control and fault-tolerant systems (sys-tol'10), 6-8 september 2010, nice, france*. IEEE.
- Do Van, P., & Berenguer, C. (2012). Condition-based maintenance with imperfect preventive repairs for a deteriorating production system. *Reliability and Quality Engineering International*, DOI: 10.1002/qre.1431.
- Do Van, P., Vu, H. C., Barros, A., & Berenguer, C. (2012). Grouping maintenance strategy with availability constraint under limited repairmen. In *8th ifac international symposium on fault detection, supervision and safety for technical processes, safeprocess-2012, 29-31 august 2012, mexico city, mexico*.
- Gebraeel, N., Lawley, M., Li, R., & Ryan, J. (2005). Residual-life distributions from components degradation signals : A bayesian approach. *IIE Transactions*, 37, 543–557.
- Ghasemi, A., Yahcouth, S., & Ouali, M. (2007). Optimal condition based maintenance with imperfect information and the proportional hazards model. *International Journal of Production Research*, 4, 989–1012.
- Grall, A., Dieulle, L., Bérengruer, C., & Roussignol, M. (2002). Continuous-time predictive-maintenance scheduling for a deteriorating system. *IEEE Transactions On Reliability*, 51, 141–150.
- Kijima, M., Morimura, H., & Suzuki, Y. (1988). Periodical replacement problem without assuming minimal repair. *European Journal of Operational Research*, 37(2), 194–203.
- Kurt, M., & Kharoufeh, J. (2010). Optimally maintaining a markovian deteriorating system with limited imperfect repairs. *European Journal of Operational Research*, 205, 368–380.
- Labeau, P.-E., & Segovia, M.-C. (2010). Effective age models for imperfect maintenance. *Journal of Risk and Reliability*, 225, 117–130.
- Levitin, G., & Lisnianski, A. (2000). Optimization of imperfect preventive maintenance for multi-state systems. *Reliability Engineering and System Safety*, 67, 193–203.
- Lie, C. H., & Chun, Y. (1986). An algorithm for preventive maintenance policy. *IEEE Transactions On Reliability*, 35(1), 71 – 75.
- Liu, Y., & Huang, H. (2010). Optimal selective maintenance strategy for multi-state systems under imperfect maintenance. *IEEE Transactions On Reliability*, 59(2), 356 – 367.
- Meier-Hirmer, C., Riboulet, G., Sourget, F., & Roussignol, M. (2008). Maintenance optimisation for a system with a gamma deterioration process and intervention delay: application to track maintenance. *Journal of Risk and Reliability*, 223, 189–198.
- Metts, A. (2000). Reliability allocation and optimization for complex systems. In *Ieee proceedings of the annual reliability and maintainability symposium* (pp. 216 – 221).
- Nakagawa, T., & Yasui, K. (1987). optimal policies for a system with imperfect maintenance. *IEEE Transactions On Reliability*, R-36(5), 631–633.
- Neves, M. L., Santiago, L., & Maia, C. (2011). A condition-based maintenance policy and input parameters estimation for deteriorating systems under periodic inspection. *Computers and Industrial Engineering*, 61, 503–511.
- Nicolai, R. P., Frenk, J., & Dekker, R. (2009). Modelling and optimizing imperfect maintenance of coatings on steel structures. *Structural Safety*, 31, 234 – 244.
- Noortwijk, J. van. (2009). A survey of the application of Gamma processes in maintenance. *Reliability Engineering and System Safety*, 94, 2–21.
- Pham, H., & Wang, H. (1996). Imperfect maintenance. *European Journal of Operational Research*, 94, 425–438.
- Ponchet, A., Fouladirad, M., & Grall, A. (2011). Maintenance policy on a finite time span for a gradually deteriorating system with imperfect improvements. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 225(2), 105–116.
- Ross, S. (1996). *Stochastic processes*. New York, John Wiley & Sons, Inc.
- Tan, L., Cheng, Z., Guo, B., & Gong, S. (2010). Condition-based maintenance policy for gamma deteriorating systems. *Journal of Systems Engineering and Electronics*, 21, 57–61.
- Wu, S., & Zuo, J. M. (2010). Linear and nonlinear preventive maintenance models. *IEEE Transactions On Reliability*, 59(1), 242 – 249.
- Yang, Y., & Klutke, G.-A. (2001). A distribution-free lower bound for availability of quantile-based inspection schemes. *IEEE Transactions On Reliability*, 50, 419–421.

## BIOGRAPHIES

**Phuc DO VAN** is currently assistant professor at Lorraine University, Research Centre for Automatic Control (CRAN CNRS UMR 7039), France. He received his Ph.D. in Systems Optimisation and Dependability in 2008 from Troyes University of Technology (France) where he held an assistant pro-

fessor position from 2009 to 2011. His research interests include stochastic modeling of systems deterioration, optimization of maintenance policies (condition-based maintenance, prognostics for maintenance decision-making, opportunistic and dynamic grouping maintenance), reliability importance measures and their related applications.

**Alexandre VOISIN** was born in Metz, France, in 1969, obtained an engineering degree in Electrical Engineering in 1992. In 1999, he received his Ph.D degree in Electrical Engineering from the Lorraine University. He is currently assistant professor at the Lorraine University. His primary research were in the field of fuzzy logic and information processing where he applied these techniques to subjective evaluation in the area of car seat comfort.

Since 2003 he is involved in a maintenance project, managed by Pr. B. Iung. His research deals with dependability, maintenance decision in a proactive maintenance strategy, prognostics and monitoring, e-maintenance. He is member of French and International projects/groups on e-maintenance such as the CNRS MACOD working group (Modeling and Optimization of Distributed vs. Collaborative Maintenance), the French scientific interest group 3SGS on "Dependability of Complex Systems" in the project DEPRADDEM (Degradation and Prognosis Modeling for Maintenance Decision Making), the French project BMCI (Condition monitoring for maintenance and Piloting of naval systems), the European Integrated Project DYNAMITE (Dynamic Decision in Maintenance), and the international project DEPEND-IMPRO (Modeling Policies for the improvement of Production Systems' Dependability). He is involved in industrial projects with EDF, DCN, ALSTOM. His main research interests deal with prognostic, maintenance, multi-criteria decision making, data analysis, subjective evaluation.

**Eric LEVRAT** received his Ph.D. in 1989 from the University H. Poincaré Nancy 1, where he currently holds the position of an assistant professor. He has been researcher at the Research Centre for Automatic Control of Nancy since 1990. Since 2003 he is involved in maintenance area, his research deals with dependability, maintenance decision in a proactive maintenance strategy, maintenance organisation, e-maintenance. He is member of French and International

projects/groups on e-maintenance such as the CNRS MACOD working group (Modelling and Optimisation of Distributed vs. Collaborative Maintenance), the French scientific interest group 3SGS on "Dependability of Complex Systems", where he's leader of the project DEPRADDEM2 (Degradation and Prognosis Modelling for Maintenance Decision Making), the Integrated Project DYNAMITE (Dynamic Decision in Maintenance), and the international project DEPEND-IMPRO (Modelling Policies for the improvement of Production Systems' Dependability). He is involved in several industrial projects with EDF, DCN, ALSTOM. His current research interests include prognosis (data driven and reliability driven prognosis), maintenance decision (opportunistic maintenance based on odds algorithm), dependability assessment, Integrated logistic support.

**Benoit IUNG** was born in 1962 in Nancy, France. He is currently full Professor at Lorraine University (France). He received his Ph.D. in Manufacturing Engineering (1992) and an accreditation to be research supervisor (2002), respectively, from this same University. Since 1988, he conducted researches at the Nancy Research Centre for Automatic Control (CRAN, UMR7039) where he is managing today a research group on Sustainable Industrial System Engineering. His research and teaching interests are related to dependability, prognostics, maintenance engineering and e-maintenance (PHM area). In relation to these topics he took scientific responsibility for the participation of CRAN in a lot of national, European (REMAFEX, DYNAMITE) and international projects (EIAM-IPE, iMaPla). He is now the chairman of the IFAC WG A-MEST on advanced maintenance, the chairman of the ESRA TC on Manufacturing, a member of the IFAC TC 5.1., a French Associate Member to CIRP, the co-chairman of the WG "Prognostics" of DIAG21 (French PHM community) and a founding Fellow to the International Society for Engineering Asset Management. Benoit Iung has authored several books including the first e-maintenance book in Springer and has published more than 90 scientific papers, 35 of them in international journals (i.e. ARC, RESS, IJPR, JIM, PP&C, JRR, JQME ...). He developed about 15 keynote speeches in international conferences. He has supervised until now about 15 MA, 12 Ph.D. Students and 2 Post-Doctorate students.

# Prognostics and Health Monitoring in the Presence of Heterogeneous Information

Gregory Bartram<sup>1</sup>, Sankaran Mahadevan<sup>2</sup>

<sup>1,2</sup>*Vanderbilt University, Nashville, TN, 37212, USA*

*gregory.w.bartram@vanderbilt.edu*  
*sankaran.mahadevan@vanderbilt.edu*

## ABSTRACT

Diagnosis and prognosis methodologies have used dynamic Bayesian networks (DBNs) to fuse many types of information. These methodologies, however, fuse problem-specific information and focus only on a subset of information types. By using only a subset of information, the interactions between or individual behaviors of subsystems, components, and faults may not be fully realized. In this paper, a general framework for system level diagnosis and prognosis of a mechanical system in the presence of heterogeneous information using dynamic Bayesian network (DBN) is proposed. Due to their ability to fuse heterogeneous information — information in a variety of formats from various sources — and give a probabilistic representation of a system, DBNs provide a platform naturally suited for diagnosis and uncertainty quantification. In the proposed methodology, a DBN is first constructed via an established machine learning algorithm from heterogeneous information. The DBN is then used to track the system and detect faults by monitoring the Bayes' factor of the system state estimate. When a fault occurs, the underlying system model changes, and the Bayes' factor of the DBN system model decreases. The state estimate provided by tracking indicates the most likely fault scenario and quantifies the diagnosis uncertainty. Estimation of remaining useful life and quantification of uncertainty in prognosis can then proceed based upon the diagnosis results. The proposed methodology is then demonstrated using a cantilever beam example with a possible loose bolt at the connection or a crack in the middle of the span.

## 1. INTRODUCTION

Diagnosis and prognosis have become increasingly important in the quest for safer, more intelligent, more efficient, and more cost effective systems. Accurate

diagnosis and prognosis are integral parts of system maintenance, operation, and design. Condition-based maintenance using automated diagnosis has gained traction in the belief that such a program can improve the safety and minimize the maintenance costs of a system. Mission-level decision making can also benefit from improved diagnosis and prognosis capabilities. Systems designed for prognosis and health management (PHM) also stand to benefit from improved diagnosis and prognosis.

One important requirement for a PHM methodology is the ability to utilize existing information about a system. Such information may be found in the form of expert opinion, operational and laboratory data, reliability data, and mathematical models. This information needs to be integrated into a system-level approach to better understand interactions between subsystems and components and make all of the information available for system-level diagnosis and prognosis procedures.

In previous work, static Bayesian networks (BNs) have commonly been used for diagnosis, particularly in medicine (e.g. Shwe et al. (Shwe et al., 1991)). Sahin et al. (Sahin, Yavuz, Arnavut, & Uluyol, 2007) learned a static BN of an aircraft engine using particle swarm optimization and used the learned BN for diagnosis. Examples of DBNs in diagnosis and prognosis are Jha et al. (Jha, Wenchao Li, & Seshia, 2009) who learn a DBN and use it to find transient faults, Przytula and Choi (Przytula & Choi, 2008) who use DBNs to obtain discrete qualitative measures of prognosis, McNaught and Zagorecki (McNaught & Zagorecki, 2009) who use a DBN of discrete variables to aid maintenance decisions, and Camci and Chinnam (Camci & Chinnam, 2005) who use hierarchical hidden Markov models for diagnosis and prognosis of machine parts. Straub (Straub, 2009) integrates model parameters into a DBN for modeling crack growth. However, existing DBN-based diagnosis and prognosis methodologies have focused on problems with a limited universe of information. In mechanical applications, many diverse sources of information must be integrated into the DBN.

Bartram and Mahadevan. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

The primary contribution of this paper is a methodology for performing diagnosis and prognosis of a mechanical system in the presence of uncertainty and in the presence of heterogeneous information. Heterogeneous information consists of information from various sources in varying formats such as expert opinion, laboratory and operational data, existing reliability data, and mathematical models. The methodology uses dynamic Bayesian networks (DBNs) for system modeling.

The remainder of the paper is organized as follows. First, DBNs and the process of building a DBN model from heterogeneous information are briefly explained. Model-based diagnosis using the DBN model and uncertainty quantification in diagnosis are then discussed. The proposed methodology is demonstrated for a cantilever beam.

## 2. PROPOSED METHODOLOGY

In this paper a methodology for performing diagnosis and prognosis of a mechanical system in the presence of heterogeneous information is proposed. First, a DBN model of the system is constructed. This model is used in conjunction with a particle filter to track the system. Binary faults are detected based Bayes' factors and isolated using the state estimate of the system. The future trajectory of the system is then estimated via particle filtering

### 2.1. Bayesian Networks & Dynamic Bayesian Networks

A static BN, also referred to as a belief network and directed acyclic graph (DAG), is a probabilistic graphical representation of a set of random variables and their conditional dependencies. Variables are represented by nodes (vertices) and conditional dependence is represented by directed edges. Unconnected nodes are conditionally independent of each other. The acyclic requirement means that no paths exist in the graph where, starting at node  $X_i$ , it is possible to return to node  $X_i$ .

A BN is a representation of the joint probability space  $U$  containing variables  $X_1, \dots, X_n$ . The joint probability distribution is represented in factored form as

$$p(X_1, \dots, X_n) = \prod_{i=1}^n p(X_i | \Pi_i) \quad (1)$$

where  $\Pi_i$  is the set of nodes on which  $X_i$  is conditionally dependent, so  $p(X_i | X_1, \dots, X_{i-1}) = p(X_i | \Pi_i)$ . The nodes in  $\Pi_i$  are commonly referred to as the parents of node  $i$ . Note that this definition of a BN depends on the variable ordering.

The factored formulation in Eq. (1) is readily extended to handle different types of multivariate continuous

distributions such as the Gaussian as well as distributions consisting of Gaussian and discrete variables.

In a DBN, the variable  $\mathbf{X}[t]$  is the value of  $\mathbf{X}$  at time  $t$ . The probability distribution describing  $\mathbf{X}$  on the interval  $[0, \infty)$  is very complex, as it is over  $\bigcup_{t=0}^{\infty} \mathbf{X}[t]$ . Using the Markov assumption simplifies this distribution by assuming that only the present state of the variable  $\mathbf{X}[t]$  is necessary to estimate  $\mathbf{X}[t+1]$  and thus  $p(\mathbf{X}[t+1] | \mathbf{X}[0] \dots \mathbf{X}[t]) = p(\mathbf{X}[t+1] | \mathbf{X}[t])$ . Additionally, the process is assumed to be stationary, meaning that  $p(\mathbf{X}[t+1] | \mathbf{X}[t])$  is independent of  $t$ . This approach to modeling DBNs is developed by Friedman et al. (Friedman, Murphy, & Russell, 1998).

A DBN may be composed of all discrete variables, all continuous variables, or hybrid set of discrete and continuous variables. For modeling systems with faults, it is advantageous to consider a hybrid system, typically with the continuous variables being modeled as Gaussian and the faults being discrete. Theory for such networks is developed in Heckerman and Geiger (Heckerman & Geiger, 1995) and Lauritzen (Lauritzen, 1992).

### 2.2. Heterogeneous Information

An important feature of Bayesian analysis is the ability to encode mixed information types. The graphical and probabilistic nature of BNs provides many opportunities to integrate information into the model. This allows them to benefit from heterogeneous information, i.e. information entrained in a variety of sources and formats. These sources include observational and experimental data, published reliability data, mathematical behavior models of components or subsystems, or expert opinion. Existing research has not fully exploited heterogeneous information, especially with respect to building DBN models for diagnosis and prognosis. The following sections summarize heterogeneous information and its integration into a DBN model.

#### 2.2.1. Expert Opinion

The first information type used in BN and DBN learning is expert opinion. This information is typically elicited from a domain expert who can provide assumptions about the system variables, such as what variables to be modeled and what distributions to model them with. An expert may even be able to provide a network structure and/or distribution parameters.

#### 2.2.2. Operational Data

Operational (observational) data is data collected through passive observation of a system during its operation. The system is allowed to operate naturally without external intervention. Each case of operational data is a set of values for each variable. A set of cases forms a database. This data

is included in the network via Bayesian network structure and parameter learning algorithms.

### 2.2.3. Laboratory Data

Laboratory (experimental, interventional) data is obtained while observing the system under outside intervention. Outside intervention comes in the form of fixing the value of one or more of the system variables. An intervention shows how a particular variable assuming a particular value affects the behavior of the system but does not provide any insight as to how likely that variable is to naturally assume a particular value. This data is included in the network via Bayesian network structure and parameter learning algorithms.

### 2.2.4. Published Reliability Data

In some instances, reliability data may be available for certain system components. Reliability data is used to establish prior distributions for specific system faults or update distribution parameters of faults in a DBN as time passes. Reliability data may take on several forms including mean time between unscheduled removal (MTBUR) and mean time between unscheduled failures (MTBUF) or more simple measures such as failure rate and failure mode distribution.

### 2.2.5. Mathematical Behavior Models

Mathematical models may contain useful information for improving the system model. Mathematical models may either be physics-based (e.g. finite element or bond graph) or data-based/empirical (neural network, BN/DBN). They can be used to obtain reliability data, calculate values for immeasurable system variables (e.g. stress intensity factor or structural damping), or otherwise make modifications to the network.

An example mathematical model is the two-term exponential model,

$$y(x) = Ae^{Bx} + Ce^{Dx} \quad (2)$$

where  $A, B, C, D$  are model parameters to be estimated from data. To integrate this model into a DBN, the parameters for this model must be included as variables in the DBN. Through simulation, it has been determined that this model provides a suitable fit for modeling the length of a fatigue crack as it grows with the number of load cycles or the deflection of a cantilever beam with a growing crack.

### 2.2.6. Summary of Heterogeneous Information Integration

The information types discussed are used in concert with a DBN learning procedure to create a system model. Prior

domain knowledge is the starting point for determining basic assumptions about the network, such as whether to use all discrete variables, all Gaussian variables, or a hybrid network. Any structural rules (e.g. banning certain nodes from being parent nodes) for the network need to be defined at this point. Once the basic assumptions are in place, operational and laboratory data and calculated data from mathematical models may then be used to determine the best structure for the DBN and the distribution parameters of the network. Reliability data and mathematical reliability models can then be used to augment probability distributions for some faults.

## 2.3. Learning Bayesian Networks & Dynamic Bayesian Networks

Learning BNs and DBNs consists of learning the structure of a network which defines the conditional independence relationships between variables and learning the distribution parameters of the network. Algorithms have been developed that can learn the structure and parameters for networks composed of discrete and continuous variables as well as hidden variables and missing data.

### 2.3.1. Learning a Static BN

Initially, it is assumed that all the variables are independent with unknown distributions. After gathering all available expert opinion about the variables (distribution types, constraints on the network structure), the structure learning algorithm evaluates any laboratory and experimental data available to determine the system structure of the system BN. The laboratory and experimental data along with any reliability data or mathematical models are used to determine the distribution parameters of the network.

One of the first algorithms for learning a BN is the K2 algorithm by Cooper and Herskovits (Cooper & Herskovits, 1992). The algorithm requires placing the variables in a fixed order. The variable that is first in the order can have no parents and subsequent variables in the ordering may have the preceding nodes as parents. The algorithm analyzes each node, attempting to determine which set of parents maximizes some scoring metric for the local structure (node + parents). The scoring metric computes the probability of observing the data given that local structure (essentially, a Bayes' factor). Once each node has been evaluated, the collection of local structures is combined into a global structure for the network. Other algorithms are available such as conditional independence testing (de Campos, 2006) or expectation-maximization (EM) methods (Friedman, 1998). EM methods can handle situations where data is missing or variables are unobservable.

### 2.3.2. Learning a Dynamic BN

DBN learning generally consists of learning the structure of a transition network which defines the relationships between



variables in different time slices and learning the parameters of this network. Variations of several algorithms have been developed that can handle various combinations of discrete and continuous variables. The basic methodology for DBN structure learning followed in this paper is that given by Friedman et al. (Friedman et al., 1998).

Structure learning in a DBN is similar to that in a static BN. The observations for a dynamic system may include multiple time histories of instantiations of the system. By lagging each time history individually, data for a variable at  $t$  and  $t + 1$  are obtained. After each time history is lagged, all the time histories can be combined into a database. In this database, a data case represents simultaneous observations of all the variables at a discrete point in time. Note that for the purpose of learning, it is here assumed that

all the variables are observable. In general this may not be true.

Consider the cantilever beam model. Time series of measurements are collected for each beam. In this case, it is desirable not only to seed the faults (i.e. loosen the bolt manually), but also to observe healthy beams over time as they are loaded and then begin to form cracks or have a bolt gradually loosen. Say evaluating 100 beams results in 100 time series of data. Each time series of data is lagged individually so as to create e.g.  $B(t)$  and  $B(t+1)$ . Instead of 5 variables per beam, there are now 10. Then, the data for 100 sets of 10 variables are combined into a database for learning the DBN structure. Fig. 1 shows the initial assumption of the variables being independent and the structure after learning.

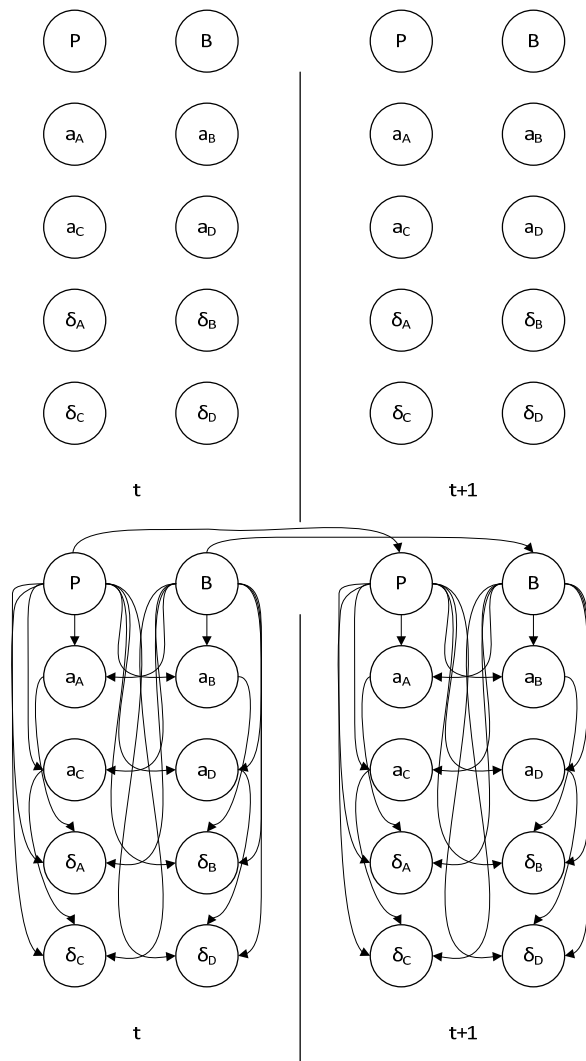


Figure 1. DBN before and after learning.

## 2.4. Diagnosis

Diagnosis is the process of detecting and isolating faults in a system and quantifying their magnitudes. Detection and isolation can be performed using a BN or DBN model of the system which treats faults as binary (true or false).

In detection, it is desired to determine the state of the system, including the states of potentially unobservable variables (faults). First, the case of a static system is considered followed by the dynamic case.

### 2.4.1. Diagnosis of a Static System

Diagnosis of a static system using a BN is a relatively straightforward problem. The purpose of diagnosis is to determine values of the unknown parameters (hidden state) from the measurements. Given observations of observable variables, the state of unobservable variables, which are hidden, can be inferred using Bayes' Theorem. The state of the fault values that best explains the observations is taken as the fault state. Inference is automated using algorithms such as the junction tree algorithm for exact inference or Monte Carlo-based approaches for approximate inference. Static BN diagnosis has been implemented in e.g. medical diagnosis.

### 2.4.2. Diagnosis of a Dynamic System

Diagnosis of a dynamic system is built around the concept of tracking. First, the tracking problem is explained as well as a Monte Carlo solution to the problem. Then, it is explained how the results of tracking can be used to detect and isolate faults in the system.

#### 2.4.2.1 Filtering

Diagnosis of a dynamic system reduces to the problem of filtering the state of the system over time, as opposed to inferring the state at once, as for a static system. Filtering combines theoretical system state estimates with noisy

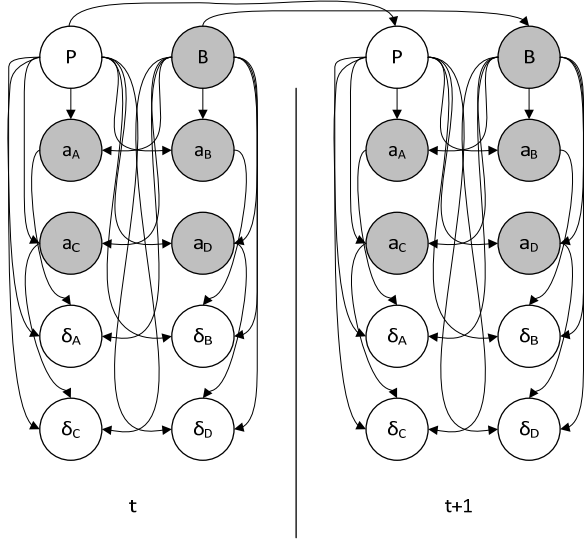


Figure 2. Cantilever beam DBN in filtering.  
Gray nodes are unobservable.

measurements to estimate the true state of the system. The state estimate includes estimates of the states of faults, whose states are otherwise unknown, which is why tracking can be used for diagnosis. The general tracking problem, explained by Arulampalam et al. (Arulampalam, Maskell, & Gordon, 2002) takes the form of a recursive Bayesian filter, where the prior pdf of the current state estimate, having been constructed from previous state estimates, is updated using a current measurement. This is also known as a predictor-corrector algorithm. The state estimate is often called the belief state due to the Bayesian nature of the estimation.

More formally, consider the  $k^{\text{th}}$  state estimate  $\mathbf{x}_k$  given by the state model  $\mathbf{f}_{k-1}(\mathbf{x}_{k-1}, \mathbf{v}_{k-1})$ , where  $\mathbf{x}_{k-1}$  is the previous state estimate and  $\mathbf{v}_{k-1}$  is an i.i.d process noise sequence. It is desired to determine  $\mathbf{x}_k$  using the measurements  $\mathbf{z}_k$  given by the measurement model  $\mathbf{h}_k(\mathbf{x}_k, \mathbf{n}_k)$  where  $\mathbf{n}_k$  is an i.i.d measurement noise sequence. In a Bayesian sense, this translates to finding the belief state, the pdf  $p(\mathbf{x}_k | \mathbf{Z}_{1:k})$ , where  $\mathbf{Z}_{1:k}$  are the measurements from 1:k. The prior pdf  $p(\mathbf{x}_0 | \mathbf{z}_0) \equiv p(\mathbf{x}_0)$  since  $\mathbf{z}_0$  is a set of no measurements.

If  $p(\mathbf{x}_{k-1} | \mathbf{Z}_{1:k-1})$  is available, the pdf  $p(\mathbf{x}_k | \mathbf{Z}_{1:k-1})$  may be estimated by the Chapman-Kolmogorov equation

$$p(\mathbf{x}_k | \mathbf{Z}_{1:k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathbf{Z}_{1:k-1}) d\mathbf{x}_{k-1} \quad (3)$$

where  $p(\mathbf{x}_k | \mathbf{x}_{k-1})$  is the probabilistic model defined by  $\mathbf{f}_{k-1}$  using the known statistics of the process noise  $\mathbf{v}_{k-1}$ . Eq. (1) makes use of the Markov property by having  $p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{Z}_{1:k-1}) = p(\mathbf{x}_k | \mathbf{x}_{k-1})$ . The estimation of the prediction pdf  $p(\mathbf{x}_k | \mathbf{Z}_{1:k-1})$  is known as the prediction stage.

Once the measurement  $\mathbf{z}_k$  is available,  $p(\mathbf{x}_k | \mathbf{Z}_{1:k})$  is determined in the update stage using Bayes' rule as

$$\begin{aligned} p(\mathbf{x}_k | \mathbf{Z}_{1:k}) &= p(\mathbf{x}_k | \mathbf{z}_k, \mathbf{Z}_{1:k-1}) \\ &= \frac{p(\mathbf{z}_k | \mathbf{x}_k, \mathbf{Z}_{1:k-1}) p(\mathbf{x}_k | \mathbf{Z}_{1:k-1})}{p(\mathbf{z}_k | \mathbf{Z}_{1:k-1})} \\ &= \frac{p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{Z}_{1:k-1})}{p(\mathbf{z}_k | \mathbf{Z}_{1:k-1})} \end{aligned} \quad (4)$$

where  $p(\mathbf{z}_k | \mathbf{Z}_{1:k-1}) = \int p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{Z}_{1:k-1}) d\mathbf{x}_k$  is a normalizing constant. The likelihood function  $p(\mathbf{z}_k | \mathbf{x}_k)$  is the probabilistic measurement model defined by  $\mathbf{h}_k$  and the known statistics of the measurement noise  $\mathbf{n}_k$ . The optimal state estimated may then be computed as the maximum a posteriori (MAP) estimate,  $\arg\max_{\mathbf{x}_k} p(\mathbf{x}_k | \mathbf{Z}_{1:k})$ , or the minimum mean-square error (MMSE),  $E\{\mathbf{x}_k | \mathbf{Z}_{1:k}\} = \int \mathbf{x}_k \cdot p(\mathbf{x}_k | \mathbf{Z}_{1:k}) d\mathbf{x}_k$  (Ristic & Arulampalam, 2004).

At first glance it appears that the structure of a DBN might allow for easy tracking of the belief state, perhaps similar to a static BN, and providing system-level diagnosis with minimal effort. Unfortunately, even with a structured process, the belief state quickly becomes very complex after only a couple of updates, except in degenerate cases, as over time the system becomes highly correlated (Boyer & Koller, 1998). Only specific implementations of the recursive filter such as the Kalman filter or grid-based methods provide analytical solutions. Even more, the computational expense of exact inference algorithms is exponential in the number of nodes for a discrete DBN and even worse for hybrid or continuous distribution DBNs. Exact inference in the conditional Gaussian distributions used in this paper is NP-hard (Lerner, 2001). Thus, it is necessary to pursue approximate inference algorithms. Particle filtering is one technique that makes the tracking problem tractable.

Particle filtering, i.e. sequential Monte Carlo (MC), is a method for approximating the distribution of the belief state. Common particle filtering method are based on sequential importance sampling (SIS), which improves upon the basic sequential MC by weighting point masses (particles) according to their importance sampling density, thus focusing on the samples that affect the posterior belief state the most. A comprehensive tutorial on particle filters is given by Ristic et al. (Ristic & Arulampalam, 2004).

A summary of the SIS algorithm for one time step is as follows. A previous (or initial if  $k = 1$ ) set of  $N$  weights  $w_{k-1}^i$  and  $N$  corresponding particles  $\mathbf{x}_{k-1}^i$  are given initially or known from the previous time step, where  $i$  denotes the  $i^{\text{th}}$  particle.  $N$  samples are drawn from the importance distribution,  $q(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{z}_k)$ , where  $\mathbf{z}_k$  are the measurements at the  $k^{\text{th}}$  time step. Unnormalized weights  $\tilde{w}_k^i$  are then computed up to a normalizing constant using a ratio of the distribution of the state estimate to the distribution of the importance sampling density. The weights  $\tilde{w}_k^i$  are then normalized so that their sum is equal to 1. The normalized weights  $w_k^i$  and points  $\mathbf{x}_k^i$  form an approximation to the state estimate  $p(\mathbf{x}_k | \mathbf{z}_k)$ . Two common choices of the importance density  $q(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{z}_k)$  are the prior  $p(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i)$  and the likelihood  $p(\mathbf{z}_k | \mathbf{x}_k^i)$ .

Many variations of the SIS PF exist using different priors, resampling techniques, etc. For the conditional Gaussian system model employed in this paper, also called a jump Markov model, hybrid state model, or switching system, a dynamic multiple model (MM) filter is appropriate for tracking. The multiple model particle filter (MMPF) given in Ristic et al. (Ristic & Arulampalam, 2004) implements this filter. This particle filter is a generalization of a standard particle filter and is equivalent to a standard particle filter when there is only one underlying discrete state. Each discrete state in  $r_k \in R_k$  represents an operating mode. Before doing SIS for the continuous components of the state, an operating mode is selected based on the probabilities of those operating modes occurring.

Considering the DBN dynamic beam model in Fig. 5, the state of the unobservable (gray) nodes at  $t + 1$  is first predicted based on the state of the unobservable nodes at  $t$ . Next, the measurements  $P = p_{t+1}$  and  $\delta = \delta_{t+1}$  are used to update (correct) the predictions and obtain the state estimate of the unobservable nodes. The process is implemented using a MMPF, which allows tracking of the discrete fault  $B$  as well as the continuous variables crack length,  $a$ , and stiffness,  $K$ .

#### 2.4.2.2 Fault Detection and Isolation

Given a DBN which can track the system, fault detection and isolation may occur in a couple ways. Roychoudhury et al. (Roychoudhury, Biswas, & Koutsoukos, 2006) use a DBN derived from a bond graph with SIS of the healthy system for tracking. The belief state estimate estimates are compared with measurements using a statistical hypothesis test to detect faults. A qualitative, bond-graph based procedure is used to isolate fault candidates. Separate DBNs for each fault candidate are then used to quantify the faults and determine the most likely fault candidate. Another example is provided by Jha et al. (Jha et al., 2009) who use a discrete DBN to determine the most probable explanation

of transient faults. However, the approach is offline and does not require any tracking. Still, it is successful in combining consistency and abduction-based diagnosis methods.

Bayes' factors (and related ratios) may be used to detect faults. Because the transition model  $p(\mathbf{x}_k | \mathbf{x}_{k-1})$  is not updated until a fault is diagnosed, after the fault occurs the model and measurements diverge and the Bayes' factor of the state estimate decreases. Thus, the Bayes' factor may be used to indicate the presence of a fault. Vaswani et al. (Vaswani, 2004) calculate  $-\log(Pr(\mathbf{z}_k | \mathbf{z}_{0:k-1}, M))$  where  $M$  is the transition model (not updated for the current fault) for abrupt faults and use the posterior expected negative log-likelihood of the state under the prior distribution of the state for incipient faults. In both cases a threshold for detection is established. Sankararaman and Mahadevan (Sankararaman & Mahadevan, 2011) use Bayes' factors to indicate faults and update the Bayes' factor as new data is collected. Again, if the Bayes' factor crosses a predetermined threshold, the fault is then isolated by checking the state estimate of the system. Following the approach of Sankararaman and Mahadevan (Sankararaman & Mahadevan, 2011), the Bayes' factor is taken as

$$B = \frac{Pr(D|H_1)}{Pr(D|H_0)} \quad (5)$$

where  $D$  is the set of measurements,  $H_0$  is the no damage hypothesis,  $H_1$  is the damage hypothesis, and  $Pr(D|H_1) = 1 - Pr(D|H_0)$ . The probability of damage is

$$Pr(H_1|D) = \frac{Pr(D|H_1)P(H_1)}{Pr(D|H_0)P(H_0) + Pr(D|H_1)P(H_1)} \quad (6)$$

If the prior probabilities of  $H_0$  and  $H_1$  are assumed to be 0.5, Eq. (5) reduces to

$$Pr(H_1|D) = \frac{B}{B + 1} \quad (7)$$

If a new Bayes' factor is calculated based on new measurement information becoming available, the updated probabilities of  $H_0$  and  $H_1$  are

$$Pr(H_1|D) = \frac{B'B}{1 + B'B} \quad (8)$$

and

$$Pr(H_0|D) = \frac{1}{1 + B'B}. \quad (9)$$

Following the results of Vaswani et al. (Vaswani, 2004), for detection of abrupt faults,  $H_0$  and  $H_1$  refer to healthy and

unhealthy realizations of the transition model,  $p(\mathbf{x}_k^n | \mathbf{x}_{k-1}^n, r_k^n)$ , and we denote the corresponding Bayes' factor as  $B_t$ . For incipient faults,  $H_0$  and  $H_1$  refer to a baseline and current value of the state estimate and we denote the corresponding Bayes' factor as  $B_t$ . By updating both Bayes' factors and comparing to a threshold probability, fault detection can occur. Fault isolation is then achieved by determining the MMSE or MAP estimate from the state estimated, as approximated by samples and weights. After isolation, any parameter changes that need to be made to the state transition or measurement models are made.

### 2.5. Prognosis

In prognosis, it is desired to make an inference about the future distribution of the state of a system given its current state and possible future scenarios. As no new measurements are available, a Bayesian recursive filter which integrates measurement data is no longer necessary. The last set of state estimates and measurements are propagated through the DBN (as was done with the particle filter). The result is an estimate of the future distribution of the state variables.

### 3. NUMERICAL EXAMPLE

To illustrate the diagnosis methodology, fault detection, isolation, and prognosis is performed on a cantilever beam system subjected repeatedly to a point load at the free end. In this system, a crack is growing near the support according to Paris' law (Paris, Gomez, & Anderson, 1961)

$$\frac{da}{dN} = C(\Delta K)^m. \quad (10)$$

where  $C$  and  $m$  are material dependent constants,  $a$  is the crack length,  $N$  is the number of load cycles, and  $\Delta K$  is the stress intensity factor.

Additionally, a bolt may become loose at the support, which is modeled by removing constraints on finite element model boundary conditions for nodes at the support up to a distance  $d$  from the bottom of the beam.

#### 3.1.1. Model Construction

To obtain operational data, 50 load histories were constructed containing 100 load cycles each. In these load histories, it was assumed that crack length, deflection, load, and whether the bolt is loose or tight are observable. For each load history, a load  $P$ , Paris law parameter  $m$ ,



Figure 3. Meshed cantilever beam finite element model

Variable	Symbol	Distribution
Bolt state	$B$	Categorical, binary (tight, loose)
Load	$P$	Categorical, ternary (-.5 kN, -1 kN, -1.5 kips)
Crack length parameter $A$	$a_A$	Gaussian
Crack length parameter $B$	$a_B$	Gaussian
Crack length parameter $C$	$a_C$	Gaussian
Crack length parameter $D$	$a_D$	Gaussian
Deflection parameter $A$	$\delta_A$	Gaussian
Deflection parameter $B$	$\delta_B$	Gaussian
Deflection parameter $C$	$\delta_C$	Gaussian
Deflection parameter $D$	$\delta_D$	Gaussian

Table 1. Cantilever beam model variables.

exponential failure rate for the bolt loosening, and elastic modulus  $E$  (finite element model input) are selected. Paris' law parameter  $C$  was held constant across all experiments. A crack is initiated near the support of the beam. The beam is then repeatedly subjected to load  $P$ . With each application of  $P$ , the crack grows slightly larger, according to Paris' law. Additionally, there is a growing chance with each application of  $P$  that the bolt will become loose. The severity of damage is determined by  $d$ , which is selected before each load history. Table 1 describes the distributions used to model these variables. A rendering of the system in ANSYS is shown in Fig. 6.

A DBN model of the beam (Fig. 3) is systematically constructed from heterogeneous information. First expert opinion is solicited to determine the variables to be used in modeling the system and the structure of the DBN. The variables selected to represent the system model are the loose bolt indicator variable,  $B$ , the load  $P$ , the parameters of the two-term exponential model describing the crack length,  $a$ , and the deflection,  $\delta$ . The two-term parameters of the exponential models are estimated over a sliding 10 load cycle window from smoothed crack length and deflection data. The parameters of the DBN are then calculated via the principle of maximum likelihood.

#### 3.1.2. Diagnosis

To perform diagnosis, a 100 cycle load history was synthesized with  $P = -.5$  kN and known deflection parameters,  $\delta_A, \delta_B, \delta_C, \delta_D$ . The bolt condition,  $B$ , crack length parameters,  $a_A, a_B, a_C, a_D$  were assumed to be unobservable. The DBN is used in conjunction with a MMPF with 200 particles to obtain filtered estimates of the system state. After each load cycle, detection probabilities corresponding to the loose bolt condition are

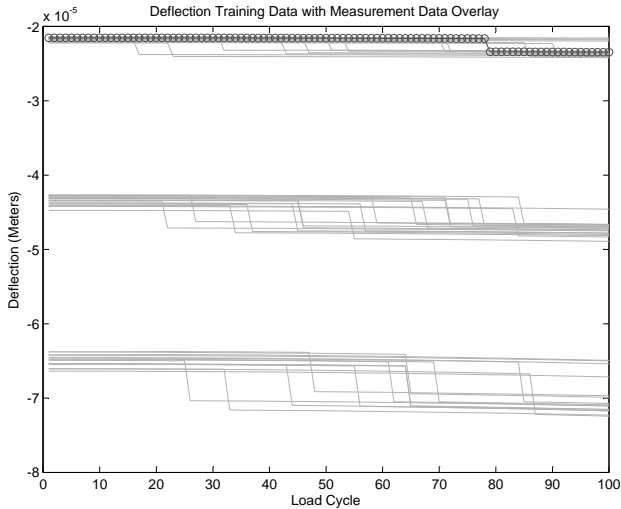


Figure 4. Deflection data for training DBN (gray lines) compared against measurement data.

updated as per Section II.D. The threshold probability of detection used is 95%. As the fault considered was abrupt, the Bayes' factor corresponding to the transition model as opposed to the belief state was used in detection.

### 3.1.3 Prognosis

After the last measurement data has been filtered, the system state is estimated 100 cycles into the future. As no new measurements are available, a Bayesian recursive filter which integrates measurement data is no longer necessary. The last set of state estimates and measurements are propagated through the DBN (as was done with the particle filter) with one estimate per particle. The result is an estimate of the future distribution of the model parameters. By using Eq. (2) with the model parameters, estimates of the crack length and deflection distributions are obtained. The model parameters are smoothed offline before estimating crack length and deflection so that trends are easier to visualize.

### 3.1.3. Results and Discussion

Data for the DBN for crack length and deflection used in training the DBN are overlaid with the measurement data used in diagnosis in Fig. 4. Considering the measurement data, the first item to discuss is the smoothness of the crack length and deflection curves versus the jagged model parameter curves. The jagged curves of particle values for the crack length parameters overlaid with the MAP estimate and true value are shown in Fig. 5. This is due to several uncertainties in estimating the model parameters. First, the curves produced by the FEM model may not be perfectly smooth. Locally, parts of the curve may not fit the two-term

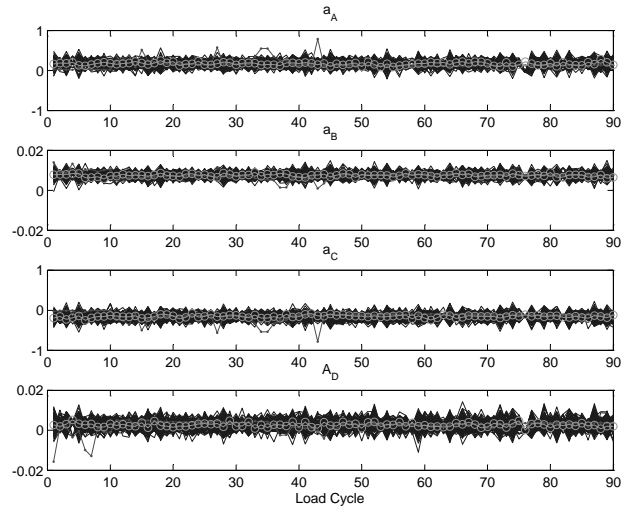


Figure 5. Overlay of MAP filtered estimate (gray with circles), true values (gray), and filtered estimates (black) for crack length parameters

exponential model as well as others. A limited window of data is used in estimating these parameters, resulting in further errors. Also, the optimization procedure used to estimate the model parameters induces errors.

The operational data lies in three bands, one for each value of  $P$ . When the bolt becomes loose, it is important to note that the deflection does not fall far outside the range of healthy deflection curves. This highlights the importance of integrating a mathematical model into this system. The parameters of the mathematical model, on the other hand, are able to isolate the loose bolt fault shortly after occurrence as well as estimate the crack length and deflection. The MAP estimate of the bolt condition, the true value of the bolt condition, and the measured values of  $\delta_D$ , which is the variable most strongly connected with the bolt condition, are shown in Fig. 6.

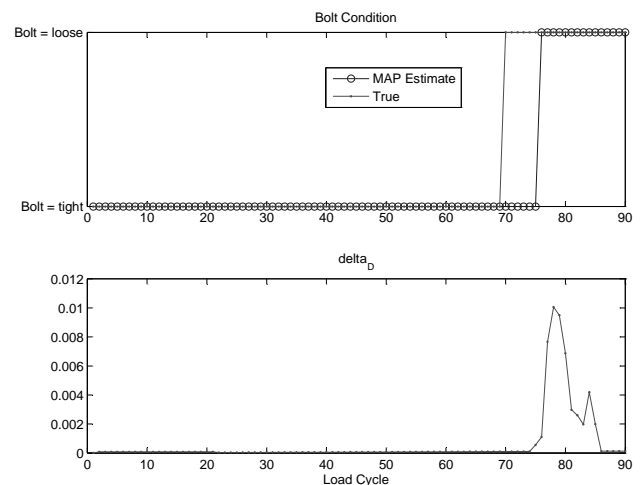


Figure 6. MAP estimate vs. true value of bolt condition and plot of  $\delta_D$  when a fault occurs.

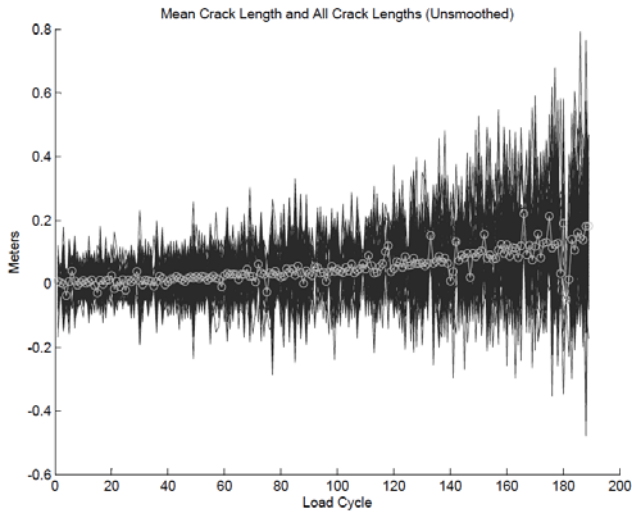


Figure 7. All crack length estimates (black) and mean crack length estimate (gray) computed from crack length mathematical model.

Prognosis of the system shows how small changes in the model parameters result in large amounts of uncertainty in prognosis. The prognosis results (Fig. 7) are in line with expectations, giving the trends evident in the training data (Fig. 4). Figure 8 shows histograms of crack length data after 115 and 175 load cycles, showing a gradual shift to towards larger crack lengths. Using MAP estimates and offline smoothing of the crack length (Fig. 9) and deflection parameters enhances the legibility of the plots but can induce some error, particularly near the beginning and end of a time series.

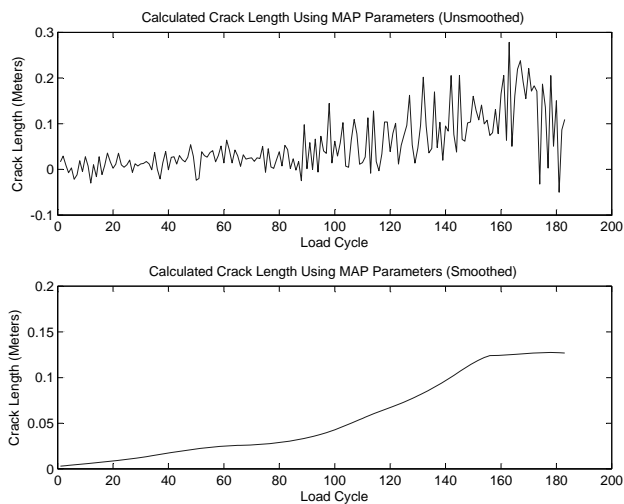


Figure 9. Crack length estimate using smoothed and unsmoothed MAP parameters.

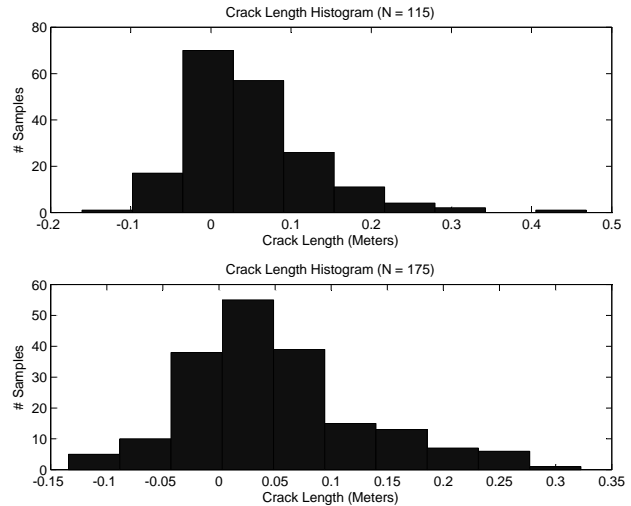


Figure 8. Histograms of crack length after 115 and 175 load cycles).

The results show that a DBN-based methodology can be used for diagnosis and prognosis of a mechanical system in the presence of many different types of information. Where other methodologies may exclusively model crack growth or detect faults, the DBN-based methodology is capable of doing both as it integrates expert opinion, a database of observational information, failure rate information, and a mathematical model for crack growth and deflection.

#### 4. CONCLUSION

In this paper, a methodology for diagnosis and prognosis of a mechanical system in the presence of heterogeneous information has been proposed. First, a DBN is constructed using heterogeneous information such as expert opinion, operational and laboratory data, published reliability data, and mathematical models. Next, the DBN is employed for filtering the system. Filtering is performed via a particle filter. Faults are detected using Bayesian hypothesis testing. Then, by determining the MAP estimate of the state, faults are isolated. After the diagnosis has been performed, a prognosis estimate of the future distribution of the system state is obtained. In future work, a hydraulic actuator system constructed from heterogeneous information and containing multiple fault scenarios will be considered for PHM.

#### ACKNOWLEDGEMENT

This work is partly supported by the U. S. Air Force Research Laboratory through subcontract to General Dynamics Information Technology (Contract No. USAF-0060-43-0001, Project Monitor: Mark Derriso). The support is gratefully acknowledged.

## NOMENCLATURE

B	=	Bayes' factor
DBN	=	dynamic Bayesian network
MAP	=	Maximum a posteriori estimate
MMSE	=	Minimum mean-square estimate
PF	=	Particle filter
SIS	=	Sequential importance sampling

## REFERENCES

- Arulampalam, M. S., Maskell, S., & Gordon, N. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 50, 174–188.
- Boyer, X., & Koller, D. (1998). Tractable inference for complex stochastic processes. *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, UAI'98 (pp. 33–42). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. Retrieved from <http://dl.acm.org/citation.cfm?id=2074094.2074099>
- Camci, F., & Chinnam, R. B. (2005). Dynamic Bayesian networks for machine diagnostics: hierarchical hidden Markov models vs. competitive learning. *2005 IEEE International Joint Conference on Neural Networks, 2005. IJCNN '05. Proceedings* (Vol. 3, pp. 1752–1757 vol. 3). Presented at the 2005 IEEE International Joint Conference on Neural Networks, 2005. IJCNN '05. Proceedings, IEEE. doi:10.1109/IJCNN.2005.1556145
- Cooper, G. F., & Herskovits, E. (1992). A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9(4), 309–347. doi:10.1007/BF00994110
- de Campos, L. M. (2006). A Scoring Function for Learning Bayesian Networks based on Mutual Information and Conditional Independence Tests. *J. Mach. Learn. Res.*, 7, 2149–2187.
- Friedman, N. (1998). The Bayesian Structural EM Algorithm. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.24.1555>
- Friedman, N., Murphy, K., & Russell, S. (1998). Learning the Structure of Dynamic Probabilistic Networks. *UAI'98 Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence* (pp. 139–147). Morgan Kaufmann Publishers Inc. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.32.6615>
- Heckerman, D., & Geiger, D. (1995). Learning Bayesian Networks: A unification for discrete and Gaussian domains. *PROCEEDINGS OF ELEVENTH CONFERENCE ON UNCERTAINTY IN ARTIFICIAL INTELLIGENCE*. Retrieved from <http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.156.7976>
- Jha, S., Wenchao Li, & Seshia, S. A. (2009). Localizing transient faults using dynamic bayesian networks. *High Level Design Validation and Test Workshop, 2009. HLDVT 2009. IEEE International* (pp. 82–87). Presented at the High Level Design Validation and Test Workshop, 2009. HLDVT 2009. IEEE International, IEEE. doi:10.1109/HLDVT.2009.5340170
- Lauritzen, S. L. (1992). Propagation of Probabilities, Means, and Variances in Mixed Graphical Association Models. *Journal of the American Statistical Association*, 87(420), 1098–1108. doi:10.2307/2290647
- Lerner, U. (2001). Exact inference in networks with discrete children of continuous parents. in: *J. Breese, D. Koller (Eds.), Uncertainty in Artificial Intelligence* (pp. 319–328). Morgan Kaufmann.
- McNaught, K. R., & Zagorecki, A. (2009). Using dynamic Bayesian networks for prognostic modelling to inform maintenance decision making. *IEEE International Conference on Industrial Engineering and Engineering Management, 2009. IEEM 2009* (pp. 1155–1159). Presented at the IEEE International Conference on Industrial Engineering and Engineering Management, 2009. IEEM 2009, IEEE. doi:10.1109/IEEM.2009.5372973
- Paris, P. C., Gomez, M. P., & Anderson, W. E. (1961). A Rational Analytic Theory of Fatigue. *The Trend in Engineering*, 13, 9–14.
- Przytula, K. W., & Choi, A. (2008). An Implementation of Prognosis with Dynamic Bayesian Networks. *2008 IEEE Aerospace Conference* (pp. 1–8). Presented at the 2008 IEEE Aerospace Conference, IEEE. doi:10.1109/AERO.2008.4526616
- Ristic, B., & Arulampalam, S. (2004). *Beyond the Kalman filter: particle filters for tracking applications*. Boston, MA: Artech House.
- Roychoudhury, I., Biswas, G., & Koutsoukos, X. (2006). A Bayesian approach to efficient diagnosis of incipient faults. *IN PROC. 17TH INT. WORKSHOP PRINCIPLES OF DIAGNOSIS*, 243–250.
- Sahin, F., Yavuz, M. Ç., Arnavut, Z., & Uluyol, Ö. (2007). Fault diagnosis for airplane engines using Bayesian networks and distributed particle swarm optimization. *Parallel Computing*, 33(2), 124–143. doi:10.1016/j.parco.2006.11.005
- Sankararaman, S., & Mahadevan, S. (2011). Uncertainty quantification in structural damage diagnosis.

- Structural Control and Health Monitoring*, 18(8), 807–824. doi:10.1002/stc.400
- Shwe, M. A., Middleton, B., Heckerman, D. E., Henrion, M., Horvitz, E. J., Lehmann, H. P., & Cooper, G. F. (1991). Probabilistic Diagnosis Using a Reformulation of the INTERNIST-1/QMR Knowledge Base. *Methods of Information in Medicine*, 30, 241–55.
- Straub, D. (2009). Stochastic Modeling of Deterioration Processes through Dynamic Bayesian Networks. *Journal of Engineering Mechanics*, 135(10), 1089–1099. doi:10.1061/(ASCE)EM.1943-7889.0000024
- Vaswani, N. (2004). *Bound on Errors in Particle Filtering with Incorrect Model Assumptions and Its Implication for Change Detection*.



# A Multi-resolution Experimental Methodology for Fatigue Mechanism Verification of Physics-based Prognostics

Jian Yang, Wei Zhang, and Yongming Liu

*Department of Civil and Environmental Engineering, Clarkson University, Potsdam, NY, 13699, USA*

*jiyang@clarkson.edu  
zhangw@clarkson.edu  
yliu@clarkson.edu*

## ABSTRACT

An experimental methodology is proposed in this paper for mechanism verification of physics-based prognosis of mechanical damage, such as fatigue. The proposed experimental methodology includes multi-resolution in-situ mechanical testing, advanced imaging analysis, and mechanism analysis based on digital measurements. A case study is presented for fatigue crack growth mechanism investigation. In-situ fatigue testing at lower resolutions, i.e., optical microscopy, and digital image correlation is used to analyze the plastic deformation behavior and strain distribution near crack tips. In-situ fatigue testing under higher resolutions, i.e., scanning electron microscopy, and automatic image tracking is used to obtain detailed crack tip deformation and crack growth kinetics at the nanometer scales. Following this, the proposed experimental methodology is applied to two different metallic materials, aluminum alloys and steels. Very different experimental observations are observed and the underlying mechanisms are discussed in detail. The impact on the prognosis algorithm development is also discussed. Finally, the potential application of the proposed experimental methodology to other materials systems and to other types of mechanical damage is discussed.

## 1. INTRODUCTION

Structural damage prognosis mainly relies on two kinds of methods: data-driven methods and physical-based methods. Though they are often combined in application, the physical based method is much more precise and flexible for new loading condition even new systems with a similar mechanism (Farrar & Lieven, 2007). Among the various structural failures modes, fatigue caused failure is a main failure mode of structures under cyclic loading and it can

have catastrophic consequences, with loss of life and property (Campbell & Lahey, 1984).

The theory of metal fatigue has been developed for a long time. In 1962, Forsyth first gives a detail of the two stages in fatigue initialization and propagation (Forsyth, 1962). In stage I, when the initial crack length is less than a few grain diameters, an initial stress is required to open the crack as it might be fully closed, and additional stress is applied for crack propagation. In stage II, the crack growth process involves simultaneous or alternating flow along two slip system and it largely depend on the stress intensity at the crack tip. Paris' law (Paris, Gomez, & Anderson, 1961) gives a power relationship between fatigue crack growth rate and stress intensity range of constant amplitude loading. However, when considering the failure physics, the crack does not propagate throughout the whole unloading cycle, though the unloading has a sharpening effect to the crack tip (Laird, 1967). Moreover, when the stress is removed, the residual plastic region may force the crack close fully at the crack tip (Wolf, 1970). However, many experiments show the difficulties in measuring the crack-opening load, depending on the measuring location (Macha, Corbly, & Jones, 1979) and the techniques employed, such as by electrical potential method (Shih and Wei 1974), by ultrasonic method (Bouami & De Vadder, 1986; Singh, Srivastav, Gupta, Keller, & Ray, 2009), and numerical method (Riddell, Piascik, Sutton, Zhao, McNeill & Helm, 1999). Further, it is found that commonly employed notch-mouth clip-gauge method is not sensitive enough to detect the closure of short cracks in regions of notch plasticity (Fleck & Shin, 1985) and changes in the stresses ahead of the crack tip are more important than closure behind the crack tip (Sadananda, 1999).

For crack closure hypothesis, many experimental studies are based on the indirect measurement and there lacks a method for the direct model verification at various different scales. The mechanism/hypothesis verification is critical for a

Jian Yang et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

sound physics-based prognosis framework and must be carefully investigated.

In this paper, a multi-resolution in-situ testing methodology is proposed for the mechanism/hypotheses verification for mechanical damage prognosis. The proposed methodology combines advanced experimental instrumentation, digital imaging technique, and mechanical analysis to investigate the mechanical deformation and damage progression behavior under mechanical loadings at two scales, micrometer scale (optical) and nanometer scale (SEM). The two experimental testing results are compared and verified each other for non-closure hypotheses. The application in two materials show significantly different fatigue behaviors and the direct observations at nanometer to millimeter scales provide fruitful information for the new model development and model hypothesis verification.

This paper is organized as follows. First, the in-situ experiment methodology and the detail steps are introduced. In-situ fatigue testing at lower resolutions, i.e., optical microscopy, and digital image correlation is used to analyze the plastic deformation behavior and strain distribution near crack tips. In-situ fatigue testing under higher resolutions, i.e., scanning electron microscopy, and automatic image tracking is used to obtain detailed crack tip deformation and crack growth kinetics at the nanometer scales. Following this, the proposed experimental methodology is applied to two different metallic materials, e.g., aluminum alloys and steels. Very different experimental observations are observed and the underlying mechanisms are discussed in detail. The impact on the prognosis algorithm development is also discussed. Finally, the potential application of the proposed experimental methodology to other materials systems and to other types of mechanical damage is discussed.

## 2. IN-SITU OPTICAL EXPERIMENT

### 2.1. Experiment Setup

The experimental set-up for the in-situ optical microscopy experiment is shown in Fig. 1. The experimental instrument mainly includes two parts: a palm-sized loading system and an optical microscope system. The tensile stage is manufactured by Fullerm Inc. (now MTI) and the maximum gage length between mechanical grips is about 27 mm. The load capacity is 5 kN. The sub-stage is fixed on the microscope and the cyclic loading is applied to the specimen. The testing is controlled by the Admet MTESTQuattro system. A Nikon metallurgical microscope is used to monitor the specimen surface and a high resolution imaging acquisition system is used to record images during the testing. Post-processing of images is performed using a high performance workstation and the software package ImageJ

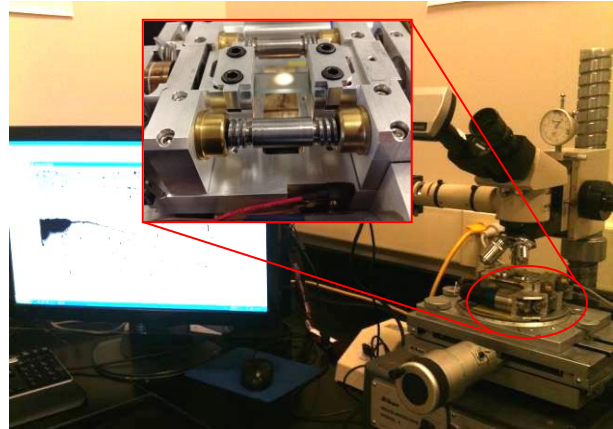


Fig. 1 Experiment setup

Single edge notched plate specimens with width  $W = 8.1$  mm, length  $L = 52$  mm and thickness 0.86 mm made of aluminum alloy sheet (7075-T6) were tested. An edge notch of length 1mm was machined on specimen by slow saw cutting. After machining, the specimens are pre-cracked under a hydraulic tension machine INSTRON 1331 until the initial crack reaches about 1 mm. The pre-cracking procedure follows the ASTM standard E647-99. Following this, both surfaces of the specimen are polished with the sandpaper whose average particle diameter is smaller than 10  $\mu\text{m}$ . The final polishing is done using a vibration polishing machine with 1~3 $\mu\text{m}$  polishing suspension.

### 2.2. Test Procedure

The experiment consists of four major steps:

1. Specimen manufacturing and pre-cracking.
2. Polish the specimen carefully to form randomly distributed small dark regions on the smooth surface, which is a requirement for digital image correlation (DIC) analysis.
3. Apply load on the specimen under the monitoring of a microscope, while images of the crack tip region are taken during the loading and unloading process.
4. process the images by DIC software to get the strain field of the crack tip region and calculate the plastic zone using mechanical analysis.

In the experiment, a load cycle is divided into several steps in the load profile. At each step, the image around the crack region is taken for strain calculation. The step load is set according to the precision requirement and computation resource limitation. An example profile is shown in Fig. 2, which has about forty segments during the loading and unloading path.

### 2.3. Test Procedure

Before the images are processed using VIC-2D for strain calculation, some pre-processes are required depending on

the image quality. The most common process is the contrast enhancement. The raw images may not be sharp enough for DIC calculation and enhancing the contrast could make the pixels easy to be recognized. In addition, it could also mitigate the negative effect of blur edge by out of focus.

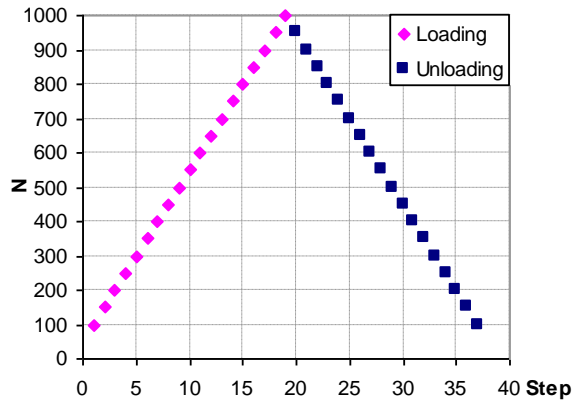


Fig. 2 Load profile

After the image pre-processing, digital image correlation can be used to estimate the strain and displacement of the region of interest. The obtained strain field is used for the mechanical and damage analysis.

In fatigue, plastic deformation is one of the most important factors affecting damage accumulation. Thus, the focus in this paper is on the plastic zone estimation. Under cyclic loading, the crack closure during the reverse loading path (or the crack opening level during the loading path) will be critical for fatigue analysis. Some previous studies have shown the crack closure level and crack opening level are very close and can be assumed to be equal (Zhang & Liu, 2011). The identification during the reverse loading path will be sufficient to estimate this value. It should be noted that the proposed experimental methodology is not limited to fatigue and other types of mechanical damage can be analyzed using the obtained strain field as well.

The theoretical reversed plastic zone under cyclic loading can be estimated by Eq. (1) without the consideration of crack closure (Rice, 1968)

$$\rho = \frac{1}{\pi} \left( \frac{\Delta K}{2\sigma_y} \right)^2 \quad (1)$$

where  $\sigma_y$  is the yielding strength and

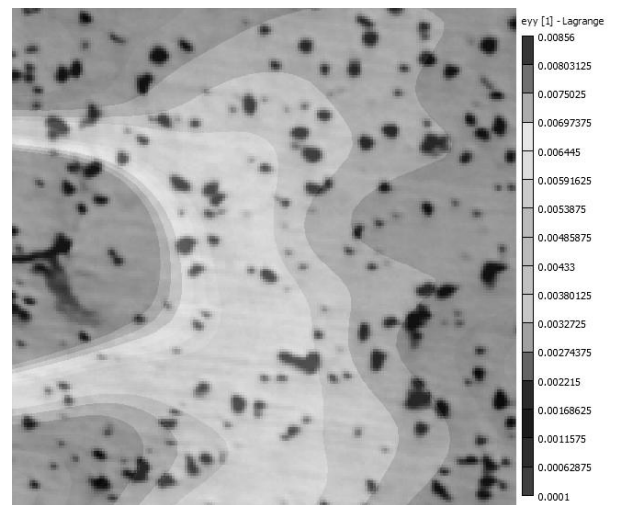
$$\Delta K = F \cdot \Delta\sigma \sqrt{\pi a} = F \cdot (\sigma_{\max} - \sigma_{\min}) \sqrt{\pi a} \quad (2)$$

where  $F$  is the geometry factor which depends on the geometry of specimen and the crack length.  $a$  is the crack length. A sample contour of strain field in front of the crack is shown in Fig. 3(a), the crack grows from the center of left

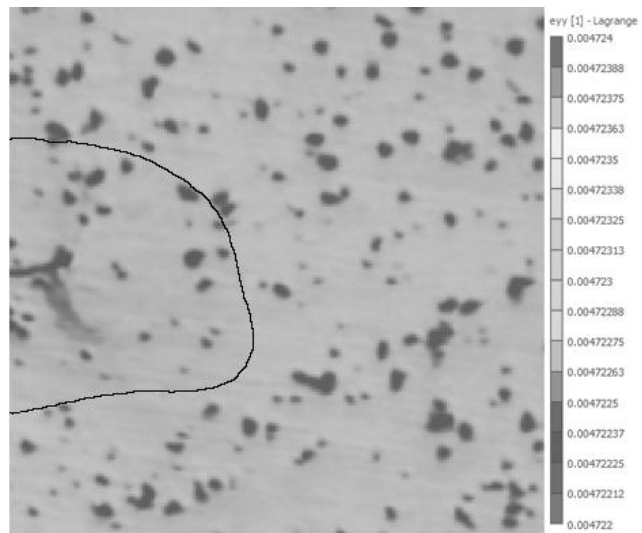
edge. The calculated plastic zone is shown as the red region in Fig. 3 (b). Finally, the plastic zone size is defined along the crack propagation direction and it can be measured directly from the processed image.

By repeating the above procedures for images taken at different loading levels, the strain field and plastic zone evolution can be obtained. By comparing the experimental observations with theoretical model predictions (for example, Eq. (1), the hypothesis of the model can be justified.

In-situ optical testing has the resolution limit to about 0.5~5 microns. If more detailed information of the materials at finer resolution is desired, new imaging capabilities is required. In-situ SEM testing is required and is discussed below.



(a) Strain contour from VIC-2D



(b) Plastic zone from VIC-2D

Fig. 3 Strain field and plastic zone from VIC-2D

### 3. IN-SITU SEM EXPERIMENT

#### 3.1. Experiment Setup

The process of in-situ SEM experiment is similar with optical experiment. The palm-sized loading machine is put in a scanning electron microscope for imaging (JEOL-7400F SEM in the current investigation). The experimental setup is shown in Fig. 4.

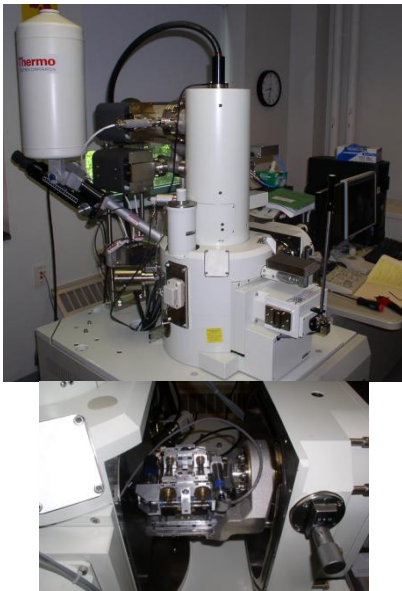


Fig. 4 SEM experiment set up

#### 3.2. Test Procedure

The load profile is the same with optical experiment. There is a minor difference. Under the optical observation, the lens field depth is large enough to cover the specimen moving distance. Thus, the images can be taken continuously. However, the field depth of SEM is very short. Thus, at each step, it is required to stop the motor and refocus the lens on the specimen surface by a quick scanning. Following that, scan the crack tip again with higher resolution.

#### 3.3. Image processing and Measurements

After SEM images are taken, some pre-processing are also needed to improve the image quality. Different from the optical image, SEM image consists of pixels calculated from the reflected electrons. As the electron reflection is not continuous in atomic length scale, the calculated pixels color changing is not smooth. The images taken under the visible light with wavelength much longer than atom scale length, the pixel colors are changing smoothly. The common process methods are noise reduction and smoothing. An example of comparison of images before and

after processing is shown in Fig. 5, which is from the case study of steel.

Finally, crack tip opening displacement (CTOD) can be measured directly from the processed images. The definition of CTOD is defined as the distance of crack surface at the right angle edge from the crack tip, as Fig. 6(a) shows (Shih, 1981). Fig. 6(b) is an example SEM image with CTOD marked with white lines.

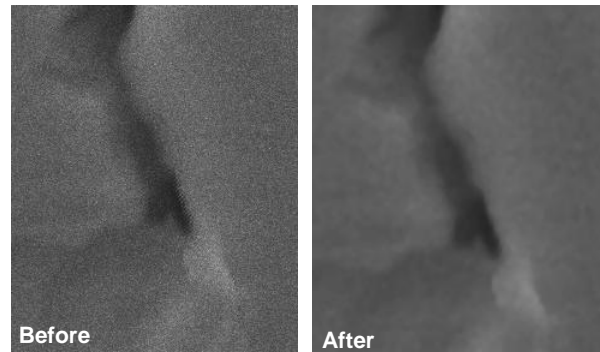
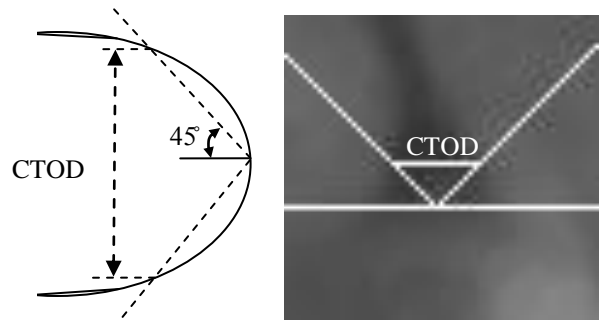


Fig. 5 Comparison of SEM image before and after processing



(a) CTOD definition (b) CTOD marked in SEM image

Fig. 6 CTOD measuring method and example

### 4. CASE STUDY

Using the above described in-situ optical and SEM testing methodology, two set of experiments are carried out on aluminum and steel, respectively. The objective is to investigate the plastic deformation and crack growth behavior and verify the classical mechanical models that have used widely for the plastic zone estimation and crack propagation law.

#### 4.1. Case I: Aluminum 7075-T6

Aluminum 7075-T6 is a high strength to density material used for highly stressed structures but under strict weight requirement, such as automotive, marine, aircraft and many other daily staffs. In this case, three specimens are tested.

The chemical composition of this material is listed in Table 1. The aluminum is the balance in the total weight. The basic physical properties are listed in Table 2.

Element	Min	Max
Zn	5.1	6.1
Mg	2.1	2.9
Cu	1.2	2.0
Fe	0	0.5
Si	0	0.4
Mn	0	0.3
Cr	0.18	0.28

Table 1. Chemical composition Al 7075-T6 (weight, %)

Elastic Modulus /GPa	Yield Strength /MPa	Tensile Strength /MPa
71.7	502~516	573~582

Table 2. Basic mechanical properties of Al 7075-T6

Following the above discussed general methodology, the plastic zone size at each loading step can be estimated using the DIC analysis. The plastic zone size with the applied stress intensity factor is shown in Fig. 7. It could be easily found that during the unloading process, the reversed plastic zone size stops to grow at about 30% of the maximum loading. Crack closure is a sound reason for this behavior since the crack contact will reduce the stress intensity ahead of the original crack tip and plastic deformation disappears. Because as the crack closes, the crack surfaces are contacted to each other and it releases the reverse compression force from the crack tip region.

The theoretical value is calculated by Eq. (1) and (2), the value of geometry factor  $F$  is defined in Eq. (3) according to the ASTM standard E647-99.

$$F = 1.12 - 0.231 \left( \frac{a}{W} \right) + 10.55 \left( \frac{a}{W} \right)^2 - 21.72 \left( \frac{a}{W} \right)^3 + 30.39 \left( \frac{a}{W} \right)^4 \quad (3)$$

In order to full verify this hypothesis, the in-situ SEM experiment is taken to verify the crack closure and another crack parameter CTOD is measured directly.

Selected SEM images are shown in Fig. 8(a) and all the CTOD measurements are shown in Fig. 8(b). The crack closure is directly observed. The first two images shows that crack tip remain closed at two different loading levels. If further increase the loading level, crack becomes open. This explains why the plastic zone stops increases under in-situ optical testing. The testing results verify the crack closure hypothesis and indicate that the Eq. (1) is not applicable to aluminum alloys under the current investigation.

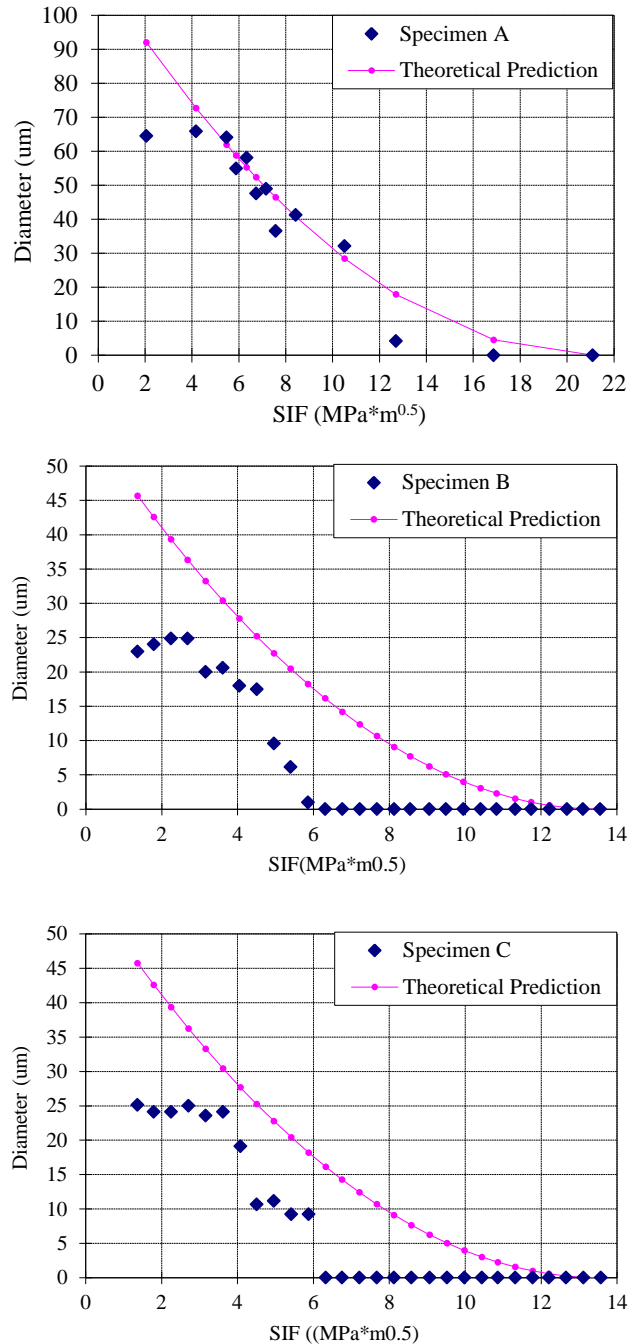


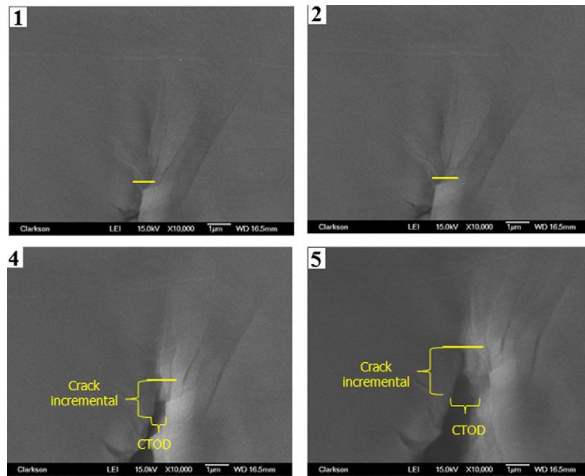
Fig 7 Plastic zone size of aluminum (Zhang & Liu, 2011)

#### 4.2. Case II: Steel AISI 4340

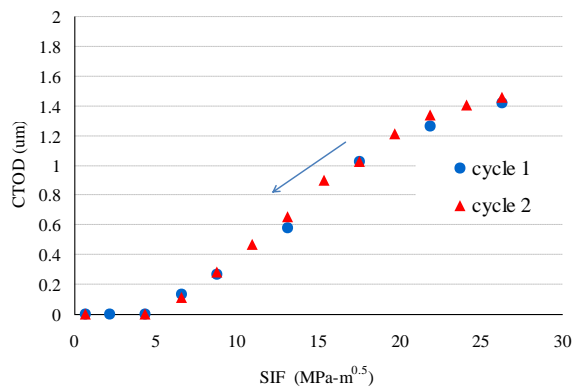
AISI 4340 steel is a heat treatable, low alloy steel containing nickel, chromium and molybdenum. It is widely used in industry, such as aircraft landing gear, power transmission gears and shafts and other structural parts (after heat treatment and/or surface hardening treatment). The chemical

composition of this material and its basic physical properties are listed in Table 3 and 4.

results for aluminum alloys. The testing results indicate no crack closure during the entire unloading duration.



(a) Selected SEM images of aluminum during unloading process



(b) CTOD during unloading process

Fig. 8 SEM images of aluminum in unloading process

The specimens are pre-cracked under load from 100 N to 1000 N at the frequency up to 5 to get an initial crack with length about 0.5 mm (excluding the notch length). A pre-cracked image is shown in Fig. 9.

After polishing, the average diameter of dark dots is about 15 μm, as shown in Fig. 10.

Then, the images are taken during the continuously loading or unloading process. In our test, the load profile starts from 100 N to 1000 N at step 50 N, as shown in Fig. 3. Finally, the plastic zone size variation during loading and unloading is obtained. In order to reduce the measurement noise, multiple tests have been performed. The result in Fig. 11 shows a good coherence of each test with the theoretic plastic zone size without consideration of plastic closure, i.e. the reversed plastic zone keeps increasing during the unloading process. This is very different compared to the

Element	Min	Max
C	0.38	0.43
Cr	0.7	0.9
Mn	0.6	0.8
Mo	0.2	0.3
Ni	1.65	2
P	0	0.035
Si	0.15	0.3
S	0	0.04

Table 3 Chemical composition of AISI 4340 steel

Elastic Modulus /GPa	Yield Strength /MPa	Tensile Strength /MPa
190~210	472.3	744.6

Table 4 Basic physical properties of AISI 4340 steel

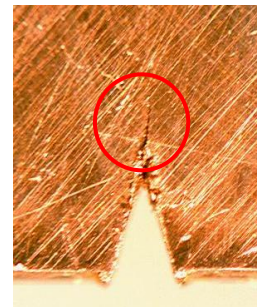


Fig. 9 Crack tip of steel specimen

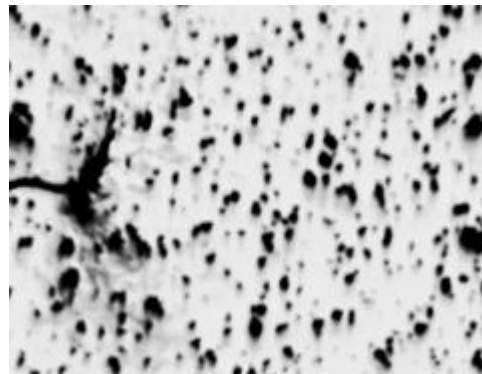


Fig. 10 Crack tip with markers for DIC process

In order to fully support this hypothesis, SEM images are taken one of the specimens and the CTOD in two unloading processes is obtained. The selected images are shown in Fig. 12(a) and the CTOD is shown in fig. 12(b). The images show that the crack edges do not contact to each other and the measured CTOD keeps growing during the whole unloading process according. The result proves that there is

no crack closure for ASNI 4340 steel under loading ratio 0.1 in the current investigation.

Compared with test on aluminum 7075-T6, ASNI 4340 steel does not have crack closure phenomenon under the loading profile with load ratio 0.1. Thus, it can be concluded that for different materials, crack closure does not always exist. This comparison also shows that the method proposed is a general method for crack fatigue mechanism investigation and model hypothesis verification.

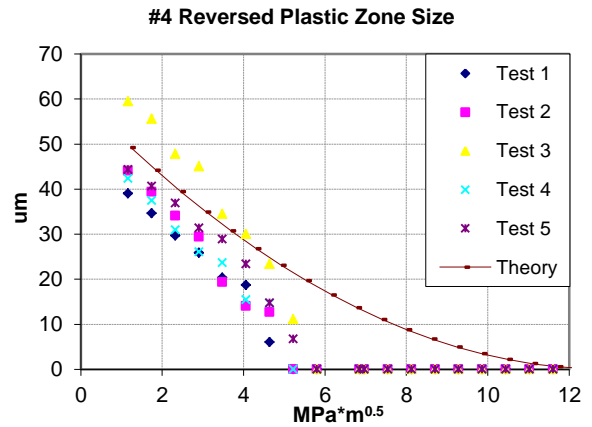
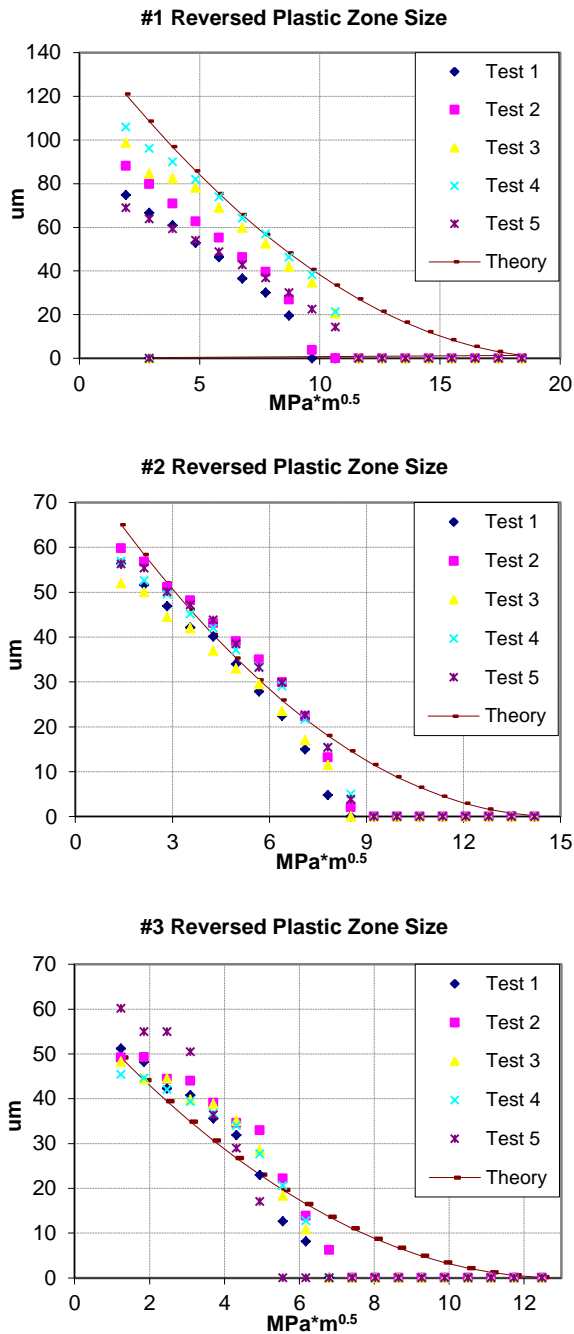
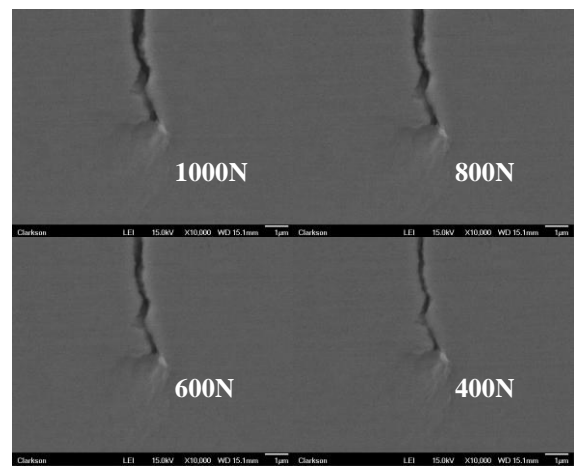
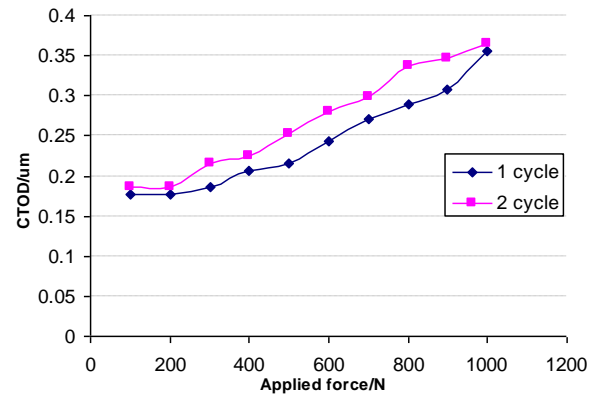


Fig. 11 Plastic zone size of steel specimens



(a) Selected SEM images during the unloading process



(b) CTOD during unloading process

Fig. 12 Crack tip condition of steel during unloading process

For aluminum 7075-T6, crack closure is observed and the opening stress is obtained. Thus it is suggested to use modified McClung's model (McClung & Sehitoglu, 1989)

to predict the crack growth rate provided by Wei (Zhang & Liu, 2011):

$$\rho_f = \begin{cases} 0 & K < K_{\text{open}} \\ \frac{1}{\pi} \left( \frac{K - K_{\text{open}}}{2\sigma_y + (\sigma_{\text{closure}} - \sigma_{\text{min}})} \right)^2 & K \geq K_{\text{open}} \end{cases} \quad (4)$$

$$\rho_r = \begin{cases} \frac{1}{\pi} \left( \frac{K_{\text{max}} - K_{\text{closure}}}{2\sigma_y} \right)^2 & 0 \leq K \leq K_{\text{closure}} \\ \frac{1}{\pi} \left( \frac{K_{\text{max}} - K}{2\sigma_y} \right)^2 & K_{\text{closure}} < K \leq K_{\text{max}} \end{cases} \quad (5)$$

where  $\sigma_{\text{closure}}$  is the crack closing stress and  $\Delta K_{\text{eff}} = F \cdot \Delta \sigma_{\text{eff}} \sqrt{\pi a} = F \cdot (\sigma_{\text{max}} - \sigma_{\text{closure}}) \sqrt{\pi a}$ . It can be noticed that Rice's model is a special case of McClung Model in that  $\sigma_{\text{closure}} = \sigma_{\text{min}}$ .

The two materials show quite different behaviors under the same experiment condition. One probable reason is the difference of ductility. For 4340 steel, the failure elongation could be 12.2~22.0%, but for aluminum 7075-T6, that is only 5~8%, much more brittle than the former one. This could cause the fracture surface in aluminum more smooth than steel, thus the crack surface could not contact to each other during the unloading process. With the different behaviors for these two materials, different crack prognosis methods can be verified or new prognosis methods could be proposed, such as the one in Zhang and Liu's paper (Zhang & Liu 2011).

## 5. CONCLUSION AND FUTURE WORK

This paper proposes a general methodology to verify the failure mechanism for physics-based mechanical damage prognosis. Testing under both optical microscope and SEM are discussed in detail. Imaging processing technique and advanced instrumentation provides a unique capability to investigate the mechanical damage behavior of structural materials.

Several conclusions can be drawn based on the current investigation:

- Multi-resolution in-situ testing and imaging analysis is powerful to explore the detailed mechanical deformation and damage accumulation behavior of investigated metals;
- Aluminum 7075 in the current investigation shows has crack closure under cyclic loadings and the reversed plastic zone stop growing after the crack closure
- Steel 4340 in the current investigation shows no crack closure and the plastic zone continues to grow during the entire unloading path

- Different mechanisms indicate that different prognosis models should be developed for these two materials, especially on the plastic zone calculation

The proposed methodology provides a general way to get the mechanical information of fatigue cracks. However, it does not only pertain to fatigue research, In the future, other metal failure mechanisms such as creep or brittle fracture mechanism will be studied using this method. In addition, composite materials failure mechanism such as delamination and matrix will also be studied using advanced 3D imaging technique, such X-ray computed tomography.

## ACKNOWLEDGEMENT

The research reported in this paper was supported by funds from Air Force Office of Scientific Research: Young Investigator Program (Contract No. FA9550-11-1-0025, Project Manager: Dr. David Stargel). The support is gratefully acknowledged.

## REFERENCES

- Bouami, D., & De Vadder, D. (1986). Detection and measurement of crack closure and opening by an ultrasonic method. *Engineering Fracture Mechanics*, 23(5), 913–920. DOI:10.1016/0013-7944(86)90101-3
- Campbell, G., & Lahey, R. (1984). A survey of serious aircraft accidents involving fatigue fracture. *International Journal of Fatigue*, 6(1), 25–30. DOI: 10.1016/0142-1123(84)90005-7
- Farrar, C. R., & Lieven, N. a J. (2007). Damage prognosis: the future of structural health monitoring. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 365(1851), 623–32. DOI: 10.1098/rsta.2006.1927
- Fleck, N., & Shin, C. (1985). Fatigue crack growth under compressive loading. *Engineering fracture mechanics*, 21(1), 173–185. DOI: 0013-7944/85
- Forsyth, P. J. E. (1962). A two stage process of fatigue crack growth. *Crack Propagation: Proceedings of Cranfield Symposium*. Cranfield (76–94), England: College of Aeronautics.
- Laird, C. (1967). The influence of metallurgical structure on the mechanisms of fatigue crack propagation. *Fatigue Crack Propagation, ASTM STP, 415*, 131–180.
- Macha, D. E., Corbly, D. M., & Jones, J. W. (1979). On the variation of fatigue-crack-opening load with measurement location. *Experimental Mechanics*, 19(6), 207–213. DOI: 10.1007/BF02324983
- McClung, R. C., & Sehitoglu, H. (1989). On the finite element analysis of fatigue crack closure—1. Basic modeling issues. *Engineering Fracture Mechanics*, 33(2), 237–252. DOI:10.1016/0013-7944(89)90027-1
- Paris, P. C., Gomez, M. P., & Anderson, W. E. (1961). A Rational Analytic Theory of Fatigue. *Trend in Engineering*, 13(1), 9–14.



- Rice, J. (1968). A path independent integral and the approximate analysis of strain concentration by notches and cracks. *Journal of Applied Mechanics*, 35, 379–386.
- Riddell, W., Piascik, R., Sutton, M., Zhao, W., McNeill, S., & Helm, J. (1999). Determining Fatigue Crack Opening Loads from Near-Crack Tip Displacement Measurements. *ASTM SPECIAL TECHNICAL PUBLICATION*, 1343, 157–174.
- Sadananda, K. (1999). Analysis of overload effects and related phenomena. *International Journal of Fatigue*, 21, 233–246. DOI:10.1016/S0142-1123(99)00094-8
- Shih, C. F. (1981). Relationships between the J-integral and the crack opening displacement for stationary and extending cracks. *Journal of the Mechanics and Physics of Solids*, 29(4), 305–326. DOI:10.1016/0022-5096(81)90003-X.
- Shih, T. T., & Wei, R. P. (1974). A study of crack closure in fatigue. *Engineering Fracture Mechanics*, 6(1), 19–32. DOI:10.1016/0013-7944(74)90044-7.
- Singh, D. S., Srivastav, A., Gupta, S., Keller, E., & Ray, A. (2009). Ultrasonic measurement of crack opening load for life-extending control of mechanical structures. *2009 American Control Conference* (210–215), June 10-12. Piscataway, NJ, USA: IEEE. DOI: 10.1109/ACC.2009.5160021.
- Wolf, E. (1970). Fatigue crack closure under cyclic tension. *Engineering Fracture Mechanics*, 2(1), 37–45. DOI: 10.1016/0013-7944(70)90028-7.
- Zhang, W., & Liu, Y. (2011). Plastic zone size estimation under cyclic loadings using in situ optical microscopy fatigue testing. *Fatigue & Fracture of Engineering*

*Materials & Structures*, 34(9), 717–727. DOI: 10.1111/j.1460-2695.2011.01567.x.

#### BIOGRAPHIES

**Jian Yang** is a Ph.D. candidate in the department of civil & environmental engineering at Clarkson University. He received his Bachelor's and Master's degrees in reliability & system engineering at Beihang University (Beijing University of Aeronautics and Astronautics), Beijing, China, 2008 and 2011 respectively. His research interests are fatigue propagation mechanics, simulation and test, structural health monitoring and prognosis.

**Wei Zhang** is a graduate research assistant in the department of civil and environmental engineering at Clarkson University. He received her B.S. degree in reliability engineering and M.S. degree in aerospace system engineering from Beihang University in China in 2005 and 2008, respectively. His research interests are fatigue analysis, structural dynamics, diagnosis and prognosis.

**Yongming Liu** is an associate Professor in the department of civil & environmental engineering. He completed his Bachelors` and Masters` degrees from Tongji University in China and obtained his Ph.D. at Vanderbilt University. His research interests include fatigue and fracture analysis of metals and composite materials, probabilistic methods, computational mechanics, and risk management. He is a member of ASCE and AIAA and serves on several technical committees on probabilistic methods and advanced materials.

# Reliability Growth Analysis of Satellite Systems

<sup>1</sup>Dr. John W. Evans, <sup>2</sup>Dr. Mark P. Kaminskiy, and <sup>3</sup>Mr. Luis D. Gallo Jr.

<sup>1,2,3</sup>NASA Goddard Space Flight Center, Greenbelt, MD, 20771, U.S.A

John.W.Evans@NASA.gov

Mark.P.Kaminskiy@NASA.gov

Luis.D.Gallo@NASA.gov

## ABSTRACT

A reliability trend/growth analysis methodology for satellite systems is suggested. A satellite system usually consists of many satellites successively launched over many years, and its satellites typically belong to different satellite generations. This paper suggests an approach to reliability trend/growth data analysis for the satellite systems based on grouped data and the Power Law (Crow-AMSAA) Non-Homogeneous Poisson process model, for both one (time) and two (time and generation) variables. Based on the data specifics, the maximum likelihood estimates for the Power Law model parameters are obtained. In addition, the Cumulative Intensity Function (CIF) of a family of satellite systems was analyzed to assess its similarity to that of a repairable system. The suggested approaches are illustrated by a case study based on Tracking and Data Relay Satellite System (TDRSS) and Geostationary Operational Environmental Satellite (GOES) data.

## 1. INTRODUCTION

The objective of this study is to develop a reliability growth analysis methodology applicable to *satellite systems*. A satellite system usually consists of many satellites successively launched during many years, and its satellites can belong to different satellite generations. For example, the United States National Environmental Satellite, Data, and Information Service (NESDIS) is now developing its fourth generation (gen.) of the GOES satellites. The GOES first satellite, GOES 1, was launched in 1975 and the latest, GOES 15, was launched in 2010 (see Table 2).

During the system life, its satellites can be in different states, like *active*, *in-orbit testing*, *failed*, *standby*, *retired*, etc. The satellite system reliability improvements are based on the analysis of anomalies (failures) observed on the in-orbit satellites, and the respective corrective actions can

usually be implemented only in the next and the following satellites to be launched. In other words, the traditional reliability growth "Test-Analyze-Fix" concept is not applicable to the on-orbit satellite systems, which makes the data model and data analysis of the satellite systems rather different.

## 2. DATA AND RELIABILITY GROWTH MODEL

A satellite system (SS) is considered. Let's assume that the SS currently consists of  $k$  satellites  $S_1, S_2, \dots, S_k$ , where  $S_1$  is the first (oldest) successfully launched satellite,  $S_2$  is the second satellite, . . . , and  $S_k$  is the latest successfully launched satellite. Let  $T_1, T_2, \dots, T_k$  denote, respectively, the cumulative times during which the  $S_1, S_2, \dots, S_k$  anomalies were recorded, and let  $N_1, N_2, \dots, N_k$  denote the random numbers of corresponding failures (anomalies). These data can be represented using Table 1.

The Crow-AMSAA model is suggested to apply for the SS reliability trend analysis. This model is the most popular reliability growth model. The model is used in Military Handbook 189 (MIL-HDBK -189 C, 2011). The model was applied in the following traditional form:

$$\lambda(t) = \lambda_0 \beta t^{\beta-1} \quad (1)$$

where  $\lambda(t)$  is the ROCOF for a given satellite,  $\lambda_0$  and  $\beta$  are positive parameters, and  $t$  is the satellite order number, so that the variable  $t$  takes on the following values: 1, 2, 3, 4, 5, . . . . Other choices of the independent variable  $t$  can be budget or other monetary or manpower resources spent to improve the satellite reliability. It should be noted that in the case of reliability growth, the parameter  $\beta$  should satisfy the following inequality:  $0 < \beta < 1$ . The model (1) is sometimes referred to as the Weibull process, because it coincides with the failure rate of the Weibull distribution.

---

Dr. John W. Evans et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Satellite	Time Interval	Number of Anomalies
$S_1$	$T_1$	$N_1$
$S_2$	$T_2$	$N_2$
...	...	...
$S_k$	$T_k$	$N_k$

Table 1. Satellite System Anomaly Data

### 3. DATA ANALYSIS

For each satellite of the system considered, the ROCOF estimate is calculated as

$$\hat{\lambda}(i) = \frac{N_i}{T_i} \quad i = 1, 2, \dots, k \quad (2)$$

where the estimator (2) is known as the *natural* estimator of ROCOF (Basu, A.P. & Rigdon S.E., 2000; Crowder, M. J., Kimber A. C., Smith, R. L., & Sweeting, T. J., 1991).

Assuming that ROCOF is constant (but different) for each satellite, it is clear that  $N_i$  is distributed according to the Poisson distribution with the mean equal to  $\lambda(i)T_i$ , where  $\lambda(i)$  is the unknown true value of ROCOF for  $i$ th satellite. If the number of the observed failures  $N_i$  is great enough, the distribution can be approximated by the Normal Distribution, having the same mean and the variance equal to this mean.

Based on the above considerations, the following regression model (3) is suggested for estimating the parameter of the Crow-AMSAA model (1)

$$\lambda(t_i) = \lambda_0 \beta t_i^{\beta-1} + \delta_i \quad (3)$$

where  $\delta_i$  is a normally distributed error with zero mean and the variance is inversely proportional to the number of the observed failures  $N_i$ ;  $t_i$  is the satellite order number, taking on the following values: 1, 2, 3, . . . . The model (3) is non-linear in the parameters regression model, where parameters  $\lambda_0$  and  $\beta$  should be estimated under the following restrictions:  $\lambda_0 > 0$  and  $1 > \beta > 0$ .

Another way to estimate the parameters of the reliability growth model (1) is to apply the Maximum Likelihood (ML) approach. For the data discussed above, the likelihood function  $L(\lambda_0, \beta)$  can be written as

$$L(\lambda_0, \beta) = \dots \quad (4)$$

$$= \prod_{i=1}^k \frac{\left( \int_0^{T_i} \lambda_0 \beta t_i^{\beta-1} dt \right)^{N_i} \exp \left( - \int_0^{T_i} \lambda_0 \beta t_i^{\beta-1} dt \right)}{N_i!} \quad (4)$$

$$= \prod_{i=1}^k \frac{\lambda_0^{N_i} \beta^{N_i} t_i^{N_i(\beta-1)} T_i^{N_i} \exp \left( - \lambda_0 \beta t_i^{\beta-1} T_i \right)}{N_i!} \quad (4)$$

and its logarithm as

$$\ln(L(\lambda_0, \beta)) = \sum_{i=1}^k N_i \ln(\lambda_0) + N_i \ln(\beta) + \dots \quad (5)$$

$$N_i(\beta - 1) \ln(t_i) + N_i \ln(T_i) - \lambda_0 \beta t_i^{\beta-1} T_i - \ln(N_i!) \quad (5)$$

Writing the first derivatives of (5) with respect to  $\lambda_0$  and  $\beta$  and equating them to zero, we arrive at the following system of non-linear equations for  $\lambda_0$  and  $\beta$ :

$$\frac{\partial}{\partial \lambda_0} (L(\lambda_0, \beta)) = \frac{1}{\lambda_0} \sum_{i=1}^k N_i - \beta \sum_{i=1}^k t_i^{\beta-1} T_i = 0 \quad (6)$$

$$\frac{\partial}{\partial \beta} (L(\lambda_0, \beta)) = \frac{1}{\beta} \sum_{i=1}^k N_i - \dots \quad (7)$$

$$\lambda_0 \sum_{i=1}^k T_i [t_i^{\beta-1} + \beta(\beta - 1)t_i^{\beta-2}] = 0 \quad (7)$$

which must be solved under the restrictions:  $\lambda_0 > 0$  and  $1 > \beta > 0$ .

### 4. CASE STUDY: TRACKING AND DATA RELAY SATELLITE SYSTEM

The Tracking and Data Relay Satellite System (TDRSS) is a network of satellites (each called a *Tracking and Data Relay Satellite* or TDRS) and ground stations used for space communications. The TDRSS space segment currently consists of nine on-orbit TDRSSs located in geosynchronous orbit, distributed to provide global coverage.

The available data on the TDRSSs are 327 NASA Spacecraft Orbital Anomaly Report System (SOARS) records related to the satellites of the first TDRS generation (A, C, D, E, F and G) and the second TDRS generation (H, I, J), listed in Table 2. It should be noted that there is much less data on TDRS H, I and J (only about 25 cumulative mission years) compared to the first generation, i.e., TDRS A, C, D, E, F and G (about 101.4 cumulative mission years).

The Crow-AMSAA model (1) and the data from Table 2 were used for the reliability trend analysis. The parameters of the reliability growth model were estimated as:  $\lambda_0 = 3.156917$  1/day and  $\beta = 0.006$ . The ROCOF estimates and the fitted Crow-AMSAA model are shown in Figure 1 below. The model provides a good fit to the data: the squared correlation coefficient  $R^2 = 0.963$ . Using the fitted model, the ROCOF for the future TDRS M was predicted as 0.00151/day. The predicted value indicates a 30% – 40% reliability growth for TDRS 13 (TDRS M) compared to TDRS 10 (TDRS J) in terms of ROCOF.

Generation (Gen.)	Satellite Name	Other Satellite Name	Launch Date	Last Record Date	Time Interval, days	Number of records	ROCOF ( $\lambda_{est}$ ) 1/day
1	TDRS A	TDRS 1	4-Apr-83	5/04/2006	8431	192	0.0228
1	TDRS C	TDRS 3	29-Sep-88	9/28/2004	5843	28	0.0048
1	TDRS D	TDRS 4	13-Mar-89	11/02/2010	7904	35	0.0044
1	TDRS E	TDRS 5	2-Aug-91	11/06/2004	4845	21	0.0043
1	TDRS F	TDRS 6	13-Jan-93	7/16/2006	4932	20	0.0041
1	TDRS G	TDRS 7	13-Jul-95	6/4/2009	5075	8	0.0016
2	TDRS H	TDRS 8	30-Jun-00	9/02/2010	3716	5	0.0013
2	TDRS I	TDRS 9	8-Mar-02	9/26/2010	3124	12	0.0038
2	TDRS J	TDRS 10	4-Dec-02	7/21/2009	2421	6	0.0025

Table 2. Data and Estimated Rate of Occurrence of Failures

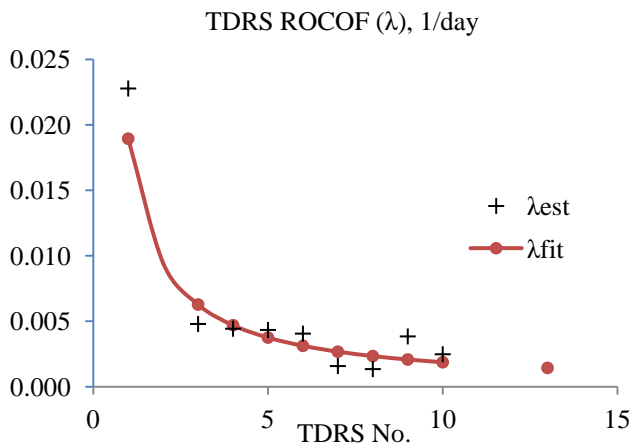


Figure 1. Estimated ROCOF and fitted reliability growth model. The extreme right point is the projected ROCOF for TDRS M (13).

### 5. RELIABILITY GROWTH MODEL WITH TWO VARIABLES

Earlier, we applied the power law (Crow-AMSAA) relationship to model satellite ROCOF dependence on the satellite order (operational) number. The relationship we are going to introduce below can be used to take into account a possible jump of ROCOF attributed to a new satellite generation, which is illustrated by the GOES ROCOF (see Figure 2 and Table 3).

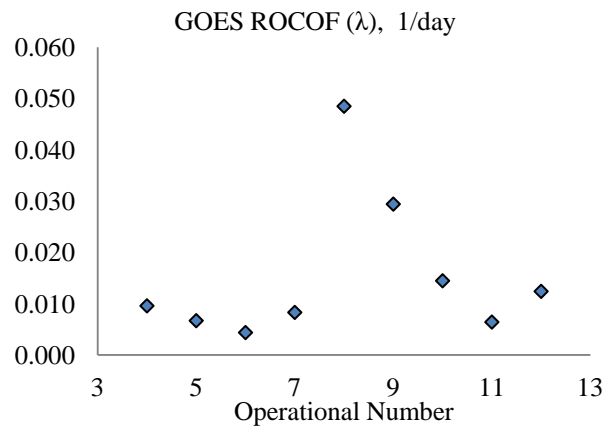


Figure 2. GOES ROCOF dependence on satellite operational number. The interval [4, 7] is the first generation of satellites; the interval [8, 12] is the second generation of GOES satellites.

Figure 2 shows a significant jump in ROCOF between the first generation and the second generation of GOES. This increase in ROCOF of the second generation can be explained by more complex satellite design and functions. The figure also reveals a minor ROCOF increase for each last satellite of the first generation and the second generation. The GOES 7 increase in ROCOF compared to its predecessor GOES 6 can be attributed to the GOES 7 new feature – it was the first GOES satellite capable of detecting 406 MHz distress signals from emergency beacons carried aboard aircraft and vessels and sending them to ground stations. In its turn, the GOES 12 increase in

ROCOF (compared to GOES 11) can be attributed to the new instrument -- GOES 12 was the first satellite to carry a Solar X-Ray Imager (SXI) type instrument.

In order to take into account a ROCOF dependence on the satellite generation, the following model is suggested:

$$\lambda(t_i, T) = \lambda_0 t_i^{\beta_1} T_j^{\beta_2} \tag{8}$$

The model has the following two independent variables-- operational number ( $t_i = 4, 5, \dots, 12$ ) and a dummy variable,  $T_j$ , ( $j = 1, 2, \dots, J$ ), which is the satellite generation order number. The dummy variable  $T$  value is  $e$  for the first generation, i.e.,  $T_1 = e$ , and  $T$  takes on the value  $e^0 \equiv 1$  for the second satellite generation, i.e.,  $T_2 = 1$ . The choice of these values becomes obvious if we take the natural logarithm of (8) in order to make the model linear:

$$\ln(\lambda(t_i, T)) = \ln(\lambda_0) + \beta_1 \ln(t_i) + \beta_2 \ln(T_j) \tag{8.1}$$

It is clear that the transition from the first generation to the second generation changes the intercept of the above linear dependence by  $\beta_2$  because of a unit change in  $\ln(T_j)$ , i.e.,  $\ln(T_1) - \ln(T_2) = 1$ . The variable  $T_j$  can be called the *generation code*. The available GOES ROCOF data needed to fit the above model are given in Table 3.

GOES Gen.	Gen. Code (T)	ln(T)	GOES Oper. Number (t)	ROCOF ( $\lambda_{est.}$ ) 1/day
1	EXP(1)	1	4	0.00956
1	EXP(1)	1	5	0.00667
1	EXP(1)	1	6	0.00435
1	EXP(1)	1	7	0.00828
2	EXP(0)	0	8	0.04848
2	EXP(0)	0	9	0.02940
2	EXP(0)	0	10	0.01445
2	EXP(0)	0	11	0.00638
2	EXP(0)	0	12	0.01239

Table 3. GOES History and Estimated Rate of Occurrence of Failures (ROCOF)

Using the above data, the parameter estimates of model (8.1) are given in Table 4.

	Parameter Estimate	Std. Err.	t(6)	p-level
$\ln(\lambda_0)$	0.4017	2.3920	0.1679	0.8722
$\beta_2$	-2.1074	0.7308	-2.8839	0.0279
$\beta_1$	-1.9407	1.0380	-1.8697	0.1107

Table 4. Regression analysis summary of model (8.1)

As it follows from Table 4, the parameter  $\ln(\lambda_0)$  is statistically insignificant, so that our model (8) can be written as:

$$\lambda(t_i, T) = t_i^{\beta_1} T_j^{\beta_2} \tag{8.2}$$

The fitted model is shown in Figure 3.

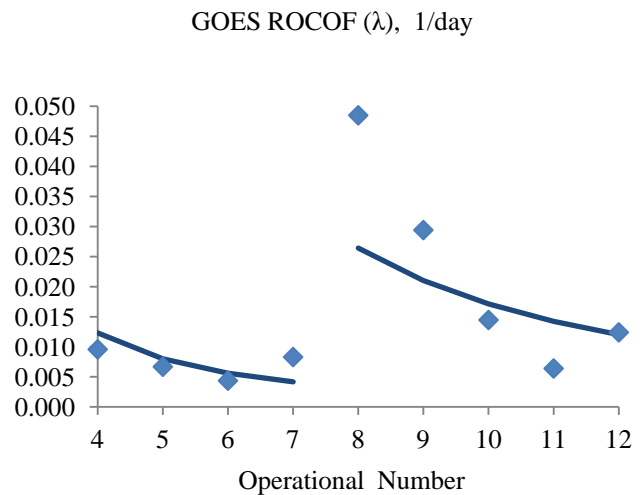


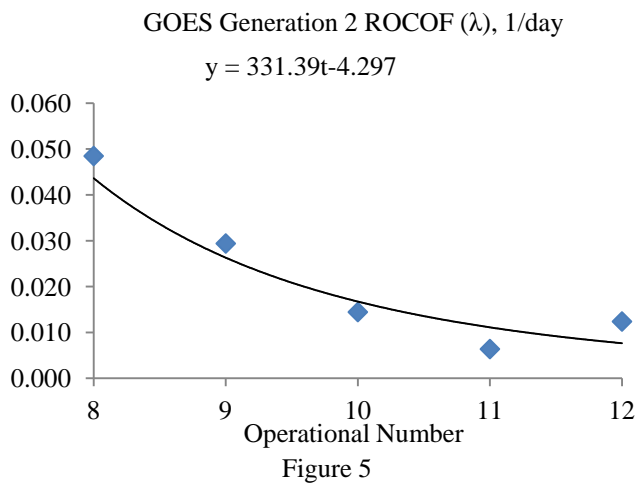
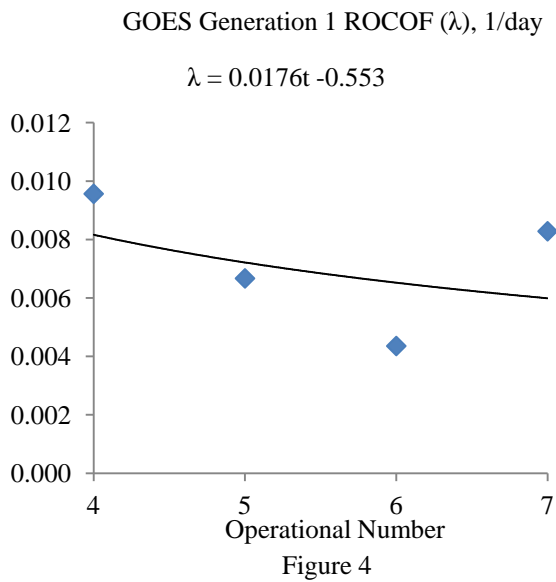
Figure 3. The GOES ROCOF and fitted model (8.2)

Based on the model and data, the jump in values of ROCOF  $\frac{\lambda(8, T_2)}{\lambda(7, T_1)}$  is about 8.

In order to compare the reliability growth rate for GOES generations 1 and 2, the following ROCOF model was fitted for each generation:

$$\lambda(t_i) = \lambda_0 t_i^\beta \tag{8.3}$$

The fitted models are shown in Figures 4 and 5.



### 6. FAILURE TIME OCCURRENCES DURING EACH MISSION

Based on its cumulative intensity function (CIF), each satellite in a system of satellites, such as GOES, can be considered as a repairable system. The cumulative intensity function of an idealized repairable system is depicted in Figure 6. At the beginning of mission, the CIF is concave down (has a decreasing derivative (ROCOF)). This part of system mission lifetime corresponds to the reliability growth. Then CIF becomes approximately linear, which corresponds to constant in time ROCOF and normal (from reliability standpoint) system operation. At the end of system life, the CIF becomes concave up, corresponding to increasing ROCOF, and this part of the system mission lifetime corresponds to the reliability deterioration (aging).

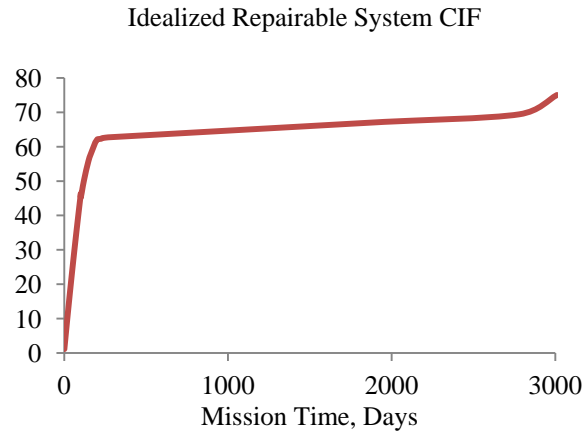
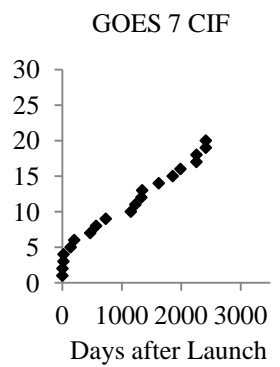
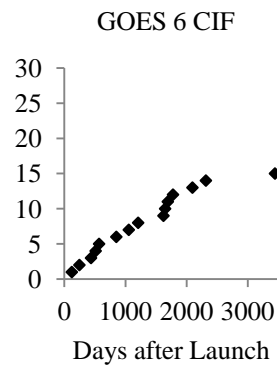
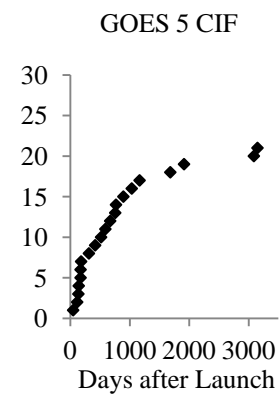
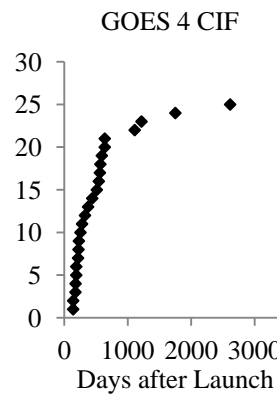


Figure 6. Cumulative Intensity Function of Idealized Repairable System

Figures 7a through 7i display the real CIF for a variety of GOES missions. These cumulative intensity functions have shapes similar to the idealized CIF.



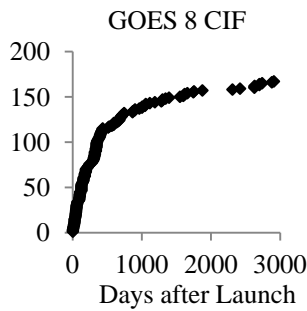


Figure 7e

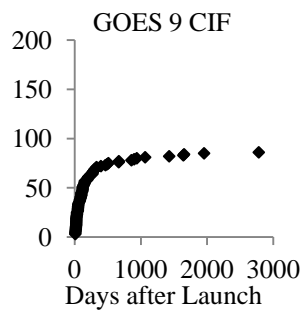


Figure 7f

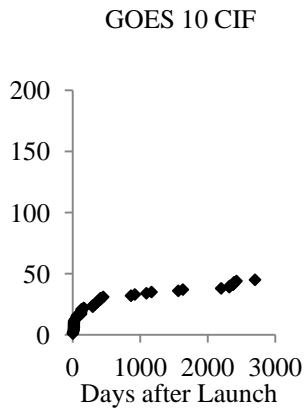


Figure 7g

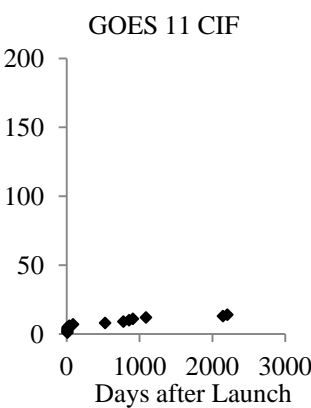


Figure 7h

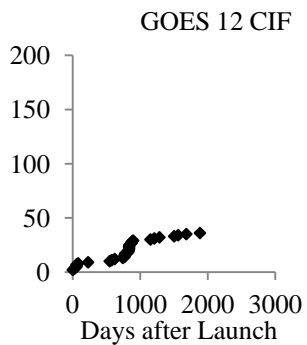


Figure 7i

## 7. CONCLUSION

In this study we set out to suggest a reliability trend/growth analysis methodology for satellite systems. We used the number of recorded anomalies for a given satellite mission over a given time period of operation as the data to measure this growth. Using this data, we modeled reliability growth as both a function of time and as a function of both time and satellite generation. Finally, we observed trends in ROCOF over a single satellite's operational lifetime and discussed the implications of all the observed trends on the evolution

of satellite systems. In order to do this, we assumed that an anomaly entry in the SOARS database corresponded to a failure of the given satellite, since anomalies are positively correlated with failures. Such an assumption is reasonable as long as the model used to fit the data is not expected to predict/measure the number of actual failures of a given satellite system.

We can model reliability growth across multiple satellite generations in a satellite system with a Crow-AMSAA model. This model is a good fit, having a squared correlation coefficient that is close to one ( $R^2 = 0.963$ ). The fitted model indicates that there is 30% – 40% reliability growth for the TDRS 13 (TDRS M) satellite compared to TDRS 10 (TDRS J), in terms of ROCOF. The overall trend of ROCOF decrease with time implies an improving level of reliability over time and thus reliability growth in the TDRS family of spacecraft. These results are intuitive, since each satellite generation is relatively similar in design to the previous one, allowing for consecutive generation designs to be more refined.

We can model reliability growth across multiple satellite generations in a satellite system with greater accuracy after a slight modification to the Crow-AMSAA model. This modification involves introducing a dummy variable,  $T_j$ , ( $j = 1, 2, \dots, J$ ), which represents the satellite generation order number. This modification allows the model to capture any major generational changes in satellite system ROCOF data due to new technologies. The reliability growth of the model fitted to GOES satellite data is greater/more pronounced among the newer generation of GOES satellites and is able to capture and explain the radical change in ROCOF data corresponding to a significant change in technology introduced by the second generation of GOES satellites, beginning with GOES 8. This model provides a better fit than would have been possible with the single variable Crow-AMSAA model due to its ability to capture the inflection introduced by GOES 8.

We considered the plausibility of considering satellite systems, such as GOES, as repairable systems. Such systems experience a rapid increase in the reported number of failures over an initial period of operation, and maintain a fixed, less sharply increasing rate of failures for an extended period of operation, until finally the rate of failures increases again towards the end of system life (i.e. the bathtub curve effect). This turned-out to be a plausible consideration, since the observed CIF of each of the GOES family systems displayed some, if not all, of these repairable system characteristics.

We can improve the current models by introducing a Bayesian prior distribution over its parameters (i.e.  $\lambda_0, \beta$ ), considering them as random variables, and employing Bayesian inference, as opposed to classical Maximum Likelihood Estimation. All of these considerations should be made in future studies of these data.

## REFERENCES

- Rigdon, S. E., & Basu, A. P. (2000). *Statistical methods for the reliability of repairable systems*. New York: Wiley.
- Castet, J., & Saleh, J. (2009). Geosynchronous Communication Satellite. *27th IET/AIAA international communications satellite systems conference 2009: (ICSSC 2009), Edinburgh, United Kingdom, 1-4 June 2009*. (p. 431). Stevenage: IET.
- Crowder, M. J., Kimber A. C., Smith, R. L., & Sweeting, T. J. (1991). *Statistical analysis of reliability data*. London: Chapman & Hall.
- Military Specification (MIL)-HDBK-189C, Military Handbook: Reliability growth management (DOD, 2011).
- Saleh, J. H., & Castet, J. (2011). *Spacecraft reliability and multi-state failures a statistical approach*. Chichester, West Sussex, UK: John Wiley.
- Seber, G. A., & Wild, C. J. (1989). *Nonlinear regression*. New York: Wiley.

**Dr. John Evans** is a Senior Aerospace Engineer in the Reliability and Risk Assessment Branch of the NASA Goddard Space Flight Center. He is an internationally recognized expert in reliability growth and its application to complex systems. John is the author or coauthor of several books and numerous conference and journal articles on electronic system reliability, failure analysis and manufacturing technology.

**Dr. Mark Kaminskiy** is a Principal Reliability Engineer at Ares, providing Reliability Analysis support to the NASA Goddard Space Flight Center. Dr. Kaminskiy has served as the chief statistician at the Center of Technology and Systems Management, University of Maryland (College Park). He possesses extensive experience as a researcher in reliability engineering, life data analysis, and risk analysis of engineering systems and has conducted numerous research and consulting projects funded by the government and industrial companies. He has taught several graduate courses on reliability engineering at the University of Maryland and is the author or coauthor of over 50 publications in journals, conference proceedings, and reports. Dr. Kaminskiy possesses an extensive background in parametric and nonparametric distribution estimation, multivariate regression modeling for reliability applications, accelerated life testing, and probabilistic damage modeling.

**Mr. Luis D. Gallo Jr.** is an Aerospace Engineer in the Reliability and Risk Analysis branch of the NASA Goddard Space Flight Center. He holds a Bachelors of Science Degree in Electrical Engineering from Florida International University and his research interests include Bayesian inference in graphical models of spacecraft systems and machine learning as applied to reliability and risk analysis.



# Multi-turbine Associative Model for Wind Turbine Performance Monitoring

Onder Uluyol<sup>1</sup> and Girija Parthasarathy<sup>2</sup>

<sup>1,2</sup>*Honeywell International, Inc., Golden Valley, MN, 55422, USA*

*onder.uluyol@honeywell.com*  
*girija.parthasarathy@honeywell.com*

## ABSTRACT

Comparing the performance parameters of a set of wind turbines in a single region will provide insights that prevent raising unnecessary alarm, while confirming actual faults. Wind turbines operating in a wind farm experience similar operating and environmental conditions that could indicate either normality for that group or failures that manifest in those conditions. These norms and failures are an orthogonal set of data rich in information that can be utilized in performance monitoring algorithms to supply better prediction accuracy and low false positives. In this paper, we describe the use of an associative model (AM) for fault detection in a population. An associative model maps system parameters to an identical set of virtual parameters. The AM-based approach can be used to capture the underlying correlation of an observable system, such as performance parameters of a set of wind turbines in a wind farm. The residuals between the model output and the input can then be used to detect anomalies and isolate faults.

## 1. INTRODUCTION

A modern wind farm can hold hundreds of wind turbines at a site, located as close as several hundreds of meters from each other. They are usually situated in remote locations and operate under severe environments atop 60-90 m towers. It is no surprise that operations and maintenance costs of wind turbines run high. One of the biggest drivers of maintenance cost is unscheduled maintenance due to unexpected failures. Continuous performance monitoring of wind turbine health for automated failure detection can reduce maintenance costs by detecting failure pre-cursors before they reach a catastrophic stage and by keeping turbines operating at higher efficiencies.

Typical performance or health monitoring of a machine involves monitoring a few performance parameters over time to see changes with respect to its own history or

manufacturer provided baseline. It has been successfully used in the process and aerospace industries [Bell & Foslien, 2005; Gorinevsky, Dittmar & Mylaraswamy, 2002; Kim & Mylaraswamy, 2006]. In the wind industry, fault detection methods include anomaly detection based on neural network models of normal operating modes [Zaher, McArthur, and Infield 2009]; power-curve-based performance monitoring analytic [Uluyol, Parthasarathy, Foslien & Kim, 2011]; data mining techniques based on wind speed and power output [Kusiak, 2011]; and classification methods of clustering and principal components analysis [Kim, Parthasarathy, Uluyol, Foslien, Sheng & Fleming, 2011].

An alternative, complementary practice would be to monitor performance parameters with respect to similar machines in similar operating conditions. A wind farm offers such an opportunity. This opportunity converts the inherent complexity of monitoring a large number of wind turbines individually into an asset that increases the robustness of fault and performance monitoring systems for all wind turbines in a wind farm.

A study by Ye et al. (2010) presented a method for detecting wind turbine anemometer failures based on the difference in measured wind speeds between a pair of closely situated wind turbines. However, in this study, the method is a comparison of only two wind turbines and operates on accumulated week's worth of data. The present work explores the use of an associative model (AM) to capture the relationship among several wind turbines on a farm to detect anomalous conditions. The AM is applied to high resolution time-series data from seven wind turbines.

## 2. FAULT DETECTION USING ASSOCIATIVE MODELS

An associative model maps system parameters to an identical set of virtual parameters. The AM-based approach can be used to capture the underlying dynamics of an observable system, such as performance parameters of a set of wind turbines in a wind farm. The residuals between the model output and the input can then be used to detect anomalies and isolate faults. See [Uluyol 2001 and Uluyol

---

Onder Uluyol, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

2003] for previous applications of this approach for in-range sensor fault detection, isolation and recovery.

When applied to sensor data, AM captures analytical redundancy among sensors and maps the readings from a group of correlated sensors into an estimation set for an identical group. When an appreciable sensor fault is detected, the associated model estimate diverges from the actual sensor reading. Associative models in the form of an auto-associative neural network (AANN) [Kramer 1992] have also been applied for sensor validation in nuclear power plants, chemical process plants, and propulsion turbine engines.

Our application differs from conventional use of neural networks in that we employ a neural network as a model of the system that maintains dependencies among parameters of interest. A fault in our approach is a break in this overall correlation rather than a deviation in an individual parameter. The iterative associative model approach for fault isolation has been demonstrated with actual flight data collected from sensors installed on multiple types of turbine engines for fixed wing aircraft and helicopters.

### 3. WIND FARM APPLICATION

When applied to a wind farm, AM captures the performance correlation among the modeled wind turbines (see Figure 1). In this application of AM for wind farms, we explore the use of wind turbine power produced as the performance parameter. Many factors affect wind turbine performance. In applying an associative model, the analysis should include careful filtering of the data, both in training the model and in deploying it.

A wind turbine may exhibit anomalous behavior compared to its neighboring wind turbines for a number of reasons. It could be under repair or be subject to curtailment while others are operating normally. Or, the other turbines could be subject to curtailment while one is operating normally. These conditions can be obtained either through status parameters captured as part of SCADA (Supervisory Control And Data Acquisition) data or be detected using filters based on simple statistics on measured power.

The factors that affect the dynamics of wind turbine performance, and hence present an opportunity for the use of the AM approach, include location effects, park-wide

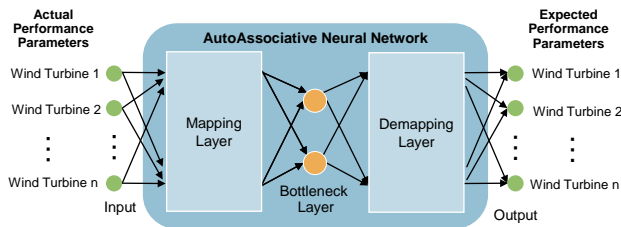


Figure 1. Wind turbine correlation mapping using AANN

control effects, and slowly-progressing fault in one of the monitored wind turbines. In a wind farm, wind turbines can be situated as far as 100 m apart from each other facing the wind direction and as far as 500 m apart along the wind direction. In order for all wind turbines to be under the same wind regime, the wind direction should be stable for a period of time that is greater than the wind turbine response time. While the wind direction is stable, the wind turbines don't necessarily produce exactly the same amount of energy; however, but we can correlate production to their location in the wind farm and capture this correlation in the associative model. Similarly, wind turbines may respond differently but consistently for their locations when the wind farm operator applies park-wide controls. These correlations can also be captured in the associative models.

After correlations among wind turbines for stable wind regimes and common control operation regimes are established, the associative model can be used to detect any deviation from the expected behavior.

#### 3.1. SCADA Data Description and Access

We obtained archived wind turbine SCADA data from three wind parks. The large data set, maintained in a MatrikonOPC Desktop Historian, includes 16000+ tags, and spans a period of about 6 months.

Before developing the algorithms, we identified the initial parameters of interest. The flat set of 16000+ tags was organized into a metadata list so we could parse the data appropriately. We also established a procedure for accessing the data with MATLAB®, which is our development environment. We used MathWorks® OPC Toolbox™ for this purpose. The tags were classified, and tags belonging to one wind farm and their data were exported to MATLAB®. The tags in this farm looked most meaningful for inclusion in a multi-turbine diagnostics algorithm for tag consistency (i.e., the same tags are available for most wind turbines in the park) and relevancy (i.e., the tags are useful for performance analysis).

#### 3.2. Data Pre-processing and Selection of Wind Turbines

Our objective was to select a representative set of turbines for developing and demonstrating the multiple wind turbine fault diagnoses. The wind park IDs in the tag names of the data were first fixed to eliminate discrepancies and redundant tags. Based on the wind turbines associated with one particular meteorological tower (MET) (502 marked with a red star in Figure 2) and their representative location in the geography around the tower, we chose seven wind turbines for further data analysis. These turbines are numbered T075, T081, T098, T104, T115, T118 and T127. There are about 30-50 wind turbines installed for each MET. The area where the turbines are installed forms a trapezoid shape behind each MET. There were 49 turbines

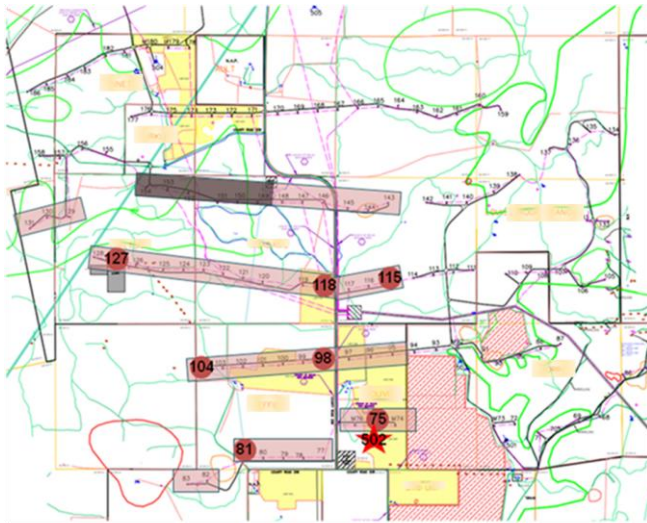


Figure 2. Location of selected wind turbines on the wind farm

installed in four rows within the trapezoid for MET #502. We selected two turbines from the first row, two from the second row, and three turbines from the third row for this analysis. The last row was omitted because of missing data.

The data from these turbines show that the tags have different sampling rates, and recorded parameters are not synchronized with each other. For example, wind turbine power, wind speed, and temperature are recorded at different timestamps and have differing timestamp intervals. The data statistics of wind turbine power produced, wind speed and ambient temperature are shown in Table 1. Notice that while the turbine power and speed may be recorded every second and the ambient temperature every 2 seconds, the median interval varies a lot from 5 seconds for power to 500 seconds for the ambient temperature. This results in having almost 300 times as many data points for power as the ambient temperature, and necessitates re-sampling for synchronizing data.

We reviewed several tags for filtering data to create a baseline data set. These tags included Operating, Faulted, and State\_fault, among others. To obtain set interval time-synchronized data, each tag was re-sampled at a higher frequency using a proprietary technique (patent pending), and means were computed for the set interval (1-min, 5-min or 10-min) for periods common to the tags of interest.

### 3.3. Analysis Using Power Curve Analytic

To detect correlations and variations for the same periods, the synchronized data from the seven selected wind turbines were analyzed according to the power curve analytic (Uluyol, Parthasarathy, Foslien and Kim, 2011). The power curve for the wind turbines was obtained from the operator. For each of the wind turbines, we calculated residuals from the power curve, for all data, with a 30-day moving window

	Power	Windspeed	Ambient temperature
Min interval (sec)	1	1	2
Median interval (sec)	5	20	500
Mode (sec)	5	5	100
Max interval (hr)	4.33	7.99	8
No of samples	2083426	424568	7331

Table 1. Data Statistics

and 1-day interval or frequency of calculation. The residuals were further processed to obtain wind speed bin-based averages, standard deviations, skewness, and kurtosis. Our analysis reveals that a correlation exists amongst the turbines in terms of power production, and with a suitable multiple turbine algorithm, anomalous conditions and faults may be detected. The plots in Figure 3 and Figure 4 show the skewness and kurtosis statistics. The seven wind turbines are represented in rows, with the columns representing the dates 25 days apart.

We observe that, in the time periods ending on Jan 9 and Feb 3, skewness and kurtosis for each wind turbine have shapes deviating from zero values across the wind speed bins. Since all wind turbines show this behavior for these time periods, and since the skewness is measured with respect to the population mean (rather than the nominal power curve), we speculate that this behavior could be found in normally operated wind turbines at this park. It is also possible that the number of samples in these time periods may have skewed the statistics.

These plot sets in Figure 3 and Figure 4 are an important visualization tool for comparing wind turbines in terms of broad parameter values and spotting anomalies. They provide the means to explain any individual power curve analytic alarms that could have been raised in the vicinity of the dates in the first two columns, when there seems to have been controlled power curtailment.

This part of the analysis leads us to conclude that wind turbine performance is strongly correlated across the wind farm. Additional automated analytics with the associative models approach, which models the correlation at a much finer timescale, could be beneficial.

### 3.4. Wind Farm Data

A set of normal operating data under nominal conditions is extracted to train an associative model. Normal operating data are the data from periods where no known fault is present in the records. The nominal conditions are defined based on power magnitude and variation, and wind direction variation. Figure 5 shows the power profile of the seven wind turbines during the six-month period between September 2009 and February 2010. Notice that the bulk of the data are recorded between the red vertical lines indicating the 250 kW and 1250 kW levels. The start-up and

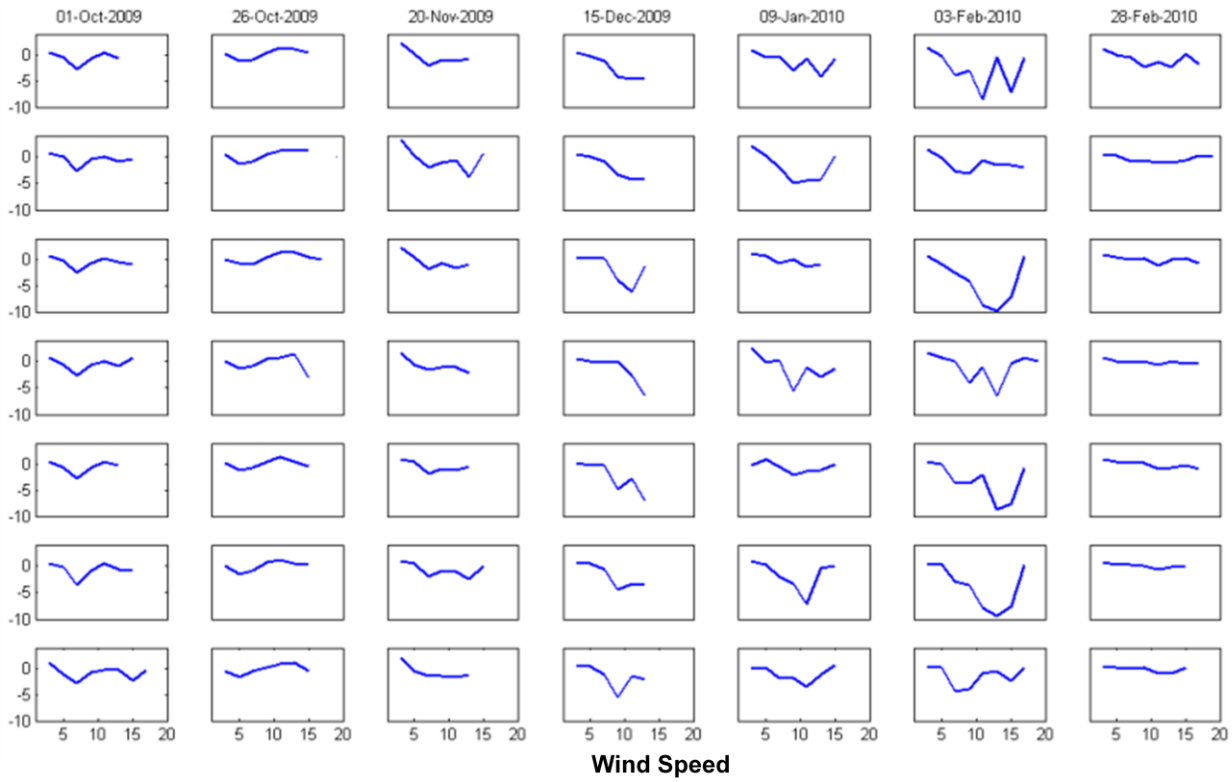


Figure 3: Skewness of power residuals in 30-day windows for 7 wind turbines (shown in rows)

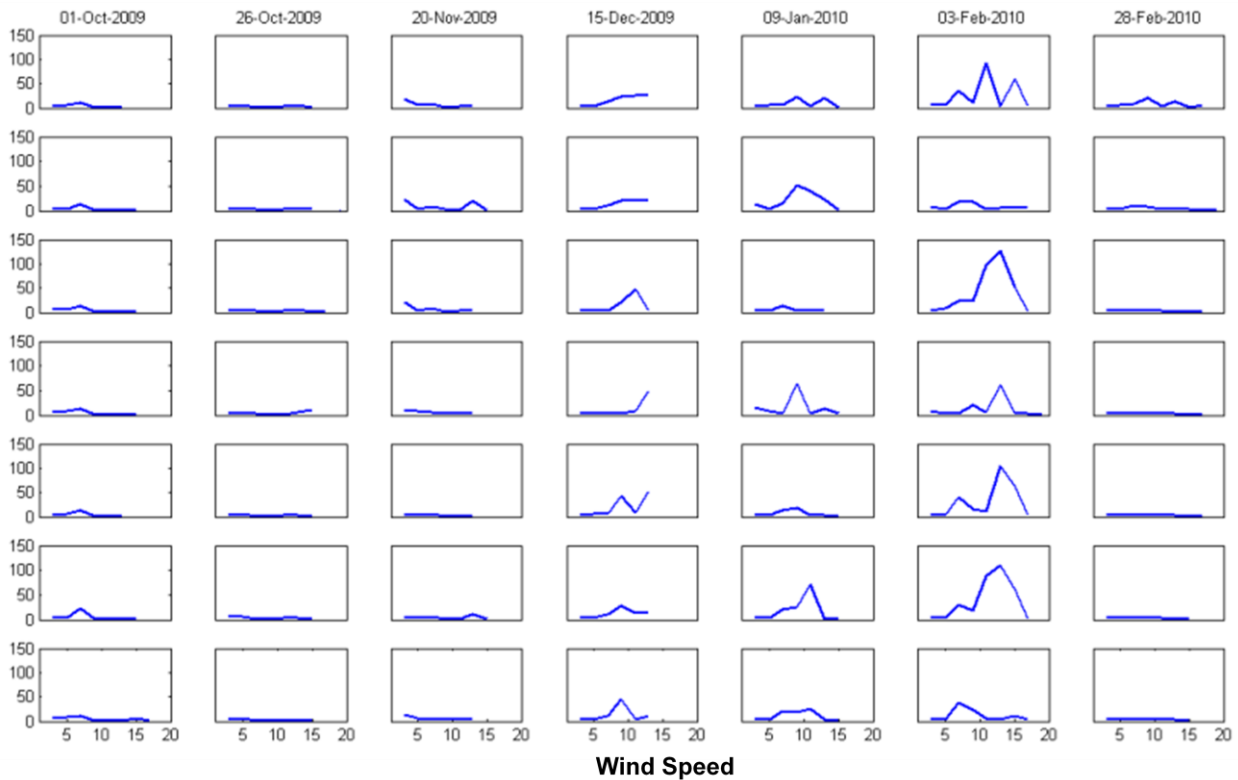


Figure 4: Kurtosis of power residuals in 30-day windows for 7 wind turbines (shown in rows)

the high power regions that fall outside of these limits contain highly non-linear operation periods and are excluded from the training set.

Figure 6 shows the variation in power in terms of the standard deviation calculated for each sampling interval. Notice that it shows a normal distribution superimposed with samples having very little or no variation. The samples that show very little variation in the wind turbine group correspond to shutdown, full saturated power, or curtailed power level conditions. For the baseline training, we exclude these points. The samples at the other end of the distribution mark very high variation conditions, and are similarly excluded from the baseline training. We chose the region between 50kW and 300 kW standard deviation as the representative region for a baseline nominal operation and

used it for the AM training.

Figure 7 and Figure 8 show the wind direction and its variation. Wind direction is measured at the MET towers. Two sets of measurements were taken at different heights at each tower. Figure 7 shows Wind Direction 1 measurement at one of the towers (MET 2 in Park 2) for about a 2.5 week period as a function of time. Notice that while the wind direction is stable most of the time as indicated by small standard deviation values plotted in green in Figure 7, the wind direction varies greatly about 8 times during the period covered as indicated by the sharp rises in standard deviation. Figure 8 is a histogram of the standard deviation of the wind direction. For the training set, we excluded the high variation points and only included samples below 10 degrees, indicated by the vertical red line.

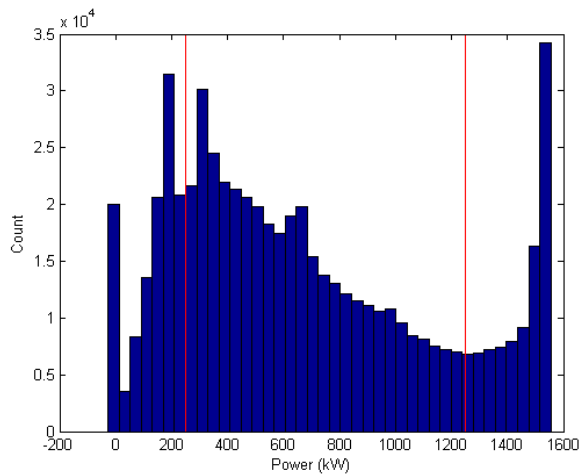


Figure 5: Power profile of 7 wind turbines during 6 months of operation

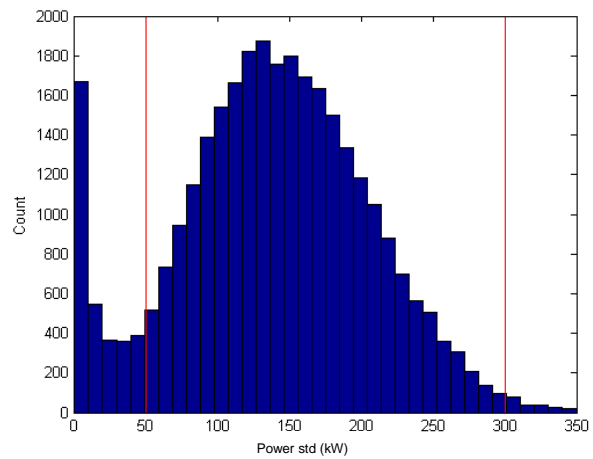


Figure 6: Standard deviation in power among 7 turbines for each sampling instance

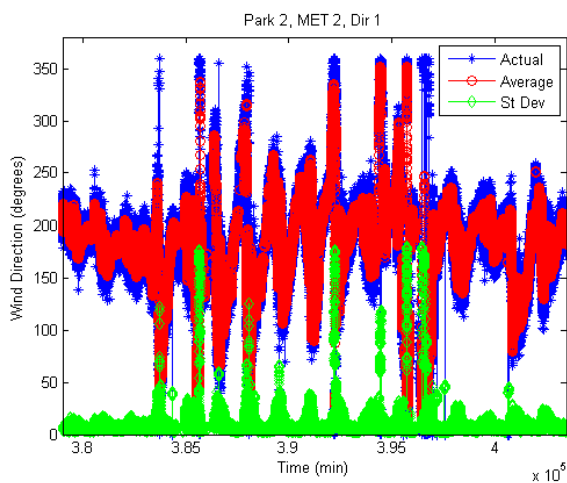


Figure 7. Wind direction at MET2

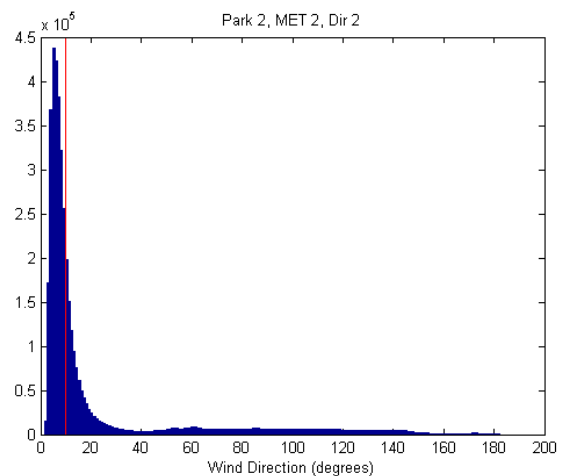


Figure 8. Variation in wind direction

#### 4. RESULTS AND DISCUSSION

We first tested our approach using the complete data set to see how often the associative model approach would indicate an anomaly. This data set was filtered only for stable wind direction as measured by both Direction 1 and Direction 2 and included data from only those instances where the variation in wind direction was below a set threshold as shown in Figure 8. Although the ground truth was not available for each anomaly, an initial assessment for the efficacy of the approach can be made based on the distribution of the indicated anomalies.

Figure 9 and Figure 10 show the associative model results for two select months from the period of September, 2009 to February, 2010. The green, yellow, and red circles in the figures indicate healthy, low-confidence and unconfirmed anomaly, and high-confidence and confirmed anomaly conditions, respectively. The input is made up of data sampled at 1-min intervals, and each sample has a corresponding output indicated by one of the three colored circles.

The yellow indicator may appear in all turbines for each unconfirmed anomaly, while the red indicator is limited to only one turbine. The unconfirmed anomaly is triggered whenever a high residual occurs in any of the turbines, which indicates a mismatch between the associated model output and the measured input signal. However, for an anomaly to be confirmed, a certain amount of persistency in the fault is expected. This confirmation is achieved by a counter implemented as a leaky-integrator. Once the fault persists for a set number of samples, then a hypothesis

testing procedure iteratively isolates the fault to one particular wind turbine.

As expected, the monthly data results show that turbines are operating without any indication of an anomaly most of the time. Moreover, when an unconfirmed fault is detected, it does not always persist long enough to trigger a confirmed fault. Notice that once the fault is confirmed, the isolation to a single turbine is sometimes achieved, as is the case on Sep 4 data in Figure 9; or not achieved, as in the case of Jan 5 data in Figure 10. The latter case shows that further tuning of the model may be necessary.

#### 4.1. No Fault Data Set

Besides the faulty data cases, performance assessment of anomaly detection methods needs to include no-fault cases. Especially in a condition-based maintenance approach for remotely located wind turbine applications, the false alarm performance of anomaly detection algorithms is highly critical, as the wind farm operator would incur high cost for an unnecessary maintenance action.

Figure 11 shows the associative model results for a data set made up of nominal power and wind speed. The data set spans six months from September, 2009 to February, 2010. Notice that the model detected a mismatch between the actual and expected output only two times in this data set, as indicated by two sets of yellow circles at 55 and 145 days from September 1, 2009. Neither of these indications persisted, and the associative model output quickly reverted back to healthy. This shows that the approach is very robust with respect to inherent variations in the data and does not produce false alarms.

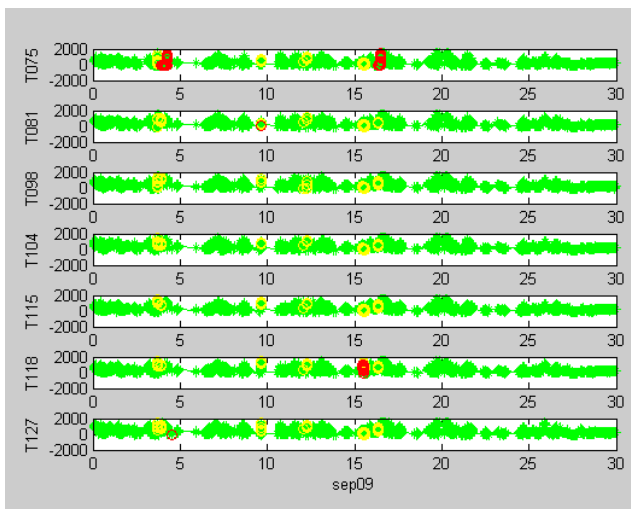


Figure 9. Associative model results for Sep 2009 data

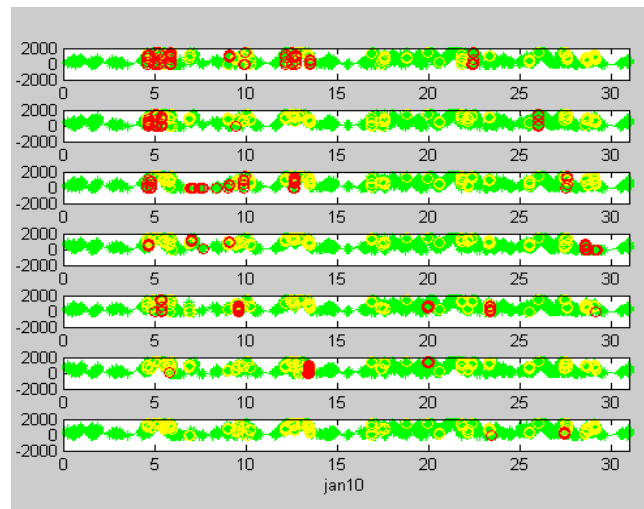


Figure 10. Associative model results for Jan 2010 data

### 4.2. Ice Condition Data

The SCADA data set includes an aggregate ice condition indicator. This parameter shown in Figure 12 indicates the sum of wind turbines in which an ice condition is detected at any given time. An ice condition denotes the possibility of icing based on low temperature and high humidity—not actual icing. The number of turbines is presented in terms of the percentage of total number of wind turbines in this particular wind park.

During the six-month period for which data is available, a high number of ice condition detections occurred on four occasions: Dec 7–8, 2009, Dec 29–30, 2009, Jan 7–8, 2010, and Jan 28–29, 2010. On these dates, more than 85 percent of the wind turbines out of 155 wind turbines in Park II experienced ice conditions.

Figure 13 and Figure 14 show the anomaly detection results around the four periods of ice conditions. The semi-transparent blue rectangles on the figures indicate the beginning and end of the periods of ice condition. The available SCADA data for the first three periods is very sparse; yet, the associative model was able detect many anomalies in all three cases. The Jan 28–29, 2010 case is preceded with continuous data and followed by no data for about a day. During this fourth ice condition period, all turbines are shown as producing power, and two of them (T104 and T115) are marked as anomalous.

### 4.3. “Faulted” Data Set

The SCADA data set includes a tag called “faulted,” which is a binary parameter recorded for each wind turbine in February, 2010. The data is not available for the earlier

periods. When this information is available, it is the closest we have to the ground truth about the state of health of each wind turbine.

One or more “faulted” conditions are recorded in the SCADA data for four of the seven wind turbines (T075, T081, T104, and T127) between Feb 13 and Feb 22, 2010. These faulted cases are indicated by the vertical red lines drawn on the subplots for the wind turbines in Figure 15. As before, the red circles indicate the anomaly isolated to a wind turbine by the associative model.

Notice that for T075 and T104, the red circles precede the red lines. The associative model is not only able to detect the developing anomaly, but it can do so up to two days in advance of the detection logic built into the SCADA system.

T081 seems to have an intermittent failure, as indicated by multiple vertical lines spanning the six days between February 12 and 18. The associative model has two clusters of indications for this wind turbine. The first one is produced on February 15—about three days after the first faulted signal. The second one is produced on February 17—about one day before the last faulted signal. In other words, the associative model shows sensitivity to the intermitted failure and is able to detect it.

T127 contains one faulted flag during this period. This case is not detected by the associative model. It is interesting to note that this fault occurs at exactly the same time as the last fault on T081. Further information about the nature of this fault could help explain why it is missed by the associative model.

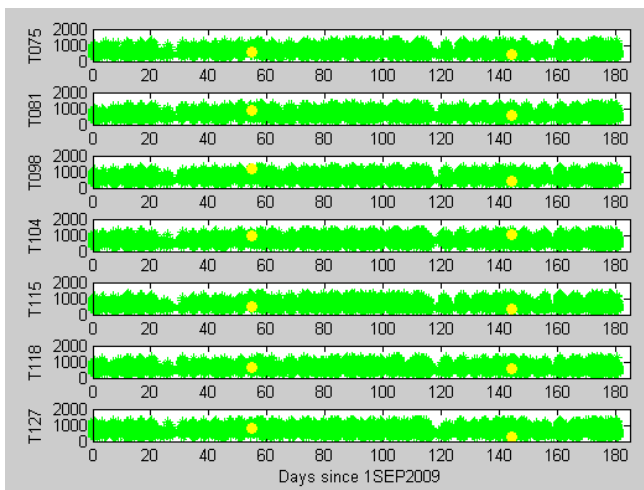


Figure 11. Associative model result for no fault data spanning 6 months

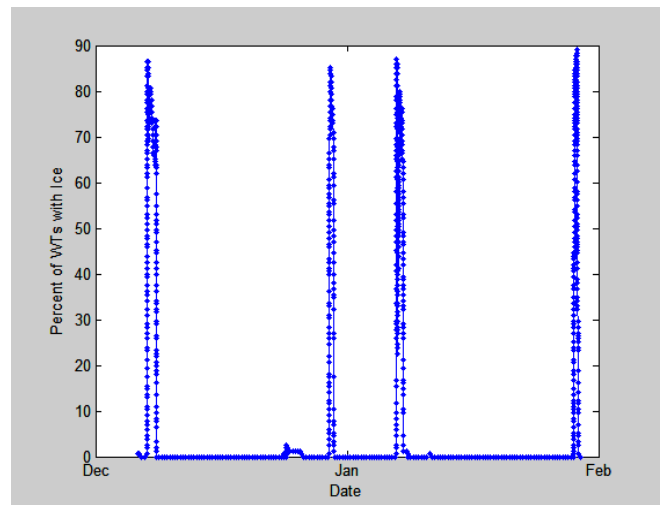


Figure 12. Ice condition times in Park II

Finally, the associative model indicates a fault on T115 on February 20 at about the same time as it indicates a fault on T104. In this case, the anomaly is correctly detected; however, the associative model could only narrow it down to T104 or T115, instead of firmly isolating it to T104.

Overall, the results in this test case show that 3 out of 4 anomalies were detected correctly; 2 out of 4 cases were detected in advance; 1 case was missed; and in one case the ambiguity set could only be reduced to 2 wind turbines.

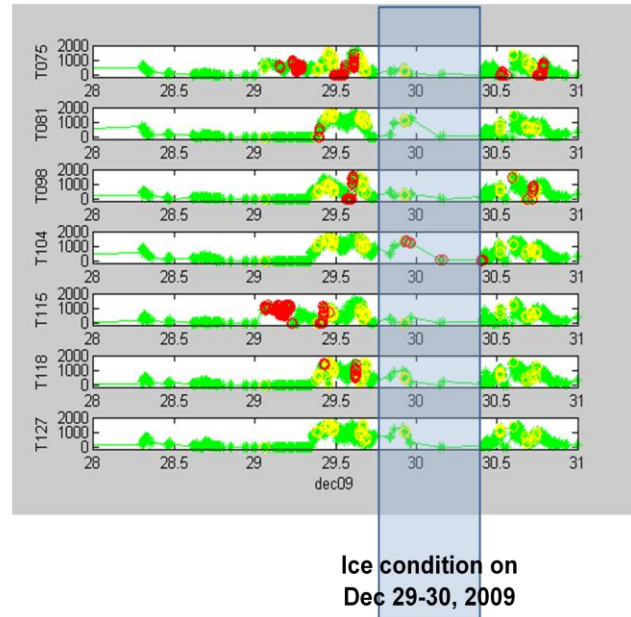
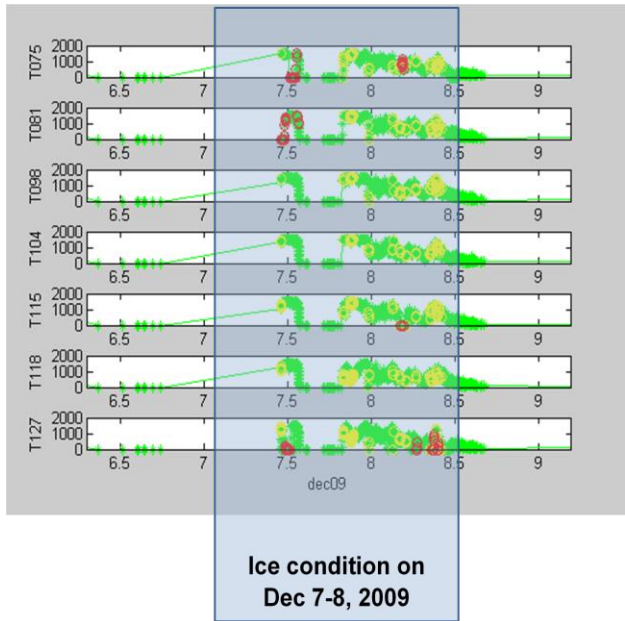


Figure 13. Associative model results for ice condition data in Dec 2009

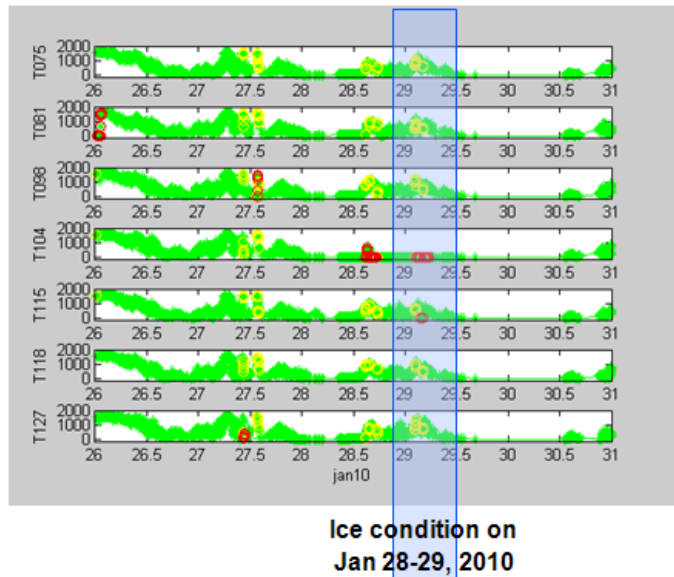
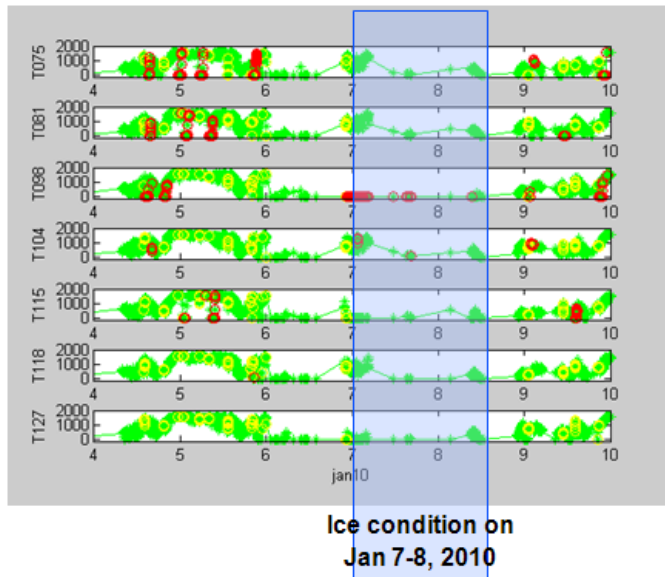


Figure 14. Associative model results for ice condition data in Jan 2010



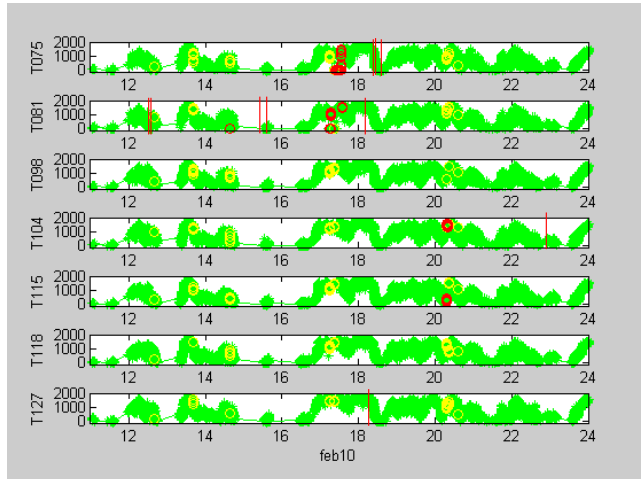


Figure 15. Associative model output for the “faulted” data

## 5. CONCLUSION

We analyzed data from several wind turbines on a wind farm and developed an automated fault detection approach using an associative model. The data analysis and visualization of the statistical features showed excellent cumulative correlation over large periods of time (several days) among different wind turbines. The associative model used correlations in a set of wind turbines on a much finer time scale, on the order of minutes, to detect anomalies. Several data filters were created to systematically segregate data for training and testing under different fault or no-fault conditions. The results show that the associative model is a promising approach for capturing correlated wind turbine behavior operating under similar conditions and for automated detection of anomalies when the correlation is broken.

Although we don't have access to the root causes behind anomaly indications, a rich set of configurable data filters provided us with testing data. Testing with the clean data showed no false positives. Testing with the data containing the 'Faulted' tag or large ice condition indicator (indicating possibility of wide-spread icing) showed that the AM is able to detect these conditions—sometimes in advance of the alarms. Further tuning with known fault data is needed to mature the approach.

## ACKNOWLEDGMENT

We thank the U.S. Department of Energy for the support for this work under Award Number DE-EE0001368.

## REFERENCES

Bell, M. B. & Foslien, W. K. (2005). Early event detection—results from a prototype implementation. In 17th Annual Ethylene Producers' Conference, Session

TA006-Ethylene Plant Process Control. Spring National Meeting (pp. 727-741), Apr. 10-14, Atlanta, GA.

- Gorinevsky, D., Dittmar, K., Mylaraswamy, D. & Nwadiogbu, E. (2002). Model-based diagnostics for an aircraft auxiliary power unit. IEEE Conference on Control Applications. (pp 215-220), Sept 18–20, Glasgow, Scotland.
- Kim, K. & Mylaraswamy, D. (2006). Fault Diagnosis and Prognosis of Gas Turbine Engines Based on Qualitative Modeling. In *ASME TurboExpo*. (881–889), May 8–11, Barcelona, Spain.
- Kim, K., Parthasarathy, G., Uluyol, O., Foslien, W., Sheng, S., Fleming, P., (2011), Use of SCADA Data for Failure Detection in Wind Turbines, Proceedings of 2011 Energy Sustainability Conference and Fuel Cell Conference, August 7-10, 2011, Washington DC, USA.
- Kramer, M. A., (1992) Autoassociative neural networks, *Computers Chem. Eng.*, Vol. 16, No. 4, 313-328.
- Kusiak, A., & Li, W. (2011). The prediction and diagnosis of wind turbine faults. *Renewable Energy*, 36, pp 16-23.
- Uluyol, O., Buczak, A.L., Nwadiogbu, E., (2001), Neural Networks Based Sensor Validation and Recovery Methodology for Advanced Aircraft Engines, *Proceedings of AeroSense 2001*, SPIE Vol. 4389, pp. 102-109, Orlando, Florida..
- Uluyol, O., Kim, K., Wrest, D., Nwadiogbu, E., (2003), Sensor Validation and Recovery through Iterative Association Models, *Proceedings of the IEEE Sensors 2003*, Oct. 22-24.
- Uluyol, O., Parthasarathy, G., Foslien, W., Kim, K., (2011), Power curve analytic for wind turbine performance monitoring and prognostics, Annual Conference of the Prognostics and Health Management Society, Montreal, Canada.
- Ye, X., Yan, Y. and Osadciw, L., (2010). Learning Decision Rules by Particle Swarm Optimization (PSO) for Wind Turbine Fault Diagnosis, Annual Conference of the Prognostics and Health Management Society, Oct 10-16, 2010, Portland, OR.
- Zaher, A., McArthur, S.D.J. & Infield, D.G., (2009), Online wind turbine fault detection through automated SCADA data analysis, wind energy. Published online in *Wiley Interscience* <http://www.interscience.wiley.com> DOI: 10.1002/we.319.

## AUTHORS



Dr. Onder Uluyol is a senior scientist at Honeywell Aerospace Condition-Based Maintenance (CBM) Center of Excellence. He received his Ph.D. in Computational Science and Engineering from the University of Illinois, Urbana-Champaign, IL. Dr. Uluyol has extensive experience in system health management and published numerous

technical articles on innovative modeling, measurement, diagnostics, and control approaches. Most recently, he has been focusing on wind and gas turbine performance monitoring using associative models and bearing diagnostics using oil debris and vibration monitoring. Dr. Uluyol holds seven patents on sensor validation, gas turbine light-off detection and start-up monitoring, substance detection using an electronic nose, and on various fault diagnostic and isolation technologies. He has eight additional patent disclosures filed and pending.



Dr. Girija Parthasarathy is a Senior Principal Research Scientist at Honeywell Automation and

Control Solutions Labs. She received her Ph. D. in Mechanical Engineering from the University of Minnesota, Minneapolis. She has broad experience with increasing leadership responsibility in research and development programs. She has served roles of principal investigator and program manager in several government- and Honeywell- funded programs. Most recently, she served as PI for a DOE-funded program on Condition Based Maintenance for Wind Farms. Her technical experience spans wind turbine health monitoring, turbine engine diagnostics and prognostics, microgrid modeling and optimization, combined heat and power, combustion and renewable energy. Dr. Parthasarathy has authored several conference and journal papers and holds several patents.

All trademarks referenced in this paper are the property of their respective owners.

# An Evidential Evolving Prognostic Approach and its Application to PRONOSTIA's Data Streams

Lisa Serir, Emmanuel Ramasso, and Nouredine Zerhouni  
FEMTO-ST Institute, UMR CNRS 6174 - UFC / ENSMM / UTBM,  
Automatic Control and Micro-Mechatronic Systems Dep., 25000, Besançon, France  
(lisa.serir, emmanuel.ramasso, noureddine.zerhouni)@femto-st.fr

## ABSTRACT

The research activity in the PHM community is in full bloom and many efforts are being made to develop more realistic and reliable methodologies. However, there still exist very few real-world applications due to the complexity of the systems of interest. Nonlinear dynamical systems identification and behavior prediction are difficult problems encountered in prognosis. The difficulty in switching from theory to practice can partially be explained by the existence of different kinds of uncertainty at each step of the implementation that must be taken into account with the appropriate tools. In this paper, we propose an evolving multi-modeling approach for the detection, the adaptation and the combination of local models in order to analyze complex systems behavior. It relies on belief functions in order to take into consideration the uncertainty related to the available data describing the system as well as the uncertainty generated by the nonlinearity of the system. The information of doubt explicitly represented in the belief functions framework is exploited to properly segment the data and take into account the uncertainty related to the transitions between the operating regions. The proposed algorithm is validated on a data provided by PRONOSTIA platform.

## 1. INTRODUCTION

Although prognosis is acknowledged as a key element in industrial maintenance strategies, there still exist very few real-world applications due to the complexity of the systems of interest. The implementation of a data-driven PHM tool in real conditions requires a learning phase for the estimation of the different parameters, that represents one of the main barriers to its applicability, as it is very costly to collect data, in particular for the failure modes. Modeling the behavior of nonlinear systems from observed data is a difficult task to perform widely encountered in prediction and fault diagnosis (Angelov, Filev, & Kasabov, 2010). One way to overcome

the complexity related to nonlinearity is to adopt multi-model approaches (Madani, Rybnik, & Chebira, 2003; Boukhris, Mourot, & Ragot, 2000; Murray-Smith, 1997; Nelles, 1995) by considering that the system's behavior gradually evolves along the operating range. Thus, the system could locally be described by simple functions corresponding to some operating regions. The description of the global system's behavior is then made by the combination of the local models, each of which being weighted by an *activation degree*. The identification consists in two main steps: the structural identification to determine the number of models, and a parametric identification to evaluate the parameters of the local models.

Understanding the response of nonlinear systems is a very challenging task leading to significant uncertainty. Uncertainty sources are numerous and may take the form of system variability, environmental and operational conditions, data acquisition errors, among other sources that vary depending on the application at hand. The quantity as well as the quality of the data are also very important aspects to take into account. Moreover, the segmentation of the data into meaningful modes is often required to help the methods of parameters' estimation to converge towards reasonable solutions. These constraints on the data are not clearly considered in the literature and we believe that the uncertainty related to the lack of data for the detection and prediction of failure modes must be quantified to understand the capabilities and limitations of the modeling process. While probability theory is well suited to deal with aleatory uncertainties (intrinsic variability), other formalisms exist that are more appropriate to manage epistemic (imprecision and incompleteness) uncertainty (G. J. Klir, 2006), among which, fuzzy sets or possibility theory and evidence theory, also known as belief functions theory (Shafer, 1976) which are the most prominent ones.

Fuzzy set theory has been used to deal with imprecision within data in multi-modeling approaches (Chandrashekar & Ganguli, 2009; Haag, Herrmann, & Hanss, 2010), among which the fuzzy rule-based models of Takagi-Sugeno (TSK) type (Takagi & Sugeno, 1985), widely used in modeling applications of complex systems, due to their flexibility and

First Author et.al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

computational efficiency. Their main advantage dwells in the fact that since the local regions are fuzzily defined, the resulting global model can be nonlinear, while the local models can be very simple (linear) (Takagi & Sugeno, 1985; Angelov, Lughofer, & Zhou, 2008). This kind of approach has been applied to build a neuro-fuzzy predictor in the context of prognosis application in (El-Koujok, Gouriveau, & Zerhouni, 2011). It was based on the evolving extended Takagi-Sugeno system (exTS) proposed by Angelov (Angelov & Filev, 2004).

Ramdani *et al.* (Ramdani, Mourot, & Ragot, 2005) exploited the theoretical framework of belief functions to deal with uncertainties in multi-modeling founded on a TSK fuzzy model. The main advantage of this approach remains in the use of belief functions theory to determine the activation degrees of the local models. However, Their proposed methodology is an *offline* approach and requires the entire dataset to be available in advance for the modeling process. Belief functions are particularly interesting because they obviate the need to introduce unjustified a priori thanks to the relaxation of the additivity constraint, which makes possible the explicit representation of doubt. The work of Smets (Smets & Kennes, 1994) has shown that the information brought by the empty set could enable to take into account "open world" situations. Although many publications using belief functions theory have recently increased, very few are related to PHM applications.

One of the constraints to address in PHM applications is related to the fact that the data concerning the failure modes are generally rare compared to the data of normal functioning modes. This can be seen as a problem of unbalanced data (He & Garci, 2009). A solution is to construct an evolving algorithm that can adapt to the lack of data. This kind of approach enables the consideration of applications for which historical data cannot be stored. Then, for more coherence and to facilitate the implementation of the prognostic approach, the detection and the prediction must be performed within the same algorithm.

In this paper, we propose to adapt the offline approach developed in (Ramdani *et al.*, 2005) to the online case, where the data arrive gradually. In the sequel, We qualify the proposed approach as "evolving", as it is able to adapt its parameters online. The proposed algorithm is called E2GKpro and relies on the Evidential Evolving Gustafson-Kessel algorithm (E2GK) initially developed in (Serir, Ramasso, & Zerhouni, 2012) to sequentially perform the clustering phase using the formalism of belief functions. The clustering is then followed by the online identification of local linear models. The activation degrees of each local model are directly provided by the E2GK algorithm in the form of belief masses, and the global model is a combination of all the local models. This work is presented as an improved version of a previous contribution (Serir, Ramasso, & Zerhouni, 2011).

The paper is organized as follows. First, the basics of belief functions theory are given and illustrated on a simple hypothetical example (Section 2). Then, the proposed E2GKpro algorithm is detailed (Section 3). And finally, the algorithm is applied on the data of PRONOSTIA platform (Section 4).

## 2. BACKGROUND MATERIALS ON BELIEF FUNCTIONS THEORY

The theory of belief functions, also called theory of evidence or Dempster-Shafer theory, is a formal framework for reasoning with partial (imprecise and uncertain) information. It was introduced by Dempster (1968) and Shafer (1976) and later developed by Smets and others. The theory of belief functions extends both the set-membership (intersection, union, inclusion, etc.) and probabilistic (conditioning, marginalization) approaches to uncertain reasoning, and a belief function may be viewed as both a generalized set and a non additive measure. Smets has developed the *Transferable Belief Model* (TBM) (Smets & Kennes, 1994) as a general framework for uncertainty representation and combination of various pieces of information without additional prior. In particular, TBM offers the possibility to explicitly emphasize doubt, representing ignorance, and conflict, emphasizing the contradiction in a fusion process. We give in this section some of the basic notions of the theory and refer the reader to (Smets & Kennes, 1994) for a more complete description.

Let  $\omega$  be a variable taking values in a finite set  $\Omega$ , called *frame of discernment*, and let  $2^\Omega$  be its power set. A belief function on  $\Omega$  is mathematically defined by introducing a set function  $m$ , called the basic belief assignment (BBA) defined by:

$$\begin{aligned} m : 2^\Omega &\rightarrow [0, 1] \\ A &\mapsto m(A) \end{aligned} \quad (1)$$

and satisfies:  $\sum_{A \subseteq \Omega} m(A) = 1$ . Each subset  $A \subseteq \Omega$  such as  $m(A) > 0$  is called a *focal elements* of  $m$ .

We collect a piece of evidence (information) about  $\omega$ , which can have different interpretations weighted by subjective probabilities. One of these interpretations can mean that we only know that  $\omega \in A$ , for some  $A \subseteq \Omega$ , and nothing more. The probability that the evidence means exactly that  $\omega \in A$  is  $m(A)$ . A BBA is assigned not only to singletons ( $|A| = 1$ ), but also to *subsets* of  $\Omega$  ( $|A| > 1$ ), *without assumption concerning additivity* of the measure  $m$  (G. J.Klir, 2006). This property permits the explicit modeling of doubt and conflict which constitutes a fundamental difference with probability theory:

- Perfect knowledge of the value of  $\omega \in \Omega$  is represented by the allocation of the unit of belief to  $\omega$ :  $m(\omega) = 1$ . In this case,  $m$  is said to be *certain*.
- Complete ignorance corresponds to  $m(\Omega) = 1$ .
- In the case of all focal elements being singletons,  $m$  boils down to a probability function and is said to be *Bayesian*.

- The value of  $m(\emptyset)$  is called *conflict* and can be positive, meaning that one accepts the *open-world assumption* stating that the set  $\Omega$  might not be complete, and thus  $\omega$  might take its value outside  $\Omega$ .
- In the case of all focal elements being singletons,  $m$  boils down to a probability function and is said to be *Bayesian*.

A BBA  $m$  is said to be *normal* if  $m(\emptyset) = 0$ . A normalized BBA  $m^*$  can be computed from a BBA  $m$  by applying the Dempster normalization:

$$m^*(A) = \begin{cases} \frac{m(A)}{1 - m(\emptyset)} & \text{if } A \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Given a BBA  $m$ , a belief function  $bel$  and a plausibility function  $pl$  can be defined, respectively, as

$$bel(A) = \sum_{\emptyset \neq B \subseteq A} m(B), \quad \forall A \subseteq \Omega, \quad (3)$$

$$pl(A) = \sum_{B \cap A \neq \emptyset} m(B), \quad \forall A \subseteq \Omega, \quad (4)$$

$bel(A)$  can be interpreted as a measure of our total belief committed to  $A$  after receiving the item of evidence, and  $pl(A)$  represents the maximal degree of belief supporting the subset  $A$  (Cobb & Shenoy, 2006). If  $m$  is Bayesian,  $pl = bel$  boils down to a probability measure. Note that a possibility measure is known to be formally equivalent to a *consonant* belief function, i.e., a belief function with nested focal elements (Dubois, Prade, & Smets, 2001). The functions  $bel$ ,  $pl$  and  $m$  are in one-to-one correspondence and represent three facets of the same piece of information.

The TBM distinguishes the credal level where beliefs are formalized, revised and combined, and the *pignistic* level used for decision making, which consists in the choice of the best hypothesis using the *pignistic probability* distribution (Smets & Kennes, 1994) defined as:

$$\mathbf{BetP}(\omega) = \sum_{A \ni \omega} \frac{m(A)}{|A|} \frac{1}{1 - m(\emptyset)}, \quad \forall \omega \in \Omega. \quad (5)$$

where each mass of belief  $m(A)$  is equally distributed among the elements of  $A$  and  $\mathbf{BetP}(A) = \sum_{\omega \in A} \mathbf{BetP}(\omega), \forall A \subseteq \Omega$ .

**Example 1** *Let us consider the example described in Fig. 1. Suppose that the figure results from a monitoring of a bearing system, and that it represents the evolution along time of a feature provided by a sensor (the available piece of evidence). There are 4 possible operating modes: Normal ( $N$ ), Degrading 1 ( $D1$ ), Degrading 2 ( $D2$ ) and Faulty ( $F$ ) ( $\Omega = \{N, D1, D2, F\}$ ). One can notice that the feature evolution curve can be segmented into 4 clear operating regions (arrows in magenta color). However, in the transitions (magenta ellipses) from one operating region to another ( $N \rightarrow D1, D1 \rightarrow D2, D2 \rightarrow F$ ), one can express doubt*

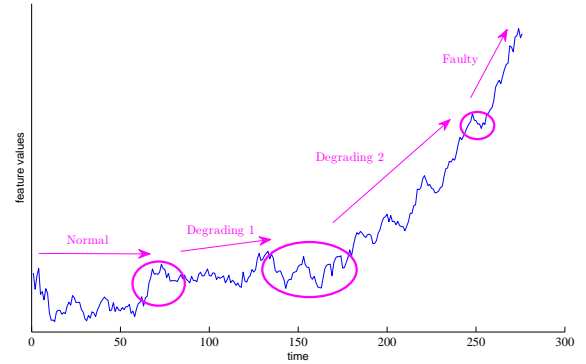


Figure 1. An example of basic belief assignments.

regarding whether the bearing is still in the same operating mode or if it had evolved to a new one. One's belief regarding the operating mode of the bearing along time can "consecutively" be expressed by:  $m(N) = 1, m(\{N, D1\}), m(D1) = 1, m(\{D1, D2\}), m(D2) = 1, m(\{D2, F\})$  and  $m(F) = 1$ .

### 3. AN EVIDENTIAL EVOLVING PREDICTION METHODOLOGY

The proposed methodology, called E2GKpro, follows a three-phase scheme, commonly used in the evolving systems modeling approaches (Angelov et al., 2010). The novelty of E2GKpro dwells in the use of belief functions for the determination of the local models and the estimation of their activation degrees.

The system under study is supposed to gradually evolve through different operating modes (Fig. 1), each of which corresponding to unknown local linear models to be identified online. At the current time-instant  $k$ , a  $n$ -dimensional input feature vector  $x_k = [x_1 \dots x_n]^T \in \mathbb{R}^n$  and an output  $y_k \in \mathbb{R}$  are observed. After an initialization phase, given  $(x_k, y_k)$ , E2GKpro starts with a clustering phase to detect the current operating regions of the system. Then, a regression phase is performed to update the local linear models corresponding to each cluster. Finally, the prediction phase estimates the value of the output  $\hat{y}_k$  given the input  $x_k$ .

#### 3.1. Initialization

To explain the belief functions generation process, let consider the existence of an initial set of available data so that two operating regions can be identified by applying a standard clustering algorithm on the input-output space. Given an initial observed datum  $z_k = (x_k, y_k)$ , the Gustafson-Kessel algorithm is used, with  $c = 2$ , to find the coordinates  $v_{i,k}, i = 1, 2$  of the two first clusters and to initialize their covariance matrices  $\Sigma_{i,k}, i = 1, 2$ .

### 3.1.1. Computing the evidential partition

Let  $\Omega$  be the set of the so far existing clusters. Given the centers and the covariance matrices, the degree of belief regarding the membership of the  $k$ -th data point to each possible subset of clusters  $A_i \in 2^\Omega \setminus \emptyset$ , can be computed as in the evidential evolving clustering algorithm (E2GK) (Serir et al., 2012):

$$m_{ik} = \frac{|A_i|^{-1} \cdot d_{ik}^{-2}}{\sum_{A_l \neq \emptyset} |A_l|^{-1} \cdot d_{lk}^{-2} + \delta^{-2}}, \quad (6)$$

and the mass assigned to the emptyset is equal to:

$$m_{\emptyset k} = 1 - \sum_{A_i \neq \emptyset} m_{ik}, \quad (7)$$

where  $d_{ik}$  denotes the Mahalanobis-like distance between data point  $z_k$  and subset  $A_i$ , and  $\delta \in \mathbb{R}^+$  controls the amount of data considered as outliers. the distance  $d_{ik}$  is computed by first defining the center of  $A_i$  as the barycenter  $\bar{v}_{i,k}$  of clusters' centers composing  $A_i$ . The corresponding covariance matrix  $\Sigma_{i,k}$  can then be computed as:

$$\Sigma_{i,k} = \frac{1}{N-1} \cdot \sum_{k=1}^N \sum_{A_j \ni \omega_i} (z_k - \bar{v}_{i,k}) \cdot (z_k - \bar{v}_{i,k})^T, \quad (8)$$

and the distance  $d_{i,k}$  is given by:

$$d_{ik}^2 = \|z_k - \bar{v}_{i,k}\|_{S_i}^2 = (z_k - \bar{v}_{i,k}) \cdot S_i \cdot (z_k - \bar{v}_{i,k})^T, \quad (9a)$$

$$S_{i,k} = [\det(\Sigma_{i,k})]^{1/n} \cdot \Sigma_{i,k}^{-1}, \quad (9b)$$

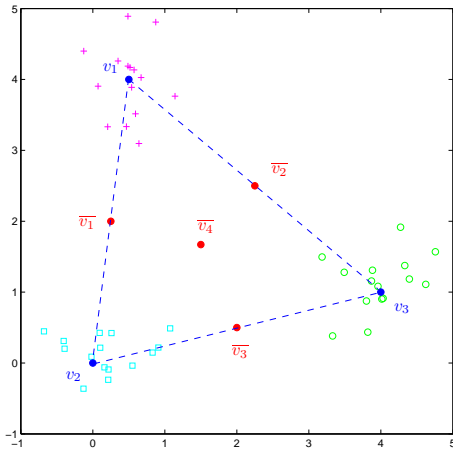


Figure 2. Centers of subsets of clusters in  $\Omega$ .

**Remark 1** Only singleton focal elements (clusters  $\omega_k \in \Omega$ ) are associated with centroids but the particularity of the BBA computation holds in the consideration of virtual centroids

located at the barycenters of subsets of clusters (Masson & Denoeux, 2008). This makes the proposed approach different from usual possibilistic and probabilistic clustering approaches.

### 3.1.2. Initializing the local models

Let  $X_i$  be the set of input points belonging to the  $i$ -th cluster,  $Y_i$  the corresponding outputs, and  $\theta_i$  the parameters of the local linear model for the  $i$ -th cluster that can be optimized by a standard least squared approach:

$$\theta_i = (X_i^T \cdot X_i)^{-1} \cdot X_i^T \cdot Y_i \quad (10)$$

After completing the initialization, E2GKpro can be run online: if the observed data is an input-output couple  $z_k$  then the clustering and the local models can be updated, whereas if only the input data  $x_k$  is observed, a prediction  $\hat{y}_k$  of the output can be estimated.

### 3.2. Online clustering phase

This phase aims at detecting the current operating regions of the system. The number of clusters  $c$  can evolve, in particular when a new operating region is detected. At the current instant  $k$ , each cluster  $i = 1 \dots c$  is identified by two parameters: a center  $v_{i,k} \in \mathbb{R}^n$  and a covariance matrix  $\Sigma_{i,k} \in \mathbb{R}^n \times \mathbb{R}^n$ , both adapted according to  $(x_k, y_k)$ . This phase relies on the evidential evolving clustering algorithm called E2GK (Serir et al., 2012).

When a new input-output datum  $z_k$  is observed, the data structure can be updated. The boundary of each cluster  $i$  is first estimated by computing its radius: The radius  $r_i$  of the  $i$ -th cluster is computed by:

$$r_i = \underset{\forall z_k \in i\text{-th cluster and } \lambda_{ik} > c^{-1}}{\text{median}} \|z_k - v_{i,k}\|_{S_i}, \quad (11)$$

where  $\lambda_{ik}$  is the confidence degree that point  $x_k$  belongs to a singleton cluster ( $\omega_i \in \Omega$ ) estimated by the pignistic transformation (Eq. 5) (Serir et al., 2012):

$$\lambda_{i,k} = \sum_{A_j, \omega_i \in A_j} \frac{m_{j,k}}{|A_j|} \quad (12)$$

The closest cluster to  $z_k$  is then found by:

$$p = \underset{i=1}{\text{argmin}}^c d_{ik}^2 \quad (13)$$

Once cluster  $p$  is found, two cases are possible:

- Case 1:  $d_{pk}^2 \leq r_p$ , i.e.  $z_k$  belongs to an existing cluster, inducing an update of the  $p$ -th cluster:

$$v_{p,k+1} = v_{p,k} + \theta \cdot \Delta, \quad (14)$$

where  $\Delta = z_k - v_{p,k}$  and  $\theta \in [0, 1]$  is the updating rate. The inverse of the covariance matrix and its determinant can be recursively adapted by (Georgieva & Filev, 2009):

$$\Sigma_{p,k+1}^{-1} = \left( I - \frac{\theta \cdot \Sigma_{p,k}^{-1} \cdot \Delta^T \cdot \Delta}{\xi} \right) \cdot \Sigma_{p,k}^{-1} \cdot \frac{1}{1 - \theta}, \quad (15)$$

where  $I$  is the identity matrix and

$$\xi = 1 - \theta + \theta \cdot \Delta \cdot \Sigma_{p,k}^{-1} \cdot \Delta^T \quad (16)$$

and

$$\det(\Sigma_{p,k+1}) = (1 - \theta)^{n-1} \cdot \det(\Sigma_{p,k}) \cdot \xi. \quad (17)$$

The partition matrix can then be computed (Eq. 9b, 6 and 7).

- Case 2:  $d_{pk}^2 > r_p$ , i.e.  $z_k$  is too far from any existing cluster, involving the creation and the validation of a new cluster: the number of clusters is incremented and the incoming data  $z_k$  is chosen as the center of the new cluster, whose parameters are initialized as those of the closest cluster  $p$ :

$$c \leftarrow c + 1, \quad (18a)$$

$$v_{c,k+1} = z_k, \quad (18b)$$

$$\Sigma_{c,k+1} = \Sigma_{p,k}. \quad (18c)$$

In (Serir et al., 2012), a constraint of validity of the partition was added by imposing that each cluster in the new partition has a minimum of points denoted  $P_{\text{tol}}$  and is simply removed otherwise. Doing so, the number of clusters can increase or decrease explains a better modeling of the data structure (Serir et al., 2012). This flexibility guarantees the validity of both the covariance matrices and the local models.

### 3.3. Online Regression phase

In this phase, the local linear models corresponding to each cluster are updated. A local model relates  $x_k$  to  $y_k$  such that, given only the input  $x_k$ , the predicted output  $\hat{y}_k$  is as close as possible to the true value  $y_k$ . Therefore, in this phase, the goal is to optimize the vector of parameters of the  $i$ -th local model.

The estimates of the parameters of the local linear models at a given instant  $k$ ,  $\pi_{i,k} = [a_{i0,k} \ a_{i1,k} \ a_{i2,k} \ \dots \ a_{in,k}]^T$  are computed by a recursive least squared approach, given their previous estimates  $\pi_{i,k-1}$  and the new input-output vector  $z_k$ :

$$\mathcal{C}_{i,k} = \mathcal{C}_{i,k-1} - \frac{\lambda_i(x_{k-1}) \cdot \mathcal{C}_{i,k-1} \cdot x_{e,k-1}^T \cdot x_{e,k-1} \cdot \mathcal{C}_{i,k-1}}{1 + \lambda_i(x_{k-1}) \cdot x_{e,k-1}^T \cdot \mathcal{C}_{i,k-1} \cdot x_{e,k-1}} \quad (19a)$$

$$\hat{\pi}_{i,k} = \hat{\pi}_{i,k-1} + \mathcal{C}_{i,k} \cdot x_{e,k-1} \cdot \lambda_i(x_{k-1}) \cdot (y_k - x_{e,k-1}^T \cdot \hat{\pi}_{i,k-1}) \quad (19b)$$

with  $k > 2$  (time-step),  $i = [1, c]$  ( $c$  is the number of clusters = the number of local models),  $x_{e,k} = [1; x_k]$  is the extended input data vector, and  $\mathcal{C}_{i,k} \in \mathbb{R}^{(n+1) \times (n+1)}$  is the covariance matrix of the  $i$ -th local model at time  $k$ , which is updated by the Ricatti equation as in the Kalman filter. The initial conditions ( $k = 1$ ) are set to  $\hat{\pi}_1 = 0$  and  $c_{i,1} = \alpha \cdot I$ , where  $I$  is the identity matrix and  $\alpha$  a large value (for example 100).

### 3.4. Prediction phase

Given an input datum  $x_k$ , the goal of the prediction phase is to which estimate the value  $\hat{y}_k$  taken by the output. The prediction is computed by a simple weighted sum of the local linear models, where the weights are provided directly by the clustering phase:

$$\hat{y}_k = \Psi^T \cdot \theta \quad (20)$$

where  $\Psi = [\lambda_{1,k} \cdot x_{ek}; \dots; \lambda_{i,k} \cdot x_{ek}; \dots; \lambda_{i,c} \cdot x_{ek}]$  is the vector of the inputs weighted by the normalized activation degrees of the local models, and  $\theta = [\pi_{1,k}; \dots; \pi_{i,k}; \pi_{c,k}]$  is the vector of the parameters of the linear local models.

**Remark 2** *When the observed datum at instant  $k$  is an input-output couple  $z_k$ , then the clustering and the local models can be updated, whereas the prediction of the output  $\hat{y}_k$  can be estimated when only the input datum  $x_k$  is observed.*

## 4. APPLICATION ON A REAL-WORLD DATASET

To demonstrate the performance of the proposed method, a real-world multi-dimensional case is considered. E2GKpro is implemented on a dataset provided by the PRONOSTIA platform described in the sequel.

### 4.1. Description of PRONOSTIA

PRONOSTIA is an experimentation platform (Figure 3) dedicated to the test and validation of the machinery prognosis approaches, focusing on bearing prognostics. It was developed in the Department of Automatic Control and Micro-Mechatronic Systems (AS2M) of FEMTO-ST<sup>1</sup> institute. The main objective of PRONOSTIA is to provide real experimental data (Serir, Ramasso, Nectoux, Bauer, & Zerhouni, 2011) that characterize the degradation of a ball bearing along its whole operational life (until fault/failure). The collected data are vibration and temperature measurements of the rolling bearing during its functioning mode.

The internal bearing ring is put in rotation, while the external bearing ring is maintained fixed. A radial load is applied on the external bearing ring in order to simulate its functioning. To speed up the degradation, the load exceeds the maximal load recommended by the supplier. The originality of this experimental platform lies not only in the conjunction of the characterization of both the bearing functioning (speed, torque and radial force) and its degradation (vibrations and temperature), but also in the possibilities, offered by the platform, to make the operating conditions of the bearing vary during its useful life. Figure 3(c) depicts a bearing before and after the experiment.

The bearing operating conditions are determined by instantaneous measures of the radial force applied on the bearing,

<sup>1</sup>Franche-Comte, Electronics, Mechanics, Thermal Processing, Optics - Science and Technology

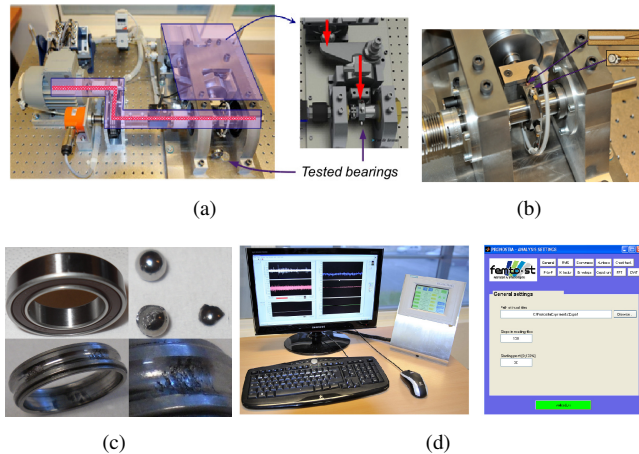


Figure 3. 3(a) The PRONOSTIA platform, 3(b) close-up view on sensors for degradation measurement, 3(c) example of degraded bearings and 3(d) the software for degradation analysis.

the rotation speed of the shaft handling the bearing, and of the torque inflicted on the bearing. During a test, the rolling bearing starts from its nominal mode until the fault state. The bearing behavior is measured using different types of sensors (Figure 3(b)) such as miniaturized acceleration sensors and temperature probe.

The raw signals provided by the sensors are processed in order to extract relevant information concerning bearings states. Several techniques have been implemented and gathered in a signal processing toolbox with Matlab (Fig. 3(d)): time-domain methods (RMS, skewness and kurtosis, crest factor, K-factor, Peak-to-Peak), frequency-domain methods (spectral and cepstrum analysis, envelope detection), time-frequency domain (short-time Fourier transform) and wavelets (discrete transform).

#### 4.2. Prognostic on PRONOSTIA

The experiments are conducted on data used for the IEEE PHM 2012 Prognostic Data Challenge provided by PRONOSTIA and available on its dedicated website<sup>2</sup>. More precisely, the data related to the first functioning condition are used. From the horizontal accelerometer data, two features are generated and used for testing (Fig. 4):

- The RMS which is computed at a given time  $k$  in a window of size 50. Let  $S$  be the signal of the accelerometer and  $\mu(S_W)$  its average value, then:

$$RMS(k) = \sqrt{\frac{1}{W} \sum_{i=k-W+1}^k (S(i) - \mu(S_W))^2} \quad (21)$$

<sup>2</sup><http://www.femto-st.fr/ieee-PHM2012-data-challenge>.

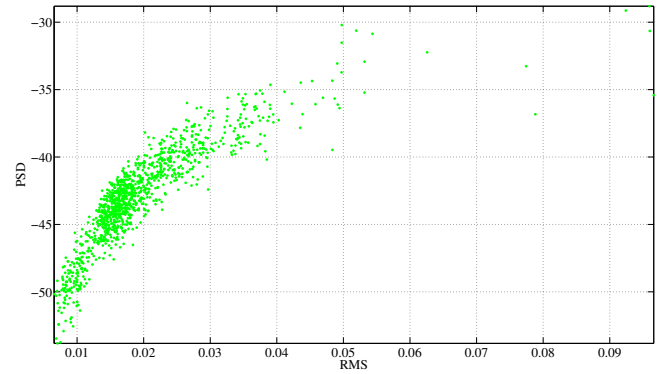


Figure 4. A set of data obtained from the PRONOSTIA platform.

- The mean value of the power spectral density in the same window as the RMS, denoted  $PSD(k)$ .

The data depict a typical behavior of wear with a small amount of data with high variance close to the end of the experiment (top, right-hand side), while the normal behavior is represented by a larger amount of data points gathered in a smaller space (bottom, left-hand side). The variance is partly due to the high level of noise generated by the vibrations during the experiment.

Let  $x(k) = [RMS(k) \quad PSD(k)]^T$  the feature vector at instant  $k$ . The goal is to predict  $x_{RMS}(k+100)$  and  $x_{PSD}(k+100)$  given  $x_{in} = [x(k-18) \quad x(k-12) \quad x(k-6) \quad x(k)]^T$ . For each data point,  $x_{in}$  is a vector with  $4 \times 2$  elements and the output vector is  $x_{out} = [x_{RMS}(k+100) \quad x_{PSD}(k+100)]^T$ . The clustering is performed in the input-output space composed of 10 elements. Given an input data vector (with 8 elements), the prediction is estimated by projecting the centers and the covariance matrix onto the input space, then computing the degree of membership to each cluster, followed by the regression.

The engine is launched and the bearing is gradually degraded. The data are processed along time by E2GKpro, so that clusters representing operating regions are created and the local models estimated. Only one experiment is used to tune the parameters of E2GKpro. The data related to the training dataset is given in Figure 4. The local models are updated as displayed in Figure 5 for the four first rules. The local model being initialized, the predictions 100 steps ahead can be estimated. The result of the prediction on the training data is depicted on Figure 6 (NDEI = 0.4986).

The second experiment is then used as a testing dataset. An interesting characteristic of the PRONOSTIA's data holds in the fact that all experiments depict different degradation trends, although bearings with similar mechanical properties were used. An illustration of this variability is represented in Figure 7. In this kind of application, the use of online



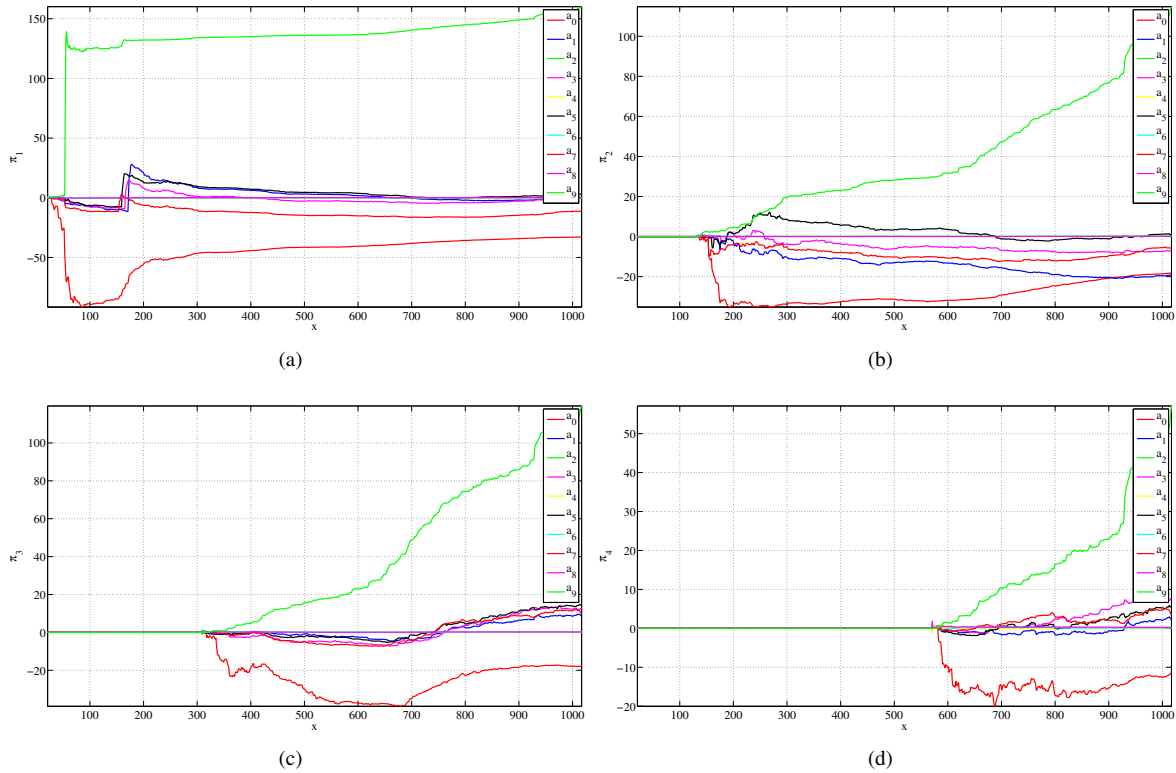


Figure 5. Evolution of the parameters for rules 1 to 4.

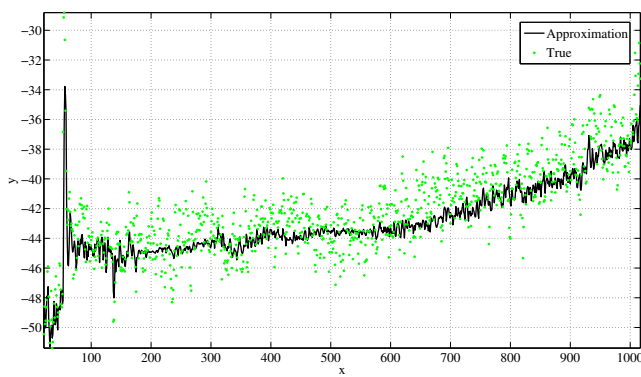


Figure 6. Real data and prediction on the training dataset.

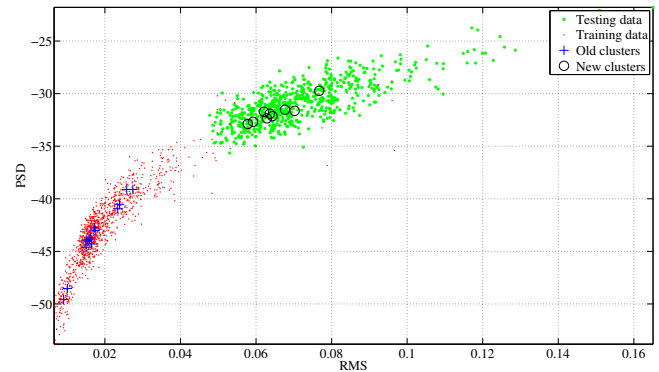


Figure 7. Illustration of high variability: training and testing data.

methods is thus well justified. This figure also shows the position of the clusters found during both training and testing. In the latter case, E2GKpro starts with the local models estimated during the former but E2GKpro still adapts the models accounting for new clusters in the new operating regions encountered in the testing dataset.

The result of the prediction on the second experiment (with features depicted in Figure 4 and on the left-hand side of Figure 7) is shown in Figure 6. In this example, E2GKpro generates smooth predictions, which appear useful in this real-

world application because it gives the global trend of the functioning behavior. Another interesting aspect is the updated set of operating regions found during testing (Figure 7), which can then be used for another experiment.

## 5. CONCLUSION

An evidential approach to analyze complex systems behavior was proposed in the context of sequential data. The proposed approach is in line with evolving systems modeling approaches and consists in three main phases performed on-line: 1) the on-line clustering of the data describing the system to determine the different operating regions, 2) the creation, adaptation or removing of models locally estimated for each cluster, and 3) the prediction of the future evolution. E2GKpro was implemented on a real-world dataset provided by PRONOSTIA platform. The results demonstrated the ability of the proposed method for an online segmentation of multi-dimensional time-series and reliable predictions. Further comparative studies have shown the great interest of using belief functions. In particular, the number of rules is decreased compared to usual approaches, while ensuring limited error by using the concept of virtual centroids to represent transitions between operating regions.

## REFERENCES

- Angelov, P., & Filev, D. (2004). An approach to online identification of Takagi-Sugeno fuzzy models. *IEEE Trans. on Systems, Man, and Cybernetics*, 34, 484 – 98.
- Angelov, P., Filev, D., & Kasabov, N. (2010). *Evolving intelligent systems: Methodology and applications*. New York: IEEE Press Series on Computational Intelligence, John Wiley.
- Angelov, P., Lughofer, E., & Zhou, X. (2008). Evolving fuzzy classifiers using different model architectures. *Fuzzy Sets and Systems*, 3160 – 3182.
- Boukhris, A., Mourot, G., & Ragot, J. (2000). Nonlinear dynamic system identification: a multiple-model approach. *International Journal of Control*, 72, 591 – 604.
- Chandrashekar, M., & Ganguli, R. (2009). Uncertainty handling in structural damage detection using fuzzy logic and probabilistic simulation. *Mechanical Systems and Signal Processing*, 23(2), 384 – 404.
- Cobb, B. R., & Shenoy, P. P. (2006). On the plausibility transformation method for translating belief function models to probability models. *International Journal of Approximate Reasoning*, 41, 314 – 330.
- Dubois, D., Prade, H., & Smets, P. (2001). New Semantics for Quantitative Possibility Theory. In *2nd International Symposium on Imprecise Probabilities and Their Applications*. Ithaca, New York.
- El-Koujok, M., Gouriveau, R., & Zerhouni, N. (2011). Reducing arbitrary choices in model building for prognostics: An approach by applying parsimony principle on an evolving neuro-fuzzy system. *Microelectronics Reliability*, 51, 310 – 330.
- G. J.Klir. (2006). *Uncertainty and information. Foundations of generalized information theory*. New York: Wiley.
- Georgieva, O., & Filev, D. (2009). Gustafson-Kessel algorithm for evolving data stream clustering. In *International Conference on Computer Systems and Technologies - CompSysTech 09*.
- Haag, T., Herrmann, J., & Hanss, M. (2010). Identification procedure for epistemic uncertainties using inverse fuzzy arithmetic. *Mechanical Systems and Signal Processing*, 24(7), 2021 – 2034.
- He, H., & Garci, E. A. (2009). Learning from Imbalanced Data. *IEEE Tr. On Knowledge and Data Engineering*, 21, 1263 – 1284.
- Madani, K., Rybnik, M., & Chebira, A. (2003). Non Linear Process Identification Using a Neural Network Based Multiple Models Generator. *LNCS series*, 647 – 654.
- Masson, M.-H., & Denoeux, T. (2008). ECM: An evidential version of the fuzzy c-means algorithm. *Pattern Recognition*, 41(4), 1384 – 1397.
- Murray-Smith, R. (1997). *Multiple model approaches to modeling and control*. Taylor & Francis Publishers.
- Nelles, O. (1995). On the identification with neural networks as series-parallel and parallel models. In *International Conference on ANN*. Paris, France.
- Ramdani, M., Mourot, G., & Ragot, J. (2005). A Multi-Modeling Strategy based on Belief Function Theory. In *CDC-ECC 05*.
- Serir, L., Ramasso, E., Nectoux, P., Bauer, O., & Zerhouni, N. (2011). Evidential Evolving Gustafson-Kessel Algorithm (E2GK) and Its Application to PRONOSTIA's Data Streams Partitioning. In *IEEE Int. Conf. on Decision and Control* (p. 8273-8278).
- Serir, L., Ramasso, E., & Zerhouni, N. (2011). E2GKpro: An evidential evolving multimodeling approach for systems behavior prediction. In *PHM Society*. Montreal, Canada.
- Serir, L., Ramasso, E., & Zerhouni, N. (2012, July). Evi-

dential evolving Gustafson-Kessel algorithm for online data streams partitioning using belief function theory. *International Journal of Approximate Reasoning*, 53, 747 – 768.

Shafer, G. (1976). *A mathematical theory of evidence*. Princeton University Press.

Smets, P., & Kennes, R. (1994). The Transferable Belief

Model. *Artificial Intelligence*, 66, 191 – 234.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its application to modeling and control. *IEEE Trans. on Systems, Man, and Cybernetics*, 15, 116 – 132.

# Time Distribution Mapping: a Generic Transient Signal Monitoring Technique for Prognostic Methods

Michael Sharp<sup>1</sup>, J Wesley Hines<sup>2</sup>

<sup>1,2</sup>*University of Tennessee Nuclear Engineering Department, Knoxville, Tennessee, 37996-2300, USA*

*msharp6@utk.edu*

*jhines2@utk.edu*

## ABSTRACT

The utilization of steady state monitoring techniques has become an established means of providing diagnostic and prognostic information for systems and equipment. This is mainly driven by both the wealth of available analysis techniques and the comparatively larger amount of data. However, steady state data is not the only, or in some cases, even the best source of information regarding the health and state of a system. Transient data has largely been overlooked as a source of system information due to the additional complexity in analyzing these types of signals. Time Distribution Mapping via the Sharp Transform allows for a fast, intuitive, generic quantification of deviations a transient signal from an established norm. Without regard to the type or source of the signal, referencing to an established Time Distribution Map can implicitly capture shifts mean, standard deviation, skewness, or even gross frequency shifts without need of additional processing. By quantifying and trending these shifts, an accurate measure of system health can be established and utilized by prognostic algorithms. In fact, for some systems the elevated stress levels during transients can provide better, more clear indications of system health than those derived from steady state monitoring.

## 1. INTRODUCTION

Signal monitoring has advanced to the point where viable and accurate information regarding the health and state of a system can quickly and effectively be obtained using purely data driven techniques. Traditionally these techniques focus on steady state signals for both their availability and regularity. Transient signals however, provide a unique challenge and opportunity in regards to health information extraction. While the nature of these signals violates many of the assumptions necessary for steady state analysis, the

elevated stress levels seen in these signals can provide faster, more accurate indications of health in many systems.

By definition, a transient signal undergoes some significant statistical shift with regards to the pertinent time frame being analyzed. Specifically, this paper refers to these types of transient signals generated from systems and equipment undergoing a normal operational transient such as startup, shut down, or a load shift. One main difficulty in analyzing these signals is the lack of means of generically and intuitively quantifying the degradation of a transient signal. Without these, each signal analysis necessitates a case-specific algorithm for processing whose development generally relies on some engineering knowledge of the signal itself and its' attributes.

Time distribution mapping provides a generic, data driven means of quickly, intuitively, and accurately quantifying the degradation of a transient signal signature through the lifetime of a system. By using historical transient signatures to establish a nominal time distribution map, and transforming signatures into the same bin space as described in following sections, a residual can be created between these maps to provide measure of signature deviation which directly relates back to system health. This additional measure of system health can be used either alone, or to augment existing information from steady state monitoring, providing more robust and accurate information regarding the overall state of the system. In fact, some systems, such as backup generators, may only have transient information on record as they are subjected to periodic pass/fail startup tests. The ability to utilize this information to better assess the current state of the unit, and predict future states of the unit could be invaluable, possibly leading to less frequent and more accurate decision-making regarding the unit.

## 2. BRIEF REVIEW OF GENERAL TRANSIENT ANALYSIS METHODS

Transient signature and signal analysis are generally thought to require complex, or at least specialized monitoring techniques due to the fact that many common statistical

Michael Sharp et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

analysis methods are built on the implicit assumption that the signal being processed is steady state. Specifically, the key difference in processing a transient signal versus a steady state one is the necessity of a temporal aspect to the analysis. This section provides a brief review of some of the more common methods of transient analysis. This is not meant to be an all-inclusive list, but merely to highlight a few of general-purpose techniques that can be applied to a wide variety of signals and equipment.

The simplest techniques involve windowed tracking of statistical properties over the progression of the signal. These, along with more complicated techniques (Gabor Transform, Hilbert Huang Transform, Wavelet Analysis, etc.), capture particular aspects and features of a signal as it evolves over time. The general trade-off for most transient analysis techniques is between the resolution of the signal feature and time. The Gabor Transform most notably suffers from this trade-off. While the Hilbert Huang and to some degree Wavelet analysis do not directly suffer from this trade-off, there is some loss in direct physical meaning of the transforms.

Many of the traditional methods and algorithms for the analysis of transient signals are particularly suited for a specific type of signal or application. One common yet powerful tool, particularly in the case of oscillatory signals, is the Fourier transform. This method, extended by the Gabor Transform [Gabor 1946], can be used to identify evolutionary frequency shifts within a signal. Unfortunately when using frequency or time/frequency analysis, it is not always easy to discover and isolate the minute changes that may be significant in regards to the health of the system, but become over shadowed by less important but more constant frequencies shifts which are merely products of the transient. Physics models of the system can help to overcome this by providing a guide to the expected dominant degradation modes. When this type of physics model, or related information is not available however, the use of the Hilbert Huang Transform (HHT) can serve to help identify this same type of information [Huang 1998].

Unlike the Gabor transform, the HHT is not based on a preconceived parametric functional form. Whereas the Fourier transform can be thought of as deconstructing the signal into a series of sine waves to determine frequency, the HHT decomposes the signal into successive intrinsic mode functions (IMFs) based on the characteristics of the signal itself. By then applying the Hilbert Transform to these IMFs, analytic amplitude and an instantaneous frequency can be obtained for each IMF. Because of this, the HHT excels at highlighting the instantaneous dominant frequencies of a signal at any given point in time. By examining the multiple layers of Intrinsic Mode Functions (IMFs) and the instantaneous frequency of each, subtle frequency shifts which may evolve over time stand out, both

in time and frequency over larger more dominant frequencies. These can be used in the form of the HHT, or as a guide to go back and reevaluated the simpler to calculate JTFS. Not only can the HHT provide frequency information, but information regarding power and amplitude as well. Though the HHT is typically computationally costly to calculate, the information it provides can be invaluable in the prognostic analysis, either from direct use or as a guide to identifying more simple trackable features.

The problem with each of these methods is that after suitable features have been found and extracted, some form of model must be additionally developed in order to quantify any degradation in these features. This is not always a trivial task, especially when the pertinent feature is represented by a certain signature based on the transient itself such as the rate of increase for the instantaneous frequency of a signal's second IMF over a 3000 observation window. Time distribution mapping, as will be shown in this paper, can be used either directly on a signal or on these extracted features to solve this problem.

### 3. TIME DISTRIBUTION MAPPING

A major drawback to all the previously discussed methods of transient signal analysis is the lack of widespread instant applicability. In other words, most techniques do not generalize across different systems or equipment without some additional knowledge and research into understanding the system. For example, although the Hilbert Huang Transform is highly adaptive and able to pick out subtle features in a given signal, there is no knowing in which level, and at what point in the decomposition an important feature will surface. With understanding of a system, one may infer, or expect to find features of interest at certain points, but these points will change between systems. This research seeks to provide and broadly examine generic processes for modeling and monitoring transient signals with the explicit goal of extracting a useable prognostic parameter.

Time distribution mapping via the Sharp Transform provides a novel method for expanding single or multiple signals into a non-uniform matrix space, or a similar cross-pattern space. In more general terms, this transform refers to taking successive measures of the empirical distribution of a signal throughout time and storing them in a serial temporally ordered fashion, thus creating a map detailing how the distribution of a signal evolves through time. For ease of interpretation and illustration, this section explains the algorithm for developing a model based on a single signal or variable. However, there is no reason that this transform could not be directly extended for the cross signal density of two or more.

The basics of this transformation rely on the creation and understanding of a signal's distribution, or probability density. An empirical measure of a signal's distribution over any given interval can easily be calculated by taking a simple histogram of the signal over that window. In order to become a true probability density, scaling based on the window size must be enforced, but as will become apparent later, this scaling can be factored out. All necessary processing can be accomplished with the unscaled discrete distribution provided by the histogram. Unfortunately, the full temporal complexities of a transient signal cannot be expressed with a single distribution. Consider the transient signal below: this modulated oscillatory signal exhibits shifts in both amplitude and variance over the observed transient.

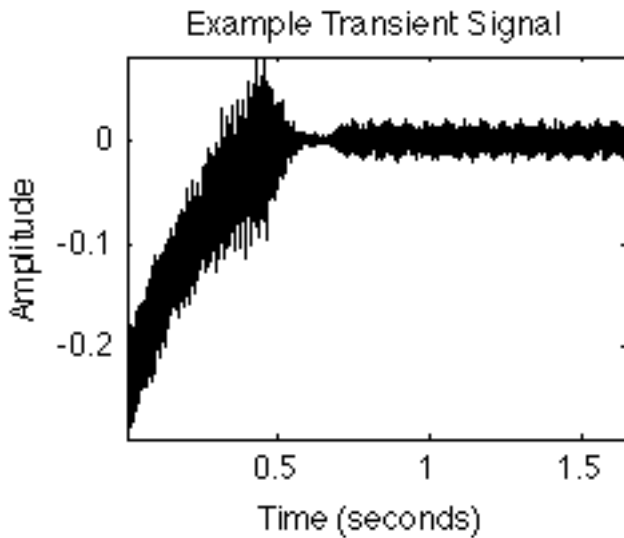


Figure 1 - Example Transient Signal

If one were to perform a standard series of tests on this data they could determine that it has a mean of -0.028, a standard deviation of 0.059 and upper and lower limits at 0.081 and -0.288 respectively. While most of this can be seen or estimated from a typical histogram such as the one shown below, none of this captures any of the highly temporal aspects of this signal.

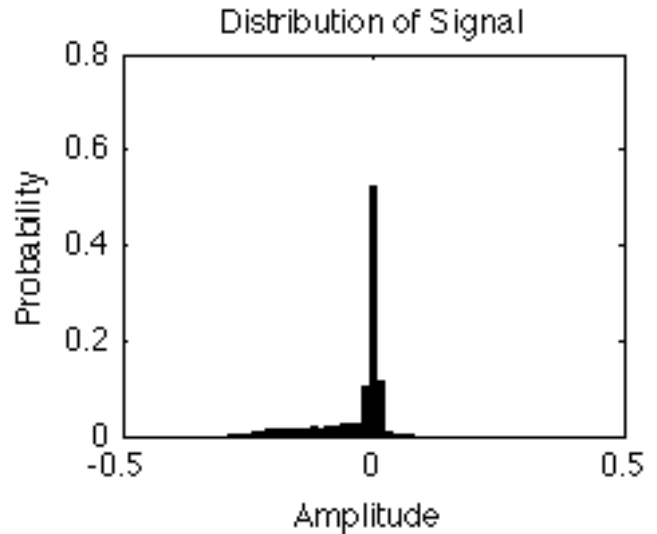


Figure 2 - Histogram of Example Transient Signal

However, if one were to calculate successive histograms of the signal through time and over smaller windows these temporal aspects are easily captured. Shown in the figure below is a series of histograms, each with 25 bins with varying bin locations based on the amplitude of the signal over each observation window. The selection of the number of bins as well as the window size can be altered based on the sample rate and time frame of the transient, but in general, a window size of greater than five times the number of bins is desirable.

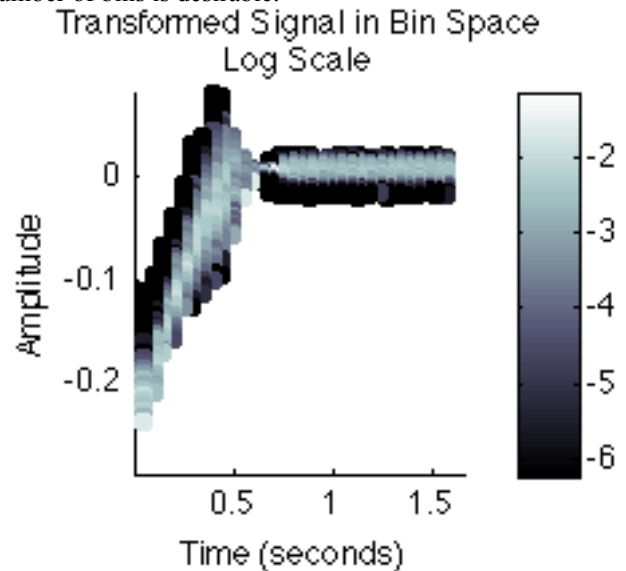


Figure 3 - Time Distribution Map of Oscillatory Transient Signal

This bin space map is able to capture both implicitly and explicitly many aspects of the signal as they evolve through time. Aspects regarding the mean, variance, and skewness as they evolve in time are all embedded within this ST map. Even gross changes in frequency are

implicitly captured in the information of this map. That is not to say that by looking at this map a non-dominant frequency shift of five hertz in one of the upper peaks would be observable, but a frequency shift significant enough to change the distribution on the scale of the observable window could be detected. In fact, the most obvious, simplest, and yet most all encompassing statement that can be made is that any alterations in the localized distributions through time are captured with this method.

Once the concept of transforming and mapping signals in the new bin space has been established, it is next important to explain how these can be used to develop prognostic parameters to be used in the creation of Remaining Useful Life (RUL) estimations. The first thing required is multiple “good” or nominal transient recordings. Between three and ten of these exemplars make a good baseline for developing the ideal reference case. Development of this transformed reference map assumes that the transients in question exhibit similar behavior. If they do not, then multiple reference maps should be made for each category or pattern of transient. For example, load shifts from half to full load may not be the same as those from one quarter to full load, but all startups from zero to full load may be expected to look the same.

After the nominal transients are collected and appropriately grouped, they need to be divided into equally spaced segments based on the chosen window size. The selection of both the number of bins and the window size is highly dependent on the observed length of the transient, the sample rate, and the level of detail desired versus the robustness of the distributions. While it is intuitive that greater number of bins provides greater detail in the map itself, greater numbers of bins also require larger windows to develop full distributions, losing some of the time resolution of the map. In practice segmenting the transient signal into between approximately 30 and 100 windows and choosing a bin number about one tenth the window size seems to produce reliable results. Of course this is dependent on the available data and can be tailored to suit any needs. Also, in this discussion, the windows are treated as discrete and wholly separate from each other, there is no reason that the windows could not overlap or even “scroll”, over each observation in time. It is simply ignored for this discussion to simplify both computations and explanation.

Once the window size and number of bins has been determined a “master” bin map can be created. The maximum and minimum values inside each window observed over all the exemplar cases define the range of the bins for that time window. Unless there is special reason not to, the bin edges can then easily be defined as linearly spaced points between these ranges. This master bin map will now be the bin edges that will define the new bin space into which all subsequent transient signatures will be

transformed into. The transformed maps of each of the exemplar cases based on these bin locations will then be averaged together to complete the creation of the reference map. With this reference map, and the master bin map, it is possible to create progressive bin space maps throughout the lifetime of the equipment, which can then be compared to create a measure of change that is relatable to degradation. By monitoring the summed square of the residuals between any given lifetime bin space map and the reference time distribution map, a quantitative value relatable to the overall degradation of the transient can be made. This can then be directly translated into a measure of health of the system based on the original signal. This value and its progression through time are then directly usable as a prognostic parameter for models estimating RUL.

It is possible for the prognostic parameter developed by this method to saturate if a signal experiences a mean shift greater than the span of the reference ST map, but this is easily overcome by simply developing a new reference ST and master bin map based on these new levels of degradation. In order to compare separate units, the residuals from the reference ST may need to be scaled by the residuals associated with the exemplar cases used to create it, but this is a trivial task and should not impair the results of the analysis.

This generic idea of mapping the progressive aspects of a signal over time through windowing and comparing them to a reference signal is the most widely applicable method of monitoring transients for degradation. Similar mappings of Joint Time Frequency Spectra (JTFS) or similar statistical evolutions can also use this methodology. Mapping of signals transformed into an empirical bin space is presented as the best generic test with the broadest number of implicit anomaly detections possible. This is not to say it is the best for every given application, but its’ broad range of applicability make it useful in many cases. Brief case studies of the development and use of this STDM modeling technique are provided in subsequent sections.

#### 4. PROOF OF CONCEPT TEST CASES

The following sections provides multiple proof of concept case studies that solidify the hypothesis that not only are these techniques valid in processing transients with the goal of prognostic modeling, but that transients in general are valid and reliable sources of degradation and system health information. These experiments utilize other more traditional analysis methods to compare and validate the more novel transformation techniques for transient analysis presented in this paper.

##### 4.1. Impeller Degradation Experiment

The first of these experiments analyzed the artificially induced degradation of small-scale neoprene horizontal

pump impellers. The pumps were chosen as analogous representations of the large pumps found in many industry and power plants applications. The focus of this experiment was to identify correlations between monitorable features found during pump startup and the amount of degradation found in the system. Twelve separate impellers were degraded over time and failure was defined as the inability to self-prime.

In order to precipitate faults in the impellers, increasingly large notches were cut into each impeller vane between regular data collection runs of the pump. These notches are analogous to the type of degradation an impeller might suffer due to a spur or defect in the impeller housing or due to particulates in the fluid. The pump differential pressure, vibration, and inrush current were recorded at ~1.6kHz for analysis in this experiment.

The analysis of each of these signals via more traditional, means discovered multiple features within each that directly relate back to priming time of the pumps. It was discovered that the priming time was an acceptable features that could be used as an indication of overall impeller health, and could be used as a prognostic parameter.

For vibration, the most pronounced of these features was an increase in amplitude as well as a rise in the vane pass frequency prior to the water entering the impeller housing. While the rise in frequency can be viewed via a Joint Time Frequency Spectrum, both features are easily extracted using the Hilbert transform. The figure below shows how this rise in amplitude evolves over the life of the impeller.

Vibration Analytic Amplitude Over Impeller Life

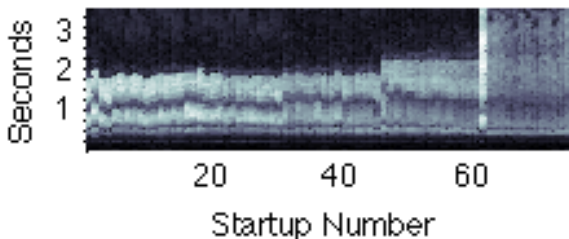


Figure 4 - Analytic Amplitude of Startup Vibration Over Single Impeller Lifetime

This figure shows that the elevated vibration amplitude reflecting priming time increases from approximately 1.8 seconds early in life, to around 3 seconds near the end of life. Extracting this time requires can be somewhat trivial task, but it still requires additional processing and model development.

Conversely, instead of calculating this amplitude and then making a model for extracting the amount of the elevated amplitude, nearly identical information can be extracted

without special development through time distribution mapping via the transform detailed above. Using several start of life vibration signatures to create a master bin map, the residual mappings from the transformed bin space come out as a near exact echo of that information.

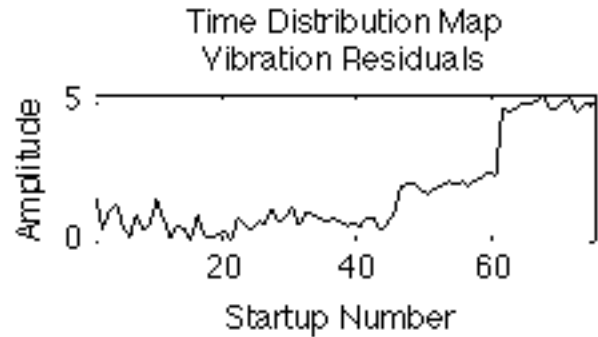


Figure 5 – Exemplar Impeller Bin Space Mapping Residuals

Each lifetime startup vibration signature is transformed into the same master bin space. The summed scaled residuals between these and the reference bin map implicitly extract the difference in priming time without a need to optimizing towards that goal.

Similar extractions can be done for both the inrush current and the differential pressure. Analysis shows that the current exhibits a drop in power prior to priming due to the lessened load, and unsurprisingly the differential pressure requires more time to reach its' maximum value. Information relating to these features can quickly be gleaned from the respective residuals of their separate time distribution maps as shown in the figure below.

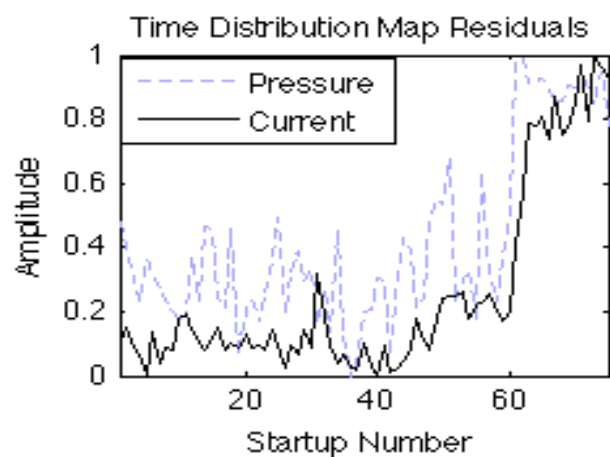
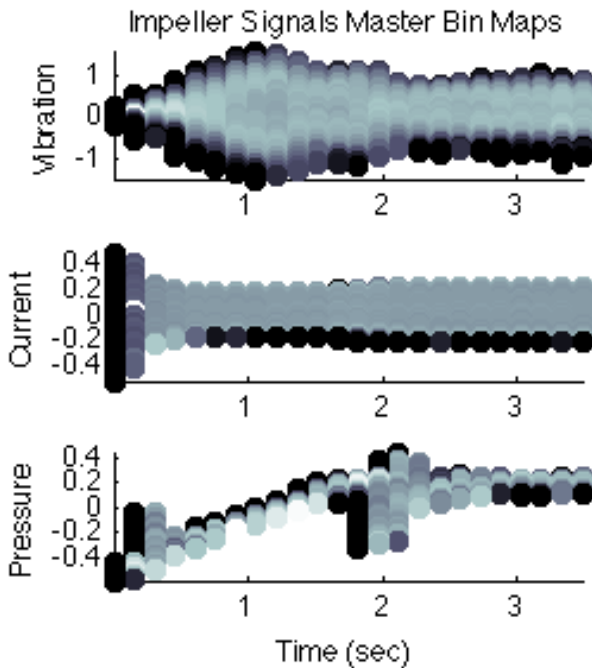


Figure 6 - Exemplar Bin Space Residuals From Additional Signals

Notice that despite the very different nature of the three signals, they all contain information related to the same system and as such the residuals of the bin space



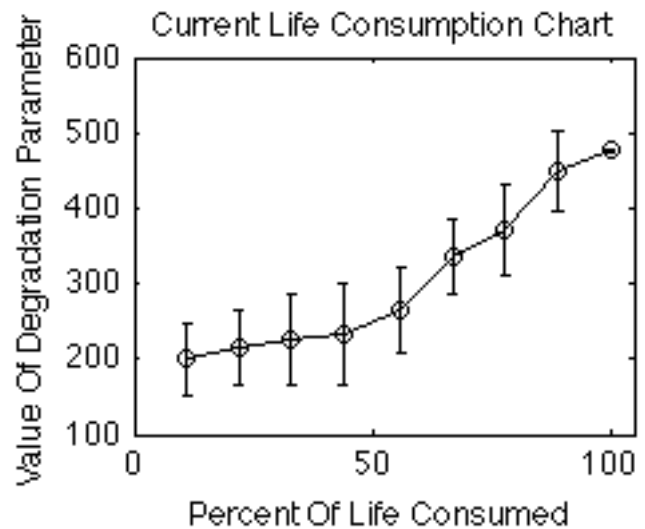
transformation have comparable information in them. The additional noise associated with the pressure residuals comes from the signal itself which exhibited significant noise. This is most likely do to voids in the sensing lines. The figure below presents the master bin maps created as the nominal startup signature for each signal.



**Figure 7 - Transformed Impeller Signals' Mater Bin Maps**

Where as differential pressure exhibits a degradation of a significantly mean shifting signal, both the current and vibration signals have an alteration in their variance shifts. Additionally, the current signal is highly modulated on the 60Hz electrical carrier wave, yet transformations of each of these signals into similarly windowed bin spaces is able to quickly provide the pertinent degradation information.

The residuals from each of these mappings can be combined together to create a more robust measure of overall priming time, and by inference, impeller health. Due to the lack of available comparison data regarding the true lifetime or rate of wear time of these impellers, direct estimation of Remaining Useful Life (RUL), as would be more traditional in prognostic applications [Hines 2009], could not be made, and it would be meaningless to attempt with this data. Instead a percentage of current life consumption (CLC) model is inferred such that with the addition of a known lifetime run, a RUL estimate model could be made. The figure below represents the CLC chart with standard error uncertainty bounds on each calculated point.



**Figure 8 - Impeller Current Life Consumption Chart**

This figure is read by calculating the optimally weighted combination of the rescaled bin map residuals on the vertical axis, then estimating the corresponding percent of life consumed on the horizontal axis. It has been show that utilizing genetic algorithms to find an optimal linear combination of trendable parameters can increase both the robustness and accuracy of a prognostic parameter [Coble 2011]. Using this technique to combine the residuals shows how the process of transforming signals into a non-uniform time/bin space can directly lead to trendable prognostic parameters.

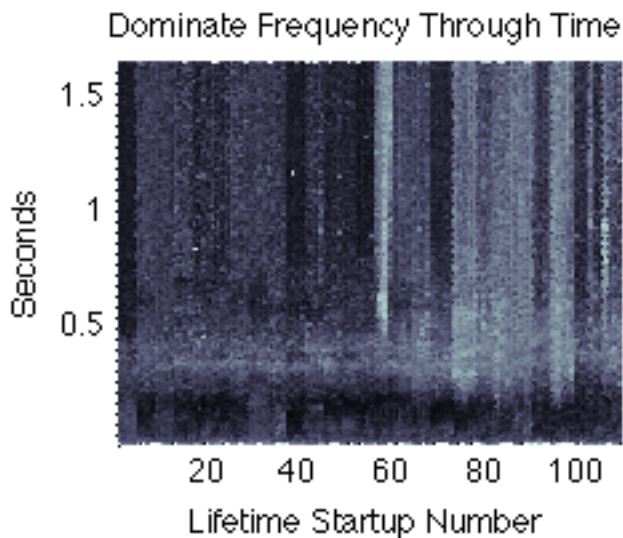
#### 4.2. Motor Aging Experiment

Another experiment conducted to verify the robustness of the bin space mapping technique was an accelerated induction motor aging experiment. In this experiment, 5HP motors were subjected to cyclical thermal stresses and tested between each cycle. Several sensors were recorded during each test startup to monitor the overall state and health of the motor. These included current, voltage, vibration and acoustic signals, each sampled at a frequency of just over 10kHz.

As these motors aged, it became apparent that the dominate failure mechanism was due to bearing fatigue. As there is much work already in the literature regarding the steady state quantification of bearing degradation, this experiment was not meant to replace those techniques, but instead to augment them through transient analysis and to verify the validity of the empirical non-uniform bin space transformation and the associated mapping technique.

To conserve space, not all of the signals analyzed will be fully illustrated here, but instead a brief overview of the signals is given, highlighting the different types of signals.

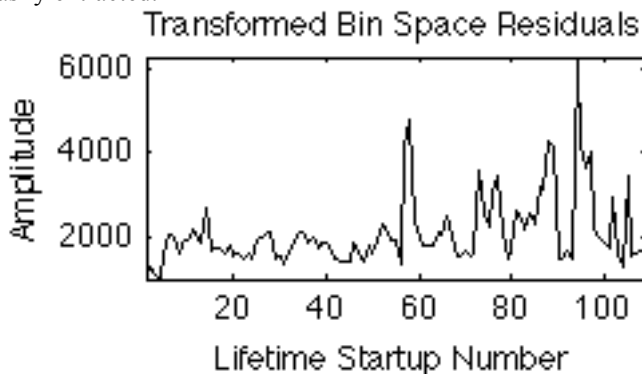
The first signal analyzed is the acoustic signature. This signal exhibits a shift in frequency and amplitude over the life of the signal. Directly and reliably extracting this requires both the Hilbert transform and significant filtering of the returned parameters. Although contain similar informational aspect relating to the degradation of the bearing itself, the more easily visually distinguished is the startup dominate frequency signatures which is shown in the figure below.



**Figure 9 - Acoustic Dominate Startup Frequency Over Motor Lifetime**

The vertical axis in this chart represents the amount of time after the energizing the motor, where the lifetime startup number is on the horizontal axis. The lighter tones represent the higher frequencies, and notice how these become more dominate towards the end of life, especially in the time before approximately 0.5 seconds after energizing.

Now examining the residuals from the time distribution map of the transformed bin space, this increasing deviation is easily extracted.



**Figure 10 – Startup Acoustic Transformed Bin Space Residuals**

These residuals actually seem to reflect a combination of the information from the frequency as well as the amplitude shifts over time. This is not only to be expected, but in many cases desirable as it automatically combines different aspects of the signal degradation into a single easily monitored value.

Another interesting signal to apply this transformation and mapping technique to is the supply voltage of the motor. This signal has properties such that, on gross inspection alone, make it appear to be a steady state signal, despite the fact that it is produced during a transient. Yet upon examination of the residuals for the signal, we see a similar clear progression as the overall motor system degrades.

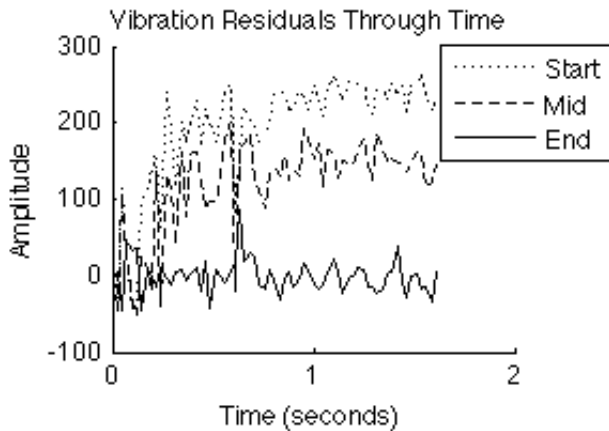


**Figure 11 - Startup Supply Voltage Transformed bin Space Residual**

In order to find this same level of information about the deviation in this signal through traditional algorithms several levels of demodulation or decomposition of the voltage would be required.

In fact, all of the signals show progressive trends as the motor is aged, but the degree of their increase and the noise level obviously are signal dependant. Again this is expected as the signals all relate to the same system, but are capturing different aspect of the system.

Another useful aspect of the developed transform is that since the bin maps have a temporal aspect, information regarding when in the transient as well as what type of deviations occur can be obtained. The figure below shows the progression of the motor vibration signature’s residuals through the transient time.



**Figure 12 - Startup Vibration Transformed Residuals in Time**

Here the horizontal axis represents the time from energizing the motor and each line represents a particular lifetime startup at the beginning, middle, and end of life. This provides information showing that in fact for vibration, the largest differences in startup come after the one-second mark. This is unlike the each of the other signals, which exhibit their largest changes prior to that time. The reason for this can be traced back to the increased steady state level of vibration through out the life of the motor.

Each signal provides easily monitored and trended residuals that can be quickly implemented by a myriad of prognostic methodologies to create Remaining Useful Life predictions.

## 5. CONCLUSION

The work presented in this paper shows time distribution mapping and the transformation of a signal into a non-uniform bin space based on the empirical aspects of a transient signature to be a clean and elegant method of generically quantifying the degradation of a transient signature. This method is able to simultaneously capture several types of statistical shifts within a signal without need to explicitly search for any one in particular. Furthermore, this transformation and subsequent mapping is shown to be robust over various forms of non-stationary signals. It is this indifference to modulation or signal source that allows it to quickly analyze and quantify a myriad of signals for a single application without the need to tailor individual algorithms to each specific signal. By capturing both statistical and temporal aspects of a signal simultaneously, time distribution mapping via the developed transform is shown to be a versatile and generically applicable tool to aid in transient analysis and to augment existing techniques.

## ACKNOWLEDGEMENT

The authors would like to acknowledge the Electric Power Research Institute (EPRI) for providing funding of this project. The author would also like to thank Nicholas Sly for his aid in collecting parts of the data used in this paper.

## REFERENCES

- Coble, Jamie, "Merging Data Sources to Predict Remaining Useful Life – An Automated Method to Identify Prognostic Parameters", Doctorial Dissertation, University of Tennessee, Knoxville TN, 2011.
- Gabor, D., "Theory of Communication", *J. Inst. Electr. Engin.* (London) 93 (III), 1946.
- Hines, J. Wesley. "Empirical Methods for Process and Equipment Prognostics". Reliability and Maintainability Symposium RAMS. 2009.
- Huang, N.E. , Z. Shen, S.R. Long, M.C. Wu, H.H. Shih, Q. Zheng, N.C. Yen, C.C. Tung and H.H. Liu, "The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Nonstationary Time Series Analysis", *Proc. Royal Society of London, A*, Vol. 454, pp. 903- 995, 1998.

## BIOGRAPHIES

**Michael E. Sharp** is a Graduate Research Assistant and PhD candidate for the University of Tennessee Nuclear Engineering department. He received both his Bachelors and Masters of Science in Nuclear Engineering from the University of Tennessee and expects to receive his PhD in August of 2012. His research includes empirical modeling methods, signal processing and prognostic methodologies. Currently working under Dr. J Wesley Hines, the department head for the UTK Nuclear Engineering Department, Michael has worked on projects for a myriad of companies and organizations including National Aeronautics and Space Administration (NASA), Oak Ridge National Laboratory (ORNL), the Electric Power Research Institute (EPRI), and Baker Hughes Incorporated (BHI).

**J. Wesley Hines** is the Nuclear Engineering Department Head at the University of Tennessee and director of their Reliability and Maintainability Engineering program. He is a past nuclear qualified submarine officer and received an MBA and both an MS and Ph.D. in Nuclear Engineering from The Ohio State University. Dr. Hines teaches and conducts research in artificial intelligence and advanced statistical techniques applied to process diagnostics, condition based maintenance, and prognostics. He has authored over 250 papers and has several patents in the area of advanced process monitoring and prognostics techniques.

# Prognostics Method for Analog Electronic Circuits

Arvind Sai Sarathi Vasan<sup>1</sup>, Bing Long<sup>2</sup>, and Michael Pecht<sup>1</sup>

<sup>1</sup>*Center for Advanced Life Cycle Engineering, University of Maryland, College Park, MD 20742, USA*

*Arvind88@calce.umd.edu  
pecht@calce.umd.edu*

<sup>2</sup>*School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu, China*

*longbing@uestc.edu.cn*

## ABSTRACT

Analog electronic circuits are an integral part of many industrial systems. Failure in such analog circuits during field operation can have severe economic implications. The presence of an expert system that can provide advance warnings on circuit failures can minimize the downtime and improve the reliability of electrical systems. Through successive refinement of circuit's response to a sweep signal, features are extracted for fault prognosis. From the extracted features, a fault indicator is developed. An empirical model is developed based on the degradation trend exhibited by the fault indicator. Particle filtering approach is used for model adaptation and remaining useful performance estimation. This framework is completely automated and has the merit of implementation simplicity. The proposed framework is demonstrated on two analog filter circuits.

## 1. INTRODUCTION

Analog circuits are extensively used in industrial systems for implementing controllers, conditioning signals, protecting critical modules and more (Wagner & Williams, 1989). The occurrence of circuit failures during field operation will affect the system functionality and the cost of failure can be enormous (Pecht & Jaai, 2010). In most cases, these failures can be related to a fault in the system's analog circuitries, where fault refers to a change in the value of a circuit component from its nominal value which leads to a failure of the whole circuit. These faults could either be catastrophic (open and short) or parametric faults (fractional deviation in circuit components from their nominal values) (Xiao & He, 2011). For example, the degradation of electrolytic capacitors in the LC filters will cause the switching-mode power converters to fail (Chen, Wu, Chou,

& Lee, 2008); an increase in resistance offered by the components within filter circuits due to the degradation of solder joints will affect the frequency band being filtered out (Brown, Kalgren, Byington & Roemer, 2007).

Prevention of circuit failures during field operation requires methods for the prediction of remaining useful performance (RUP) of the failing circuit (Lall, Bhat, Hande, More, Vaidya, & Goebel, 2011) where RUP refers to the remaining time that the circuit performance guarantees system operation. Most of the related research for analog circuits has aimed at diagnosing faults in circuits. However, no method has been suggested for predicting the remaining time till circuit failure.

In this paper, a new prognostics framework for analog circuits is proposed, aiming at "in-circuit" real-time RUP estimation. A fault indicator (FI) reflecting the evolution of fault in any of the circuit's critical component is developed. Then a model adaptation scheme using particle filters (PFs) is devised for tracking the FI and predicting the circuit's RUP.

## 2. PROGNOSTICS FRAMEWORK

Developing PHM strategies for analog circuits is a challenging task because of the unavailability of fault models (Aminian & Aminian, 2002). This makes the application of data-driven techniques appealing, as this does not require knowledge on material properties, structures, or failure mechanisms. An overview of the proposed prognostics framework is shown in Figure 1.

### 2.1. Feature Extraction

For analog circuits, the behavioral characteristics are assumed to be embedded in the time and frequency response. Hence, a test signal must be chosen such that this response can be generated and captured. Hence, we suggest the use of a sweep signal containing a frequency bandwidth larger than that of the CUT.

Arvind Sai Sarathi Vasan et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

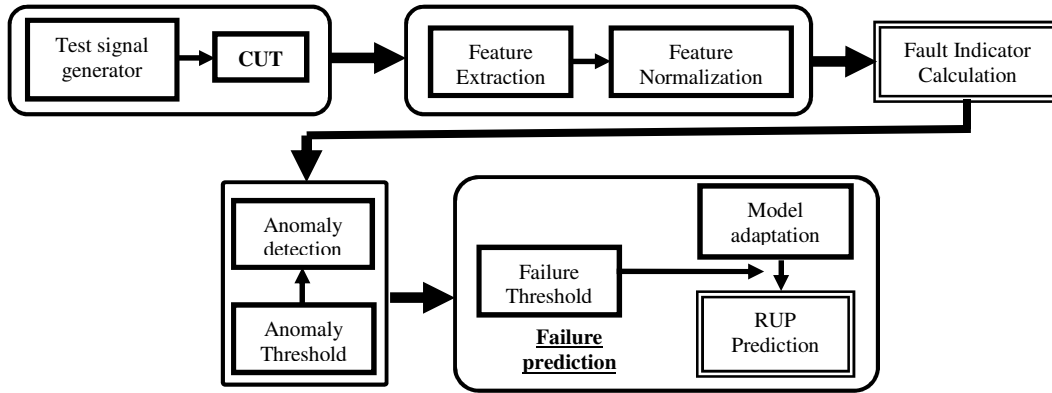


Figure 1. Overview of the proposed prognostics framework for analog electronic circuits.

This ensures that the CUT is excited by all the frequency components it is sensitive to. Also, this method allows us to acquire data at a sampling rate that depends on the CUT's bandwidth. Simple and cost effective digital forms of a sweep signal generator have been implemented with TTL logic and thus, can be incorporated on-board for testing.

From the CUT response to sweep signal, two types of features are extracted, which we call the wavelet and statistical features.

Fourier transformation is commonly used for extracting information embedded in a signal. However, Fourier transformation is suitable only for the analysis of stationary signals. A change in time of a non-stationary signal spreads over the entire frequency domain, which will not be detected through Fourier transformation. This is a drawback for fault diagnosis, as the signals to be analyzed contain time-varying frequencies. In contrast, wavelet analysis can reveal signal aspects such as trends, break points, and discontinuities. Hence, we choose wavelet features for RUP estimation in analog circuits.

In wavelet analysis, the signal's correlation with families of functions that are generated based on the shifted and scaled version of a mother wavelet  $\psi(t)$  is calculated and used to map the signal of interest to a set of wavelet coefficients that vary continuously over time. The discrete version of the wavelet transform consists of sampling the scaling and shifted parameters, but not the signal or the transform. This makes the time resolution good at high frequencies, and the frequency resolution becomes good at low frequencies.

In this work, through discrete wavelet transformation, we decompose the response of the CUT stimulated by a sweep signal into the approximation and detail signals using multirate filter banks. The information contained in the response is represented using features extracted by computing the energy contained in the detail coefficients at various levels of decomposition. This is expressed as follows:

$$E_j = \sum_k |d_{j,k}|^2; \quad j = 1:J \quad (1)$$

where  $d_{j,k}$  are the energy and  $k^{th}$  detail coefficient at  $j^{th}$  level of decomposition respectively.

The second set of features extracted includes the kurtosis and entropy of the CUT's response. Kurtosis is a statistical property which is defined as the standardized fourth moment about the mean. It provides a measure on the heaviness of the tails in the probability density function (PDF) of a signal, which is related to the abrupt changes in the signal having high values and appearing in the tails of the distribution. Kurtosis is mathematically described as follows:

$$kurt(x) = \frac{E[x-E[x]]^4}{[E[x-E[x]]^2]^2}. \quad (2)$$

On the other hand, entropy provides a measure of the information capacity of a signal, which denotes the uncertainty associated with the selection of an event from a set of possible events whose probabilities of occurrences are known. It is defined for a discrete-time signal as:

$$H(x) = -\sum_i P(x = a_i) \log P(x = a_i) \quad (3)$$

where,  $a_i$  are the possible values of  $x$ , and  $P(*)$  are the associated probabilities.

## 2.2. Fault Indicator

When implementing prognostics at a component level, generally a fault indicator (FI) parameter is identified to monitor the degradation of the component in real-time. This parameter (e.g., the ON state  $V_{CE}$  of an IGBT (Patil, Celaya, Das, Goebel, & Pecht, 2009) and the RF impedance for interconnects (Kwon, Das & Pecht, 2010)) is chosen based on an understanding of the degradation process. From this perspective, multiple parameters corresponding to all the critical components/sub-systems need to be monitored and

processed in real-time to perform system level prognostics. This is not possible in applications like analog circuits, where there is a constraint on the available resources. So to address this challenge, we have developed a method to construct a FI to represent circuit degradation. Here, circuit degradation refers to the degradation in any of the circuit's critical component which leads to the deviation of component's value from its nominal value. The procedure for calculating the FI is based on the well-known Mahalanobis Taguchi methodology (Taguchi & Jugulum, (2002), Soylemezoglu, Jagannathan, & Saygin (2010)).

The FI calculation starts with the collection of the two feature sets (wavelet and statistical features). For both the feature sets, two individual Mahalanobis spaces (MSs) are constructed using their normalized feature elements and correlation coefficients.

Let  $\bar{F}^1 = \{f_1^1, f_2^1, \dots, f_l^1\}$  and  $\bar{F}^2 = \{f_1^2, f_2^2\}$  denote the wavelet and statistical features respectively, where  $f_i^1; i = 1:l$  denotes the energy of the detail coefficients at the  $i^{th}$  decomposition level,  $l$  is the number of decomposition levels, and  $f_1^2$  and  $f_2^2$  are the kurtosis and entropy of the CUT's response. A data set formed using the feature set  $[\bar{F}_j^1, \bar{F}_j^2]; j = 1:n$  with  $n$  observations made during the no fault condition is used as training data.

The mean and standard deviation of each feature element is calculated using the training data, from which the training feature sets in the no fault condition are normalized:

$$Z_{i,j}^1 = \frac{f_{i,j}^1 - \bar{x}_i^1}{s_i^1} \text{ and } Z_{k,j}^2 = \frac{f_{k,j}^2 - \bar{x}_k^2}{s_k^2}; \quad i = 1:l; k = 1,2; j = 1:n \quad (4)$$

where

$$\bar{x}_i^1 = \frac{\sum_{j=1}^n f_{i,j}^1}{n}, \quad \bar{x}_k^2 = \frac{\sum_{j=1}^n f_{k,j}^2}{n}, \quad s_i^1 = \left( \frac{\sum_{j=1}^n (f_{i,j}^1 - \bar{x}_i^1)^2}{n-1} \right)^{1/2},$$

$$\text{and } s_k^2 = \left( \frac{\sum_{j=1}^n (f_{k,j}^2 - \bar{x}_k^2)^2}{n-1} \right)^{1/2}. \quad (5)$$

After normalizing the feature sets, the Mahalanobis distances (MDs) are calculated during the no fault condition using the following mathematical expressions:

$$MD_j^1 = C^{-1} (Z_{i,j}^1)^T (\Sigma^1)^{-1} (Z_{i,j}^1) \text{ and}$$

$$MD_j^2 = C^{-1} (Z_{k,j}^2)^T (\Sigma^2)^{-1} (Z_{k,j}^2) \quad (6)$$

where  $\Sigma^1$  and  $\Sigma^2$  represents the correlation matrices for the wavelet and statistical features respectively, and are calculated from the normalized feature sets as:

$$\Sigma_{i,j}^1 = \frac{1}{n-1} \sum_{m=1}^n Z_{i,m}^1 Z_{j,m}^1 \text{ and } \Sigma_{i,j}^2 = \frac{1}{n-1} \sum_{m=1}^n Z_{i,m}^2 Z_{j,m}^2 \quad (7)$$

where  $C$  denotes the total number of fault conditions, including the no fault condition.

In order to validate the MS, observations that fall outside the normal/no fault group are identified, and the corresponding MD values are calculated. The characteristics of the

abnormal group are normalized using the mean and standard deviation of the corresponding characteristics in the no fault group. The correlation matrix corresponding to the no fault group is used to compute the MDs of the faulty conditions. If the MS is constructed properly, then the MDs under the fault conditions should have higher values than the normal group. Thus, if we plot the MD values obtained from the two feature sets on a two dimensional plane, the fault conditions should be separable from the no fault condition, as shown in Figure 2. The figure also illustrates a typical trajectory in the two dimensional space representing circuit degradation.

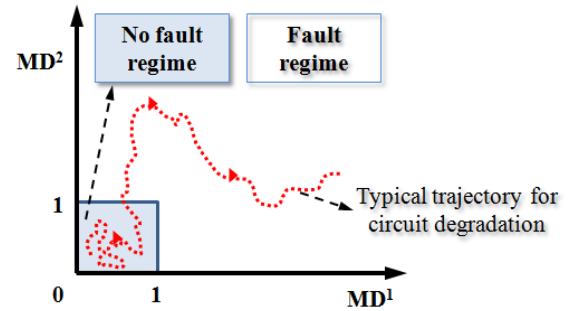


Figure 2. Illustration of fault propagation on the MS.

From Figure 2 it seems like one can observe the trajectory on the MS to understand circuit degradation. However, monitoring the trajectory in the 2D MS alone for prognostics has two drawbacks: 1) it is difficult to define a failure threshold, and 2) there can be abrupt changes in any of the MD values which are not easy to track. These drawbacks occur because of the difference in dynamic ranges for the MD values obtained from feature sets  $\bar{F}^1$  and  $\bar{F}^2$  (for example, for the sample circuit shown later, the range for  $MD^1$  is found to be around 0~10<sup>6</sup>, and for  $MD^2$  it is found to be around 0~500). This happens because of the difference in the number of feature elements present in each of the feature sets. To address this issue, we have built a FI that gives equal weight to the MD values obtained from different feature sets.

Once the MD values are calculated for the two feature sets, the FI is computed using the following transformation:

$$\phi_j = \frac{(MD_j^1)^{\frac{1}{n_1}} (MD_j^2)^{\frac{1}{n_2}}}{(MD_j^1)^{\frac{1}{n_1}} + (MD_j^2)^{\frac{1}{n_2}}}; j = 1,2.. \quad (8)$$

where  $n_1$  and  $n_2$  represents the number of feature elements in feature sets  $\bar{F}^1$  and  $\bar{F}^2$  respectively. The advantage of the FI defined above is that, it is invariant to abrupt changes in MD values. This is realized by scaling it using the number of elements present in each feature set. Our study indicates that the FI has an increasing trend with respect to the amount of deviation in the faulty component from its nominal value.

In the future, new feature sets might be suggested for fault diagnosis and prognosis of analog circuits. Even if new feature sets are proposed, the above suggested FI calculation method can still be made compatible by extending Eq. (8) as follows:

$$\wp_j = \frac{(\text{MD}_j^1)^{\frac{1}{n_1}}(\text{MD}_j^2)^{\frac{1}{n_2}} \dots (\text{MD}_j^r)^{\frac{1}{n_r}}}{(\text{MD}_j^1)^{\frac{1}{n_1}} + (\text{MD}_j^2)^{\frac{1}{n_2}} + \dots + (\text{MD}_j^r)^{\frac{1}{n_r}}}; j = 1, 2, \dots, r \quad (9)$$

where  $r$  denotes the total number of feature sets extracted, and  $n_i$  and  $\text{MD}^i$  denotes the total number of feature elements and the MD values corresponding to the feature set  $i$ , respectively.

### 2.3. Anomaly Detection Threshold

A threshold for anomaly detection using the MT method is traditionally defined as  $\text{MD} = 1$ . Here, in the calculation of the FI, a MD of 1 for both the feature sets would result in a FI value of 0.5. According to the MT method, the circuit is supposed to be displaying abnormal behavior when the FI value reaches 0.5. The interesting aspect of our study was that whenever the FI value crossed 0.5, the faulty component was found to just vary outside its tolerance range. Thus, a value of 0.5 for the FI is used as the threshold for anomaly detection.

## 3. FAILURE PROGNOSIS USING PFS

Failure prognosis is often performed by generating long-term predictions of FI signal until a predetermined failure threshold is reached (Orchard & Vachtsevanos, 2009). Since uncertainty is inherent to such prediction processes, the evolution of the FI is generally modeled as a stochastic process, and estimates of the RUP is made in the form of PDFs. For this purpose, stochastic nonlinear filters have garnered a lot of interest among the research community. The procedure for estimating RUP using stochastic filters involves estimation of the current health state of the system, and then performing  $p$ -step predictions on the future health state. These two steps are discussed next in the following subsections.

### 3.1. Particle Filters

One of the most commonly used forms of approximate nonlinear filters in the PHM arena is the Particle Filter (PF). Many variations of the PF are available. However, we shall focus on the sampling and importance resampling (SIR) form of the PF. The idea behind the PF is that the posterior PDF is represented by a set of random samples with associated weights  $\{x_{0:k}^{(i)}, w_k^{(i)}\}_{i=1}^N$ ;  $w_k^{(i)} \geq 0, \forall k$  and the Bayesian estimates are computed based on these samples (or particles) and their weights:

$$\sum_{i=1}^N w_k^{(i)} \delta(x_k - x_k^{(i)}) \xrightarrow{N \rightarrow \infty} p(x_k | z_{1:k}) \quad (10)$$

where  $\delta(*)$  is the Dirac delta function. In practice,  $p(x_k | z_{1:k})$  is usually not known. Hence, the samples  $x_{0:k}^{(i)}$  are chosen from the importance density  $q(*)$  and their associated (normalized, such that  $\sum_i w_k^{(i)} = 1$ ) weights are chosen using the principle of importance sampling, which is expressed as:

$$w_k^{(i)} = \frac{p(z_k | x_k^{(i)}) p(x_k^{(i)})}{q(x_k^{(i)} | z_{1:k})}. \quad (11)$$

If the importance density function is chosen to be  $q(x_k^{(i)} | x_{k-1}^{(i)}, z_{1:k}) = p(x_k^{(i)} | x_{k-1}^{(i)})$  then the weights can be updated using the following relation:

$$w_k^{(i)} = w_{k-1}^{(i)} p(z_k | x_k^{(i)}). \quad (12)$$

Resampling is used to address issues introduced by the degeneracy of particles, where after a few iterations all but one particle have negligible weights. During resampling particles with small weights are eliminated, allowing us to concentrate on the particles with larger weights.

### 3.2. RUP Estimation

When the threshold for anomaly has been reached, the  $p$ -step prediction is generated. However, during the prediction process the weights of the samples are kept constant and are not updated as there are no measurements. At each prediction step, the predicted health state is checked with the failure threshold. The prediction time at which the FI crosses the failure threshold denotes the time at which the system is predicted to fail. RUP estimate is then obtained by computing the distance between the predicted time of failure and the current time instant. The PDF for RUP is obtained by finding the RUP for all the  $N$  paths traversed by the  $N$ -particles, and then associating them with their weights. We can approximate a prediction distribution ( $p$ -steps forward) as follows:

$$p(x_{k+p} | z_{1:k}) \approx \sum_{i=1}^N w_k^{(i)} \delta(x_{k+p}^{(i)}) dx_{k+p}. \quad (13)$$

## 4. RESULTS AND DISCUSSION

In this section, we demonstrate our PHM framework on two analog circuits. A Sallen-Key band-pass filter centered at 25 kHz and a biquad low-pass filter with a 10 kHz upper cut-off frequency are the sample circuits built to examine our approach (Figure 3). The circuit elements have a tolerance range of 10%. Features extracted when all the components vary within their tolerance range belong to the no-fault (NF) class. Faulty responses are obtained when any of the critical components vary beyond their tolerance range.

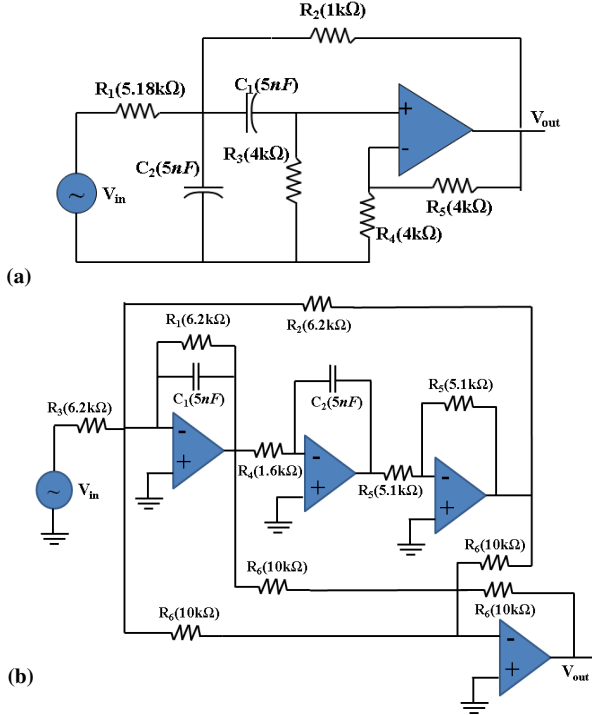


Figure 3. Sample circuits used with their component's nominal value (a) 25 kHz Sallen-Key band-pass filter and (b) biquad low-pass filter with an upper cut-off frequency of 10 kHz.

Component degradation is accompanied by a gradual change in component value (might increase or decrease). This change can have a significant impact on the circuit performance. In Section 2, the FI was developed to denote the degradation in analog circuits. Figure 4 shows the trend exhibited by the FI as the components C2 and R3 of Sallen-Key band-pass filter, and C1 and R4 of the Biquad-low pass filter, deviates from their nominal value. Here, we assume that the fault level increases gradually with respect to time, where, with each time index, the fault level increases by 0.4%. In this context, time index refers to one cycle where a 100 msec test signal stimulates the CUT and the response is measured. Through curve fitting on the FI data (obtained from features extracted during different simulated fault conditions – simulation performed using PSPICE), it is found that a model of the following form can describe the FI trend of different components in different analog circuits well:

$$\wp_{t_k} = a_{t_k}^{(1)} \left[ - \left( \frac{t_k - b_{t_k}^{(1)}}{c_{t_k}^{(1)}} \right)^2 \right] + a_{t_k}^{(2)} \left[ - \left( \frac{t_k - b_{t_k}^{(2)}}{c_{t_k}^{(2)}} \right)^2 \right] \quad (14)$$

where  $\wp$  is the FI,  $t_k$  is the time index; and  $a_{t_k}^{(1)}$ ,  $b_{t_k}^{(1)}$ ,  $c_{t_k}^{(1)}$ ,  $a_{t_k}^{(2)}$ ,  $b_{t_k}^{(2)}$  and  $c_{t_k}^{(2)}$  are the model parameters.

In order to perform prognosis, the above regression model is exploited. However, to deal with the uncertainties caused by

component tolerances and usage conditions, the model is assumed to be stochastic. Hence, the parameters of the model are subjected to Gaussian distribution. This stochastic model is fed into the online prognostics routine. Features extracted from the circuit's response are used to calculate the current FI value. This is used to estimate the model parameters. Once the diagnostic routine detects an anomaly, it triggers the prognostics module. Here, the above model's parameters are incorporated as the elements of a state vector.

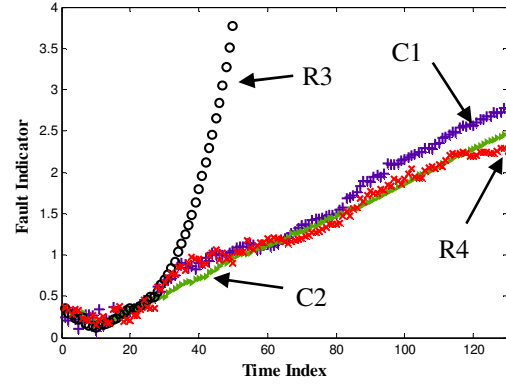


Figure 4. Trend exhibited by FI during the degradation of circuit components.

Thus, the stochastic model is a simple random walk that is employed to estimate the model parameters:

$$\left. \begin{aligned} a_{t_k}^{(1)} &= a_{t_{k-1}}^{(1)} + \omega_{1,k}; \quad \omega_{1,k} \sim N(0, \sigma_1) \\ b_{t_k}^{(1)} &= b_{t_{k-1}}^{(1)} + \omega_{2,k}; \quad \omega_{2,k} \sim N(0, \sigma_2) \\ c_{t_k}^{(1)} &= c_{t_{k-1}}^{(1)} + \omega_{3,k}; \quad \omega_{3,k} \sim N(0, \sigma_3) \\ a_{t_k}^{(2)} &= a_{t_{k-1}}^{(2)} + \omega_{4,k}; \quad \omega_{4,k} \sim N(0, \sigma_4) \\ b_{t_k}^{(2)} &= b_{t_{k-1}}^{(2)} + \omega_{5,k}; \quad \omega_{5,k} \sim N(0, \sigma_5) \\ c_{t_k}^{(3)} &= c_{t_{k-1}}^{(3)} + \omega_{6,k}; \quad \omega_{6,k} \sim N(0, \sigma_6) \end{aligned} \right\} (15)$$

$$\wp_{t_k} = a_{t_k}^{(1)} \left[ - \left( \frac{t_k - b_{t_k}^{(1)}}{c_{t_k}^{(1)}} \right)^2 \right] + a_{t_k}^{(2)} \left[ - \left( \frac{t_k - b_{t_k}^{(2)}}{c_{t_k}^{(2)}} \right)^2 \right] + \varepsilon_k; \quad \varepsilon_k \sim N(0, \sigma_\varepsilon)$$

where  $\wp_{t_k}$  is the measured value of the FI variable at time index  $t_k$ , and  $N(0, \sigma)$  is a Gaussian distribution with mean zero and standard deviation  $\sigma$ . Using the PFs, a  $p$ -step prediction can be made on the model parameters, with which the future values of the FI can be calculated. The RUP is calculated based on the time index at which the predicted FI value crosses the failure threshold. In our study, we found that by the time the FI value reaches a value of 2, the fault level for the most of the components has crossed 40%. Hence, in this work we have chosen a value of 2 for the FI as the failure threshold. This is just a qualitative threshold. In the future, a method to establish a failure threshold needs to be investigated. Also, the initial values of the model parameters (used from the curve fitting) and their



standard deviations are assumed to be known here for the sake of simplicity. In practice, an efficient method needs to be devised to choose these values for the model parameters.

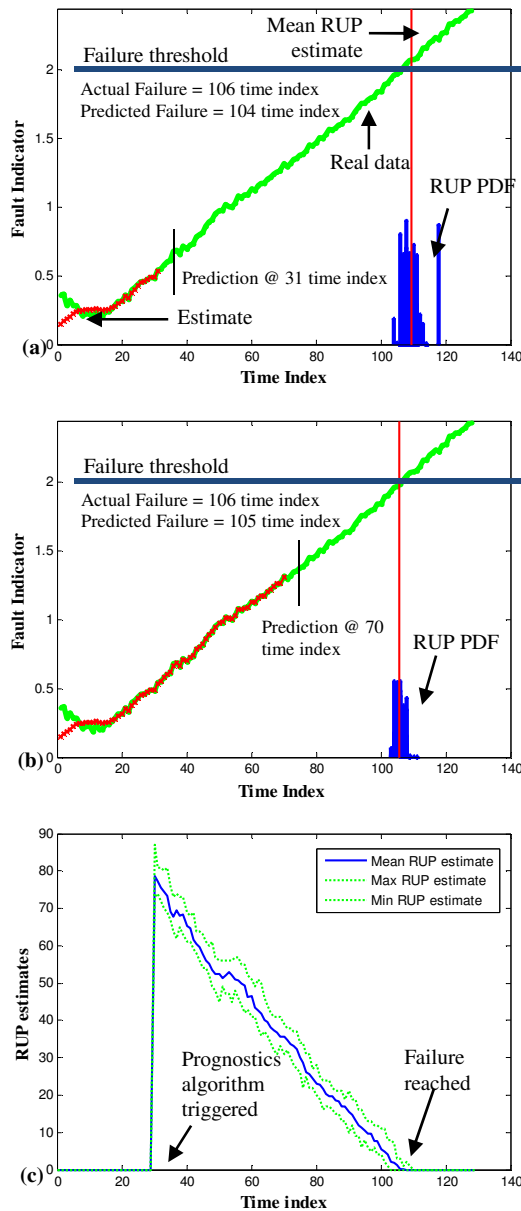


Figure 5. Prognostics results using particle filters for the Sallen-Key band-pass filter for fault progression in component C2 (a) Prediction result at time index 31, (b) Prediction result at time index 70, and (c) RUL estimation at every time index.

The results for fault progression in component C2 of the Sallen-Key band-pass filter circuit is shown in Figure 5. Figure 5(a) shows the RUP estimation at the time instant an anomaly was detected (i.e.,  $\rho > 0.5$ ). This occurred at time index 31. Thus, the data from the first 31 time indices alone are used to update the model. The estimated RUP is 73 and

thus the predicted end-of-life (EOL) is 104, and the actual failure occurs at the 106<sup>th</sup> time index. In Figure 5(b), the prognostics results performed at time index 70 are shown. Now, the predicted EOL is the 105<sup>th</sup> time index. Thus, the error in prediction is only one. Also, the prediction PDF becomes narrower as we get closer to the failure time indicating the improvement in prediction confidence. Figure 5(c) shows the RUP estimates at different time indices with 95% confidence bounds.

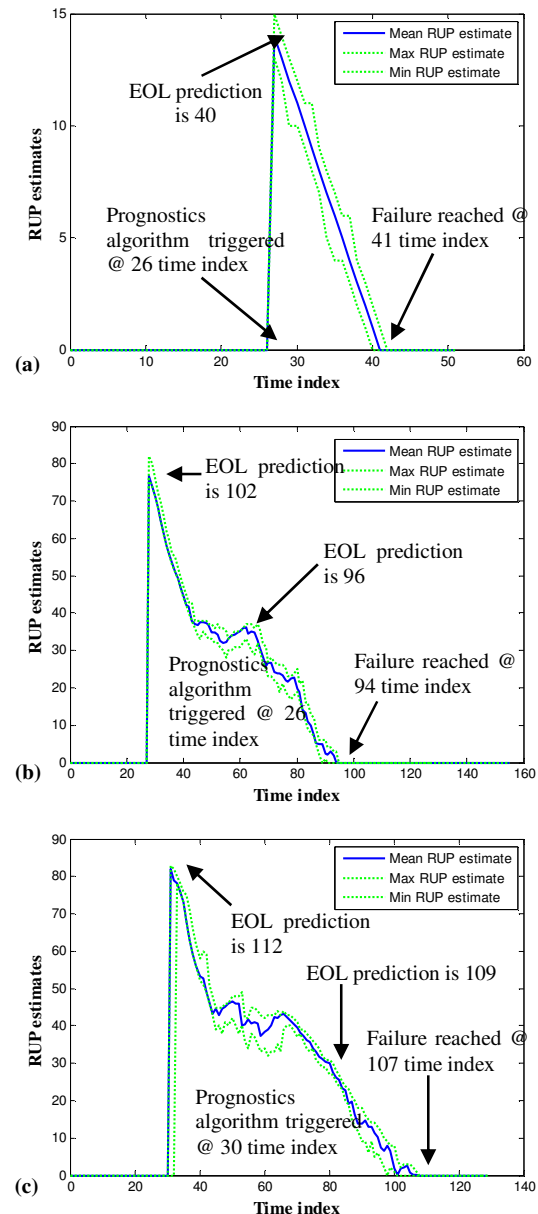


Figure 6. Prognostic results using particle filters for (a) Sallen-Key band-pass filter for fault progression in component R3, (b) biquad low pass filter for fault progression in component C1, and (c) biquad low pass filter for fault progression in component R4.

In order to show the applicability of the proposed approach for different components and circuits, the RUP estimation at different time indices for the component R3 in the Sallen-Key band-pass filter circuit, and the components C1 and R4 for the biquad low-pass filter circuit are shown in Fig. 6.

## 5. CONCLUSION

Existing test strategies for analog circuits are useful only after a failure has occurred. However, the nature of functions performed by industrial system demands test strategies that would enable the prevention of circuit failures. To address this concern, a prognostics framework for analog circuits is proposed in this paper.

The prognostics module includes a new fault indicator (FI) and a model adaptation scheme to track the evolution of FI. One feature of the FI is its ability to summarize the degradation level in any of the circuit's critical component. This allows RUP estimation to be performed without monitoring the individual components of the circuit leading to the reduction in resources needed for prognostics. Another feature of the FI is its compatibility to future changes in the extracted features. Thus, even if new features are introduced in the future, RUP predictions can be obtained from the circuit's response.

The model adaptive ability of the prognostic routine enables the real-time prediction of the circuit's RUP even under the presence of uncertainties that are introduced by component tolerances and time varying nature of the environment. Since this method involves a statistical approach for RUP prediction, the RUP estimate is given in the form of probability distributions, indicating the confidence levels in the prediction. Thus, preventive maintenance actions can be performed using the estimated RUP information for avoiding unexpected system failures due to faults in circuits.

## ACKNOWLEDGEMENT

The authors would like to thank the more than 100 companies and organizations that support research activities at the Prognostics and Health Management Group within the Center for Advanced Life Cycle Engineering at the University of Maryland annually.

## REFERENCES

- Aminian, F., & Aminian, M. (2002). Analog fault diagnosis of analog circuits using neural networks. *IEEE Trans. Instrum. Meas.*, vol. 51, no. 3, pp. 544-550.
- Brown, D., Kalgren, P., Byington, C., & Roemer, M. (2007). Electronic prognostics – a case study using global positioning system (GPS). *Microelectron. Reliab.*, vol. 47, pp. 1874-1881.
- Chen, Y-M., Wu, H-C., Chou, M-W., & Lee, K-Y. (2008). Online failure prediction of electrolytic capacitors for LC filter of switching-mode power converters. *IEEE Trans. Ind. Electron.*, vol. 55, no. 1, pp. 400-406.
- Kwon, D., Azarian, M., & Pecht, M. (2010). Prognostics of interconnect degradation using RF impedance monitoring and sequential probability ratio test. *International Journal of Performability Engineering*, vol. 6, no. 5, pp. 443-452.
- Orchard, M. & Vachtsevanos, G. (2009). A particle-filtering approach for on-line fault diagnosis. *Trans. Inst. Meas. Control*, vol. 31, no., pp. 221-246.
- Patil, N., Celaya, J., Das, D., Goebel, K., & Pecht, M. (2009). Precursor parameter identification for insulated gate bipolar transistor prognostics. *IEEE Trans. Reliab.*, vol. 58, no. 2, pp. 271-276.
- Pecht, M. & Jaai, R. (2010). A prognostics and health management roadmap for information and electronics-rich systems. *Microelectron. Reliab.*, vol. 50, no. 3, pp. 317-323.
- Soylemezoglu, A., Jagnnathan, S., & Saygin, C. (2010). Mahalanobis Taguchi system as a prognostic tool for rolling element bearing failures. *J. Manuf. Sci. Eng.*, vol. 132, no. 5, pp. 051014-1-051014-12.
- Taguchi, G. & Jugulum, R. (2002) *The Mahalanobis-Taguchi strategy*. Wiley-Interscience, New York.
- Wagner, K. & Williams, T. (1989). Design for testability of analog/digital networks. *IEEE Trans. Ind. Electron.*, vol. 36, no. 2, pp. 227-230.
- Xiao, Y. & He, Y. (2011). A novel approach for analog fault diagnosis based on neural networks and improved kernel PCA. *Neurocomputing*, vol. 74, pp. 1102-1115.

# An Efficient Deterministic Approach to Model-based Prediction Uncertainty Estimation

Matthew Daigle<sup>1</sup>, Abhinav Saxena<sup>2</sup>, and Kai Goebel<sup>1</sup>

<sup>1</sup> NASA Ames Research Center, Moffett Field, CA 94035, USA  
matthew.j.daigle@nasa.gov, kai.goebel@nasa.gov

<sup>2</sup> SGT, Inc., NASA Ames Research Center, Moffett Field, CA 94035, USA  
abhinav.saxena@nasa.gov

## ABSTRACT

Prognostics deals with the prediction of the end of life (EOL) of a system. EOL is a random variable, due to the presence of process noise and uncertainty in the future inputs to the system. Prognostics algorithms must account for this inherent uncertainty. In addition, these algorithms never know exactly the state of the system at the desired time of prediction, or the exact model describing the future evolution of the system, accumulating additional uncertainty into the predicted EOL. Prediction algorithms that do not account for these sources of uncertainty are misrepresenting the EOL and can lead to poor decisions based on their results. In this paper, we explore the impact of uncertainty in the prediction problem. We develop a general model-based prediction algorithm that incorporates these sources of uncertainty, and propose a novel approach to efficiently handle uncertainty in the future input trajectories of a system by using the unscented transform. Using this approach, we are not only able to reduce the computational load but also estimate the bounds of uncertainty in a deterministic manner, which can be useful to consider during decision-making. Using a lithium-ion battery as a case study, we perform several simulation-based experiments to explore these issues, and validate the overall approach using experimental data from a battery testbed.

## 1. INTRODUCTION

Prognostics deals with the prediction of the end of life (EOL) and remaining useful life (RUL) of components, subsystems, and systems. At its core, prognostics is a prediction problem. But, the future evolution of the system is a random process due to (i) process noise, and (ii) uncertainty in the future inputs to the system. In practice, these two sources of uncertainty cannot be avoided and thus EOL and RUL

are random variables. The prognostics algorithm itself introduces additional uncertainty because, in general, (i) it does not know exactly the state of the system at the time of prediction, (ii) it does not know exactly the description of the process noise, (iii) it does not know exactly the description of the future input uncertainty, and (iv) it does not know exactly the model of the future system behavior. All these sources of uncertainty contribute to the difficulty of the prognostics problem (Sankararaman, Ling, Shantz, & Mahadevan, 2011). While uncertainty cannot be eliminated from prognostics, an accurate assessment can be crucial in decision-making. Making decisions based on uncertain information requires characterizing the uncertainty itself to tune the risk level as needed in a particular application. In safety-critical systems it is of even higher importance, which is reflected in the fact that verification, validation, and certification protocols in the aerospace domain require provably deterministic and bounded systems.

Although the presence of prediction uncertainty is clearly a practical issue, only a few works have explored it. In (Sankararaman et al., 2011), the authors examine the various sources of uncertainty in fatigue crack growth prognostics and analyze their effects in an offline setting. In dealing with input uncertainty, future input trajectories are constructed as sequential blocks of constant-amplitude loading, and such trajectories are sampled in the prediction algorithm. In a similar approach applied to batteries in an unmanned aerial vehicle (UAV), in (Saha et al., 2012) the authors determine statistics of the battery loading for typical UAV maneuvers and analyze their effects in an offline setting. In dealing with input uncertainty, future input trajectories are constructed as sequential blocks of constant-amplitude loading, and such trajectories are sampled in the prediction algorithm. In a similar approach applied to batteries in an unmanned aerial vehicle (UAV), in (Saha et al., 2012) the authors determine statistics of the battery loading for typical UAV maneuvers based on past flight data, and construct future input trajectories as constrained sequences of flight maneuvers. In (Luo et al., 2008), constant loading is assumed for a vehicle suspension system, and predictions are made for a weighted set of three different loading values. The approach of (Edwards et al., 2010) also considers constant loading, and several uncertainty measures are defined that are then used within a framework for system life extension through actions that modify

Matthew Daigle et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

the system loading.

In this paper, we develop a general framework within the model-based prognostics paradigm for representing arbitrarily complex future input trajectories, and develop a general sample-based algorithm for predicting EOL and RUL that accounts for the sources of uncertainty in the prediction process. In particular, we introduce the unscented transform (Julier & Uhlmann, 1997), which predicts the mean and covariance of a random variable passed through a nonlinear function, as a method to efficiently sample from future input trajectories while still maintaining the statistics of the end result. This approach offers substantial computational savings as compared to a semi-exhaustive random sampling approach. Additionally, we show that since the UT allows sampling in a deterministic manner, and in this particular case where uncertainty in inputs is assumed to be uniformly distributed, we are able to bound the RUL predictions. Therefore, using the UT we realize a threefold benefit, (i) obtaining bounds for the prediction uncertainty, (ii) obtaining bounds in a deterministic manner, and (iii) keeping the statistical information intact with a considerably reduced computational burden as compared to traditional sampling approaches. Using a lithium-ion battery as a case study, we analyze the impact of uncertainty and various performance trade-offs of the prediction algorithm under several cases, and demonstrate and validate the approach with experimental data from a battery testbed.

The paper is organized as follows. Section 2 formulates the prognostics problem and describes the sources of uncertainty. Section 3 develops the general prediction algorithm and its different instantiations. Section 4 presents the battery case study and provides results using both experimental and simulated data. Section 5 concludes the paper.

## 2. PROGNOSTICS APPROACH

This section first formulates the prognostics problem. It then describes how uncertainty arises in prognostics, and examines the implications on prognostics algorithms. Finally, it provides an architecture for model-based prognostics.

### 2.1. Problem Formulation

The goal of prognostics is the prediction of the EOL and/or RUL of a system. We assume the system model may be generally defined as

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{f}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{v}(t)), \\ \mathbf{y}(t) &= \mathbf{h}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{n}(t)),\end{aligned}$$

where  $\mathbf{x}(t) \in \mathbb{R}^{n_x}$  is the state vector,  $\boldsymbol{\theta}(t) \in \mathbb{R}^{n_\theta}$  is the unknown parameter vector,  $\mathbf{u}(t) \in \mathbb{R}^{n_u}$  is the input vector,  $\mathbf{v}(t) \in \mathbb{R}^{n_v}$  is the process noise vector,  $\mathbf{f}$  is the state equation,  $\mathbf{y}(t) \in \mathbb{R}^{n_y}$  is the output vector,  $\mathbf{n}(t) \in \mathbb{R}^{n_n}$  is the

measurement noise vector, and  $\mathbf{h}$  is the output equation.<sup>1</sup>

Prognostics and health management is concerned with system performance that lies outside a given region of acceptable behavior. The desired performance is expressed through a set of  $n_c$  constraints,  $C_{EOL} = \{c_i\}_{i=1}^{n_c}$ , where  $c_i : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_u} \rightarrow \mathbb{B}$  maps a given point in the joint state-parameter space given the current inputs,  $(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t))$ , to the Boolean domain  $\mathbb{B} \triangleq [0, 1]$ , where  $c_i(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t)) = 1$  if the constraint is satisfied, and 0 otherwise.

These individual constraints may be combined into a single *threshold function*  $T_{EOL} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_u} \rightarrow \mathbb{B}$ , defined as

$$T_{EOL}(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t)) = \begin{cases} 1, & 0 \in \{c_i(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t))\}_{i=1}^{n_c} \\ 0, & \text{otherwise.} \end{cases}$$

$T_{EOL}$  evaluates to 1 when any of the constraints are violated. EOL is then defined as the earliest time point at which this occurs:

$$EOL(t_P) \triangleq \inf\{t \in \mathbb{R} : t \geq t_P \wedge T_{EOL}(\mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t)) = 1\},$$

RUL is expressed using EOL as

$$RUL(t_P) \triangleq EOL(t_P) - t_P.$$

### 2.2. Prediction Uncertainty

The above definitions of EOL and RUL are for their exact values, i.e., the system takes some path *out of many possible paths* through the state space until EOL. The actual path the system will take cannot be known in advance because the system evolution is a random process, therefore, EOL and RUL at any prediction time  $t_P < EOL$ , are actually random variables. System evolution is random due to the process noise  $\mathbf{v}(t)$  and because  $\mathbf{u}(t)$  for  $t > t_P$  is never known exactly. Since EOL is a function of  $(\mathbf{x}(t_P), \boldsymbol{\theta}(t_P))$  and  $\mathbf{u}(t)$ , which are all random variables, EOL (and RUL) must also be a random variable. This uncertainty is inherent to the system itself and cannot be avoided. Note that as  $t$  approaches EOL, the variability in the actual EOL will naturally reduce, simply because  $EOL - t$  becomes smaller.

The goal of a prognostics algorithm, then, is to compute the true distribution of the EOL and RUL. A decision that is made based on a misrepresentation of this true distribution could have a significant impact, especially if the true variability is underestimated. It is therefore critical that a prognostics algorithm comes as close to this true distribution as possible.

<sup>1</sup>Here, we use bold typeface to denote vectors, and use  $n_a$  to denote the length of a vector  $\mathbf{a}$ .

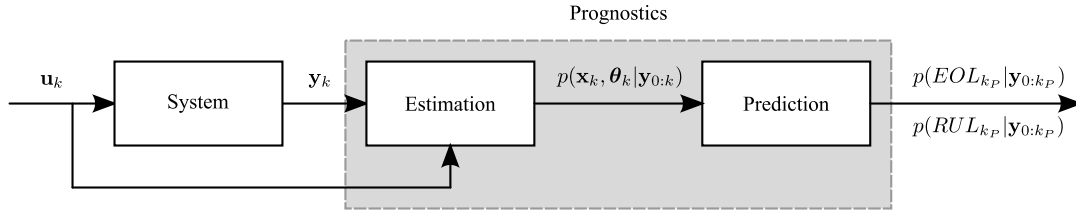


Figure 1. Model-based prognostics architecture.

However, additional uncertainty is also introduced by the prognostics algorithm itself. In order to make a prediction, the state of the system at the time of prediction must be known. At best, only a probability distribution can be estimated since (i) the system state may not be directly measured and, (ii) even if it is, there is sensor noise, (iii) the initial state of the system is not known exactly, (iv) the system model is not known exactly, and (v) there is process noise. Even if the system state is known exactly, uncertainty is introduced in the prediction process since, in general, (i) the model used for prediction is not known exactly, (ii) the correct representation of process noise is not known exactly, and (iii) the correct representation of the space of possible future input trajectories is not known exactly. Due to these additional sources of uncertainty inherent to the prognostics algorithm, the uncertainty in the predicted EOL/RUL will nominally be larger than the true variability in EOL/RUL.

The uncertainty inherent to the system cannot be eliminated, and neither can the uncertainty inherent to the algorithm. However, the uncertainty associated with the algorithm can be limited by using the best known model including the best known representation of the process noise, and by constraining the space of possible future input trajectories as much as possible and representing the associated probability distribution as accurately as possible. The potential trajectories of  $\mathbf{u}(t)$  for  $t \geq t_P$  depend on the system being monitored and can, in general, take on any number of arbitrary signals. We denote the space of possible future input trajectories as  $\mathcal{U}_{t_P}$ . A single trajectory in the set  $\mathcal{U}_{t_P}$  is denoted as  $\mathbf{U}_{t_P}$ , and defines the values of  $\mathbf{u}(t)$  for all  $t \geq t_P$ . Each possible trajectory has a certain probability of occurring in the real system, and so this is defined by a probability distribution  $p(\mathcal{U}_{t_P})$ . In practice, it is entirely possible that this exact distribution is unknown and must be approximated.

### 2.3. Prognostics Architecture

To predict EOL/RUL, first an initial state to use for the prediction must be determined. In the model-based paradigm, this is referred to as the *estimation* problem, and requires determining a joint state-parameter estimate  $p(\mathbf{x}(t), \boldsymbol{\theta}(t) | \mathbf{y}_{0:t})$  based on the history of observations up to time  $t$ ,  $\mathbf{y}_{0:t}$ . The *prediction* problem is to determine, using this estimate, EOL and RUL probability distributions,  $p(EOL_{t_P} | \mathbf{y}_{0:t_P})$

and  $p(RUL_{t_P} | \mathbf{y}_{0:t_P})$ .

The prognostics architecture is shown in Fig. 1 (Daigle & Goebel, 2011). In discrete time  $k$ , the system is provided with inputs  $\mathbf{u}_k$  and provides measured outputs  $\mathbf{y}_k$ . The estimation module uses this information, along with the system model, to compute an estimate  $p(\mathbf{x}_k, \boldsymbol{\theta}_k | \mathbf{y}_{0:k})$ .<sup>2</sup> The prediction module uses the joint state-parameter distribution and the system model, along with hypothesized future inputs, to compute EOL and RUL as probability distributions  $p(EOL_{k_P} | \mathbf{y}_{0:k_P})$  and  $p(RUL_{k_P} | \mathbf{y}_{0:k_P})$  at given prediction times  $k_P$ .

### 3. PREDICTION

Prediction is initiated at a given time of prediction  $k_P$  using the current joint state-parameter estimate,  $p(\mathbf{x}_{k_P}, \boldsymbol{\theta}_{k_P} | \mathbf{y}_{0:k_P})$ . Approaches to determine this estimate are reviewed in (Daigle et al., 2012) and will not be described here. The goal is to compute  $p(EOL_{k_P} | \mathbf{y}_{0:k_P})$  and  $p(RUL_{k_P} | \mathbf{y}_{0:k_P})$ . The representation of  $p(\mathbf{x}_{k_P}, \boldsymbol{\theta}_{k_P} | \mathbf{y}_{0:k_P})$  is determined by the algorithm used for the estimation step. In any case, here, we assume it is given as a set of weighted samples  $\{(\mathbf{x}_{k_P}^{i_x}, \boldsymbol{\theta}_{k_P}^{i_x}), w_{x,k_P}^{i_x}\}$ . In the case of the unscented Kalman filter (Julier & Uhlmann, 1997, 2004) and the particle filter (Arulampalam et al., 2002), the distribution is provided in this format, otherwise, the provided distribution can be sampled.

Here, we follow a sample-based approach (as opposed to analytical methods) to the prediction problem to incorporate the uncertainty (Sankararaman et al., 2011), in which each sample is simulated to EOL. The approach is shown as Algorithm 1. For each of the  $N_x$  samples of the state-parameter distribution, we sample from the input distribution,  $\mathcal{U}_{k_P}$ ,  $N_u$  future input trajectories (where a single trajectory  $\mathbf{U}_{k_P} \triangleq \{\mathbf{u}_{k_P}, \mathbf{u}_{k_P+1}, \dots\}$ ) with weights  $w_{u,k_P}$ , and for each of these trajectories, simulate  $N_v$  trajectories with process noise (lines 10–14). At the end, we obtain a weighted set of EOL predictions for each of these simulations, totaling  $N_x \times N_u \times N_v$ , i.e.,  $\{EOL_{k_P}^j, w_{k_P}^j\}_{j=1}^{N_x \times N_u \times N_v}$  (lines 15 and 16).

In the algorithm, line 11 samples the next state from the prior probability distribution. Effectively, this is implemented by

<sup>2</sup>Estimation does not need to be performed by the prognoser if it is provided by some other module, such as a diagnoser (Roychoudhury & Daigle, 2011).

---

**Algorithm 1** EOL Prediction with Uncertainty
 

---

```

1: Inputs:  $\{(\mathbf{x}_{k_P}^{i_x}, \boldsymbol{\theta}_{k_P}^{i_x}), w_{x,k_P}^{i_x}\}_{i_x=1}^{N_x}, N_u, N_v$ 
2: Outputs:  $\{EOL_{k_P}^j, w_{k_P}^j\}_{j=1}^{N_x \times N_u \times N_v}$ 
3:  $\{\mathbf{U}_{k_P}^{i_u}, w_{u,k_P}^{i_u}\}_{i_u=1}^{N_u} \sim p(\mathcal{U}_{k_P})$ 
4: for  $i_x = 1$  to  $N_x$  do
5:   for  $i_u = 1$  to  $N_u$  do
6:     for  $i_v = 1$  to  $N_v$  do
7:        $j \leftarrow (i_x, i_u, i_v)$ 
8:        $k \leftarrow k_P$ 
9:        $\mathbf{x}_k^j \leftarrow \mathbf{x}_{k_P}^j$ 
10:      while  $T_{EOL}(\mathbf{x}_k^j, \boldsymbol{\theta}_k^j, \mathbf{U}_{k_P}^{i_u}(k)) = 0$  do
11:         $\mathbf{x}_{k+1}^j \sim p(\mathbf{x}_{k+1}^j | \mathbf{x}_k^j, \boldsymbol{\theta}_{k_P}^j, \mathbf{U}_{k_P}^j(k))$ 
12:         $k \leftarrow k + 1$ 
13:         $\mathbf{x}_k^j \leftarrow \mathbf{x}_{k+1}^j$ 
14:      end while
15:       $EOL_{k_P}^j \leftarrow k$ 
16:       $w_{k_P}^j \leftarrow w_{x,k_P}^{i_x} \times w_{u,k_P}^{i_u} / N_v$ 
17:    end for
18:  end for
19: end for
    
```

---

sampling the process noise and executing the state equation with that process noise. Each of these trajectories individually are set to have equal weight ( $1/N_v$ ), and a statistically meaningful result can only be obtained by sampling a sufficient number of times. The longer the time to EOL, the more of an effect process noise will have and the more samples may be necessary to accurately capture the statistics. If desired, process noise can be set to zero.

Line 3 in the algorithm samples the future input trajectories. What this space looks like is highly dependent on the underlying application, and how to sample this space depends on what the space looks like. It is up to the modeler to define this space and its probability distribution. There are a few simple approaches to take that are generally applicable. One way to handle this is to define a family of parameterized functions that define  $\mathbf{u}(t)$  for all  $t \geq t_P$ . For example, let  $\mathbf{u}(t) = p$ , where  $p$  is an unknown value drawn from a known (or assumed) distribution. To sample an input trajectory one needs only to sample a value for  $p$ . Or, let  $\mathbf{u}(t) = p_1 t + p_2 t^2$ . Here, an input trajectory is sampled by sampling values for  $p_1$  and  $p_2$ . More complicated functions may also be defined, and as long as they are parameterized then it is easy to sample such functions. Another general approach is to define the input as a set of blocks where within each block the input is constant, such as in (Sankaraman et al., 2011; Saha et al., 2012). One must then sample how long the next block will last and at what magnitude it will be.

The computational complexity of the algorithm is mainly a function of the number of unique samples ( $N_x \times N_u \times N_v$ ). Secondary to this is how long each sample takes to simulate to EOL. Samples with higher rates of damage progression (e.g., due to increased loading) will simulate faster than those with lower rates of progression.

The general algorithm presented here can be instantiated in different ways, depending on how the future input trajectories are sampled (line 3). In the following subsections we describe different sampling methods.

### 3.1. Exhaustive Sampling

If the input trajectory space  $\mathcal{U}_{k_P}$  is finite, then it is possible to do predictions over the entire space. This is limited by the imposed computational requirements, because even if the space is finite, the number of discrete elements may be too large. In this case the sampling would be deterministic, so repeated executions would always get the same results.

### 3.2. Random Sampling

If the input trajectory space is infinitely large or finite but too large for exhaustive sampling, then random sampling may be used to obtain a number of sufficient samples. Because the process is stochastic, the results will be nondeterministic, which could have a significant impact on performance if too few samples are drawn. Further, repeated executions would obtain different results, which could make validation of the algorithm difficult. However, it is the most generally applicable approach.

### 3.3. Sampling with the Unscented Transform

As an alternative to random sampling, nonexhaustive deterministic sampling can also be performed. One method is to use the unscented transform (UT). The UT takes a random variable  $\mathbf{x} \in \mathbb{R}^{n_x}$ , with mean  $\bar{\mathbf{x}}$  and covariance  $\mathbf{P}_{xx}$ , that is related to a second random variable  $\mathbf{y} \in \mathbb{R}^{n_y}$  by some function  $\mathbf{y} = \mathbf{g}(\mathbf{x})$ , and computes the mean  $\bar{\mathbf{y}}$  and covariance  $\mathbf{P}_{yy}$  with high accuracy using a minimal set of deterministically selected weighted samples, called *sigma points* (Julier & Uhlmann, 1997). The number of sigma points is only linear in the dimension of the random variable, and so the statistics of the transformed random variable, i.e., mean and covariance, can be computed much more efficiently than by random sampling.<sup>3</sup>

Here,  $\mathcal{X}^i$  denotes the  $i$ th sigma point from  $\mathbf{x}$  and  $w^i$  denotes its weight. The sigma points are always chosen such that the mean and covariance match those of the original distribution,  $\bar{\mathbf{x}}$  and  $\mathbf{P}_{xx}$ . Each sigma point is passed through  $\mathbf{g}$  to obtain new sigma points  $\mathcal{Y}$ , i.e.,

$$\mathcal{Y}^i = \mathbf{g}(\mathcal{X}^i)$$

<sup>3</sup>Versions of the unscented transform also exist that compute also higher-order moments like skew (Julier, 1998).

with mean and covariance calculated as

$$\bar{\mathbf{y}} = \sum_i w^i \mathbf{y}^i$$

$$\mathbf{P}_{yy} = \sum_i w^i (\mathbf{y}^i - \bar{\mathbf{y}})(\mathbf{y}^i - \bar{\mathbf{y}})^T.$$

In this paper, we use the symmetric unscented transform, in which  $2n_x + 1$  sigma points are symmetrically selected about the mean according to (Julier & Uhlmann, 2004):

$$w^i = \begin{cases} \frac{\kappa}{(n_x + \kappa)}, & i = 0 \\ \frac{1}{2(n_x + \kappa)}, & i = 1, \dots, 2n_x \end{cases}$$

$$\mathbf{x}^i = \begin{cases} \bar{\mathbf{x}}, & i = 0 \\ \bar{\mathbf{x}} + \left( \sqrt{(n_x + \kappa) \mathbf{P}_{xx}} \right)^i, & i = 1, \dots, n_x \\ \bar{\mathbf{x}} - \left( \sqrt{(n_x + \kappa) \mathbf{P}_{xx}} \right)^i, & i = n_x + 1, \dots, 2n_x, \end{cases}$$

where  $\left( \sqrt{(n_x + \kappa) \mathbf{P}_{xx}} \right)^i$  refers to the  $i$ th column of the matrix square root of  $(n_x + \kappa) \mathbf{P}_{xx}$ . Note that the required number of samples is only linear in the size of the state space. Here,  $\kappa$  is a free parameter that can be used to tune higher order moments of the distribution. If  $\mathbf{x}$  is assumed Gaussian, then selecting  $\kappa = 3 - n_x$  is recommended (Julier & Uhlmann, 1997). Note that with the UT, weights may be negative, and are not to be directly interpreted as probabilities.

If we consider the random variable in this case to be a representation of our input space, then the UT can be used to sample the space of input trajectories. Here, the simulation to EOL is the nonlinear transformation, as in (Daigle & Goebel, 2010). A representation of the input space for this framework is required. If the future input trajectories are defined by parameterized functions, where the function parameters are themselves sampled from some distribution, then the input space is defined by these parameters and the UT can be used to sample from this parameter space. The number of samples would be linear in the number of parameters (as this defines the state space for the UT). By using the UT to sample this parameter space, in effect we obtain representative samples of the input trajectory space.

If the input space cannot be transformed to a representation amenable to the UT, then this approach cannot be used. Such cases, however, may not appear often in practice since this would imply that it is difficult for the modeler to define the input space in the first place. That is, an easy and practical way to define the input trajectory space is by sampling a finite set of parameters that define a particular input trajectory.

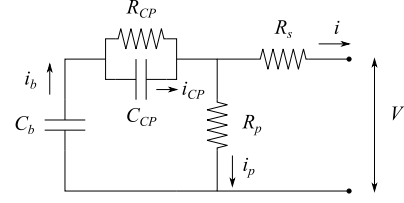


Figure 2. Battery equivalent circuit.

## 4. CASE STUDY

We select a lithium-ion battery as a case study on which to demonstrate and validate our approach. We first present the battery model. We then apply the approach to experimental data and demonstrate the impact of prognostics uncertainty. We then present a number of simulation experiments to more systematically explore these issues.

### 4.1. Modeling

The battery model is based on an electrical circuit equivalent shown in Fig. 2, similar to models presented in (Chen & Rincon-Mora, 2006; Barsali & Ceraolo, 2002; Ceraolo, 2000). The large capacitance  $C_b$  holds the charge  $q_b$  of the battery. The  $R_{CP}$ - $C_{CP}$  pair captures the major nonlinear voltage drop due to concentration polarization,  $R_s$  captures the so-called I-R drop, and  $R_p$  models the parasitic resistance that accounts for self-discharge. This simple battery model is enough to capture the major dynamics of the battery, but ignores temperature effects and other minor battery processes.

The state-of-charge,  $SOC$ , is computed as

$$SOC = 1 - \frac{q_{max} - q_b}{C_{max}},$$

where  $q_b$  is the current charge in the battery (related to  $C_b$ ),  $q_{max}$  is the maximum possible charge, and  $C_{max}$  is the maximum possible battery capacity (i.e., nominally, its rated capacity). The concentration polarization resistance is a nonlinear function of  $SOC$ :

$$R_{CP} = R_{CP0} + R_{CP1} \exp R_{CP2}(1 - SOC),$$

where  $R_{CP0}$ ,  $R_{CP1}$ , and  $R_{CP2}$  are empirical parameters. The resistance, and, hence, the voltage drop, increases exponentially as  $SOC$  decreases (Saha et al., 2012).

Voltage drops across the individual circuit elements are given by

$$V_b = q_b / C_b$$

$$V_{CP} = q_{CP} / C_{CP}$$

$$V_p = V_b - V_{CP},$$

where  $q_{CP}$  is the charge associated with the capacitance

Parameter	Value
$C_b$	9844
$R_s$	0.143014
$R_p$	500
$C_{CP}$	70.3767
$R_{CP0}$	0.019829
$R_{CP1}$	$3.68606 \times 10^{-14}$
$R_{CP2}$	31.9213
$q_{max}$	41400
$C_{max}$	6900

Table 1. Battery Model Parameters

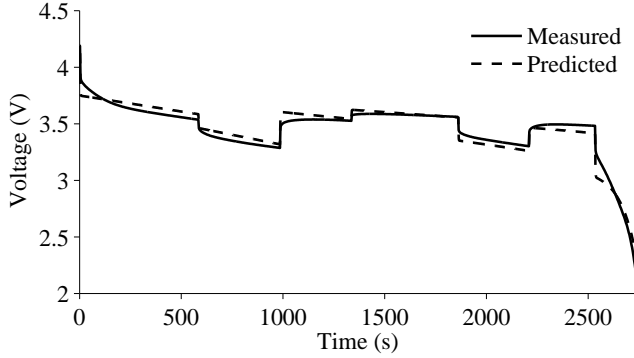


Figure 3. Battery model comparison with experimental data.

$C_{CP}$ . The terminal voltage of the battery is

$$V = V_b - V_{CP} - R_s i,$$

where  $i$  is the battery current at the terminals. Currents associated with the individual circuit elements are given by

$$\begin{aligned} i_p &= V_p / R_p \\ i_b &= i_p + i \\ i_{CP} &= i_b - V_{CP} / R_{CP}. \end{aligned}$$

The charges are then governed by

$$\begin{aligned} \dot{q}_b &= -i_b \\ \dot{q}_{CP} &= i_{CP} \end{aligned}$$

We are interested in predicting end-of-discharge as defined by a voltage threshold  $V_{EOD}$ . So,  $C_{EOL}$  consists of only one constraint:

$$c_1 : V > V_{EOD}.$$

From experimental data of battery discharges, we have identified the parameters of the battery model through model fitting, and their values are shown in Table 1. A comparison of measured and predicted behavior for known current inputs is shown in Fig. 3. The associated inputs are shown in Fig. 4. Clearly, the model is not perfect, so prognosis will have to account for the model uncertainty.

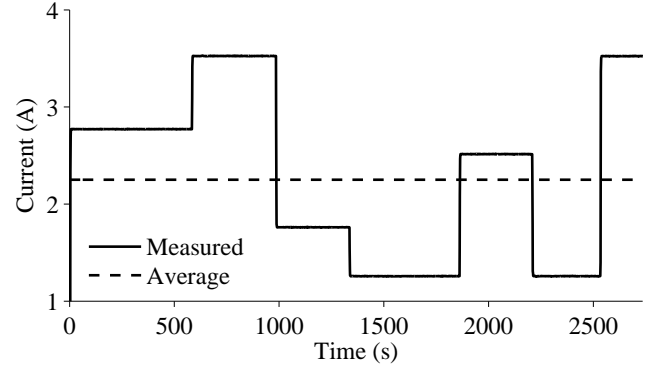


Figure 4. Battery current inputs.

## 4.2. Experimental Results

To demonstrate and validate the approach, we apply our prognostics algorithms to the experimental data given in Fig. 3. For state estimation, we use the unscented Kalman filter (UKF) (Julier & Uhlmann, 1997; Daigle et al., 2012). Here, we use the UKF since the model is nonlinear and generally performs better (and is easier to apply) than the extended Kalman filter (Julier & Uhlmann, 1997). The model uncertainty is captured through process noise, represented using normal distributions with zero mean. Sensor noise is generally very small but is also assumed to be zero-mean Gaussian.

The battery current is viewed as an input to the model, and its future values must be hypothesized. First, we assume that the future inputs are known exactly (we use the exact load profile available from the experimental data shown in Fig. 4), and that there is no process noise. The RUL predictions versus time are shown in Fig. 5. The predictions are shown against the true RUL (denoted as  $RUL^*$ ) along with an accuracy cone defined by  $\alpha = 0.15$ . Predictions are made every 100 s. Here, we see that the predictions are quite accurate and remain within 15% of the true RUL until about 2500 s. The UKF can partially correct for the model uncertainty, but towards the end of the discharge the error cannot be fully corrected and the relative accuracy is reduced, due to the high sensitivity to the final voltage drop. The only uncertainty captured by these predictions is that in the state estimate, which is very small (so is not visible in the figure). Clearly, these predictions do not capture the true uncertainty so would be incorrect to use for decision-making.

Process noise must be correctly represented in order to yield usable results. Fig. 6 shows the RUL predictions versus time in this case, using  $N_v = 100$ . Now, we see that the uncertainty represented in the prediction covers the decrease in accuracy observed towards the end of the discharge, i.e., the true RUL is now contained within the uncertainty bounds.

If the future inputs are not known, then some assumption must be made about what they look like. First, we assume



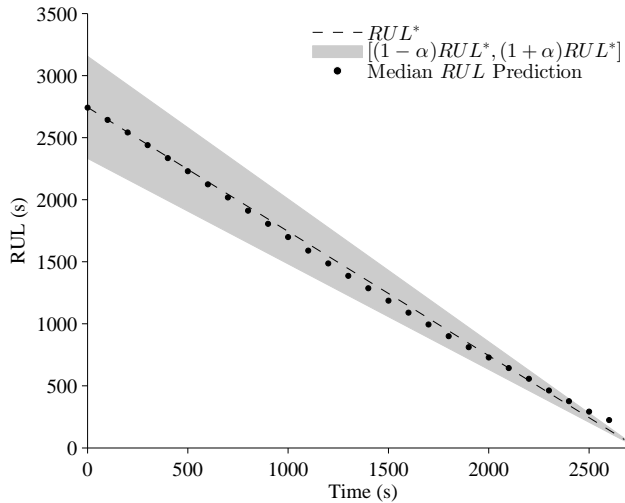


Figure 5. Prediction performance assuming known future input trajectory and no process noise with  $\alpha = 0.15$ .

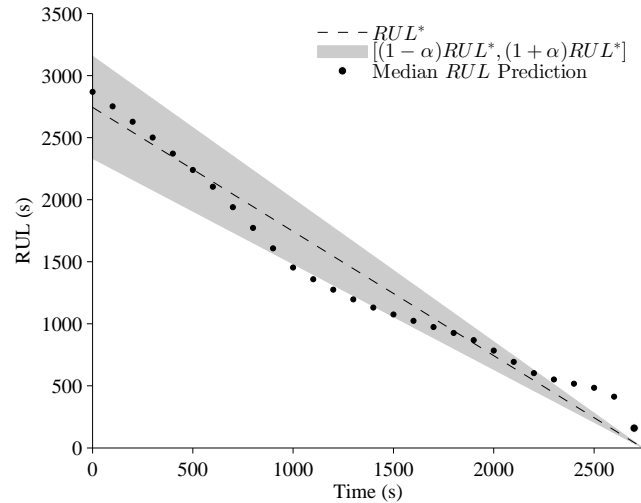


Figure 7. Prediction performance assuming average future input trajectory and no process noise with  $\alpha = 0.15$ .

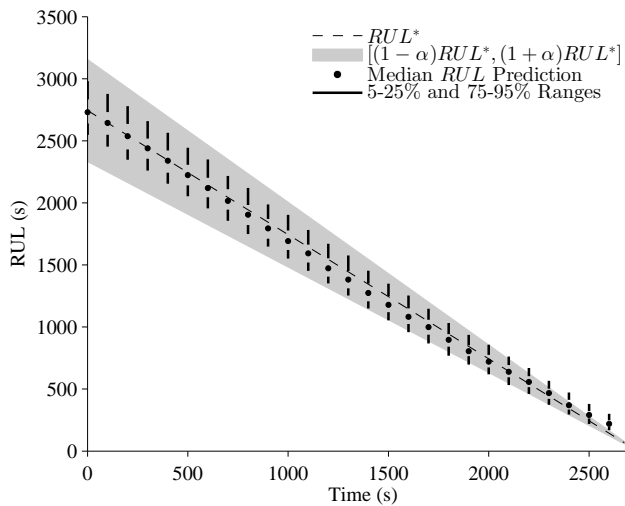


Figure 6. Prediction performance assuming known future input trajectory and 100 process noise samples with  $\alpha = 0.15$ .

just a constant current of 2.25 A, which is the average current drawn during the actual experiment (shown as the dashed line in Fig. 4). Here, we assume there is no process noise. The results are shown in Fig. 7. Clearly, assuming the average current results in a significant performance degradation. Around 1000 s, the average current until discharge is lower than hypothesized, so RUL is underestimated. Around 2500 s, the average current until discharge is higher than hypothesized, so RUL is overestimated. It is apparent that the RUL prediction is very sensitive to the uncertainty in the future inputs. In this case, the predictions are much more sensitive to input uncertainty than model uncertainty, since the model itself is more sensitive to changes in input than to the added process noise.

Instead of assuming a single possible future input trajectory, we now consider multiple trajectories. At each prediction step, we assume the future current is drawn from a uniform distribution between 1 and 4 A and remains constant for the remainder of the discharge. Looking at Fig. 4, the current does not remain constant, but serves as a reasonable assumption for prediction purposes, i.e., rather than assuming variable discharge currents we assume constant discharge currents within the range of possible currents, which is much easier to sample from and still captures the best- and worst-case inputs. Fig. 8 shows the results using 10 samples and Fig. 9 shows the results using 100 samples. The uncertainty in the RUL predictions now is much more accurately represented. Clearly, the more samples used, the smoother the predictions and the better the description of the uncertainty. It is also clear here that the uncertainty in the future inputs causes about an order of magnitude more spread in the RUL predictions than the uncertainty associated with the process noise, comparing the figures (specifically, relative median absolute deviation averaged 6.2% in Fig. 6 and 27.9% in Fig. 8). Therefore, it would be acceptable to drop the process noise, which would save also on the required amount of computation.

Further improvements in computation can be achieved by using the unscented transform to select input trajectories. In this case, the future input trajectories are parameterized by a single number, representing the future current draw, taken from a uniform distribution. Applying the UT to this case yields only three future input trajectories that need to be simulated. There is no guidance to choosing a value for  $\kappa$  when the distribution is uniform, so we use the suggested value of  $\kappa$  for when the distribution is Gaussian, which, for a one-dimensional input

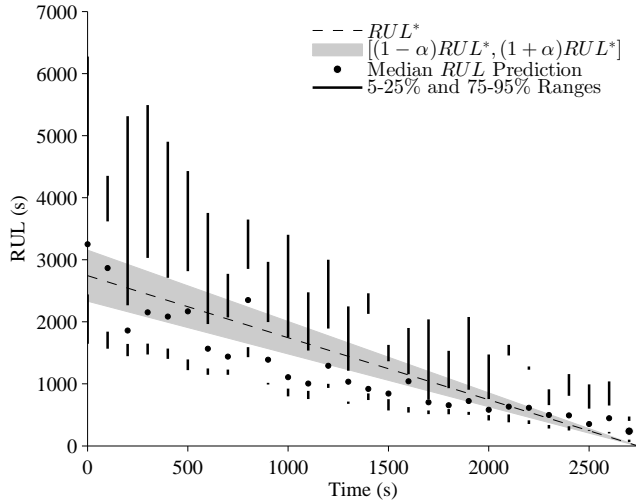


Figure 8. Prediction performance with 10 future input trajectories drawn from a uniform distribution and assuming no process noise with  $\alpha = 0.15$ .

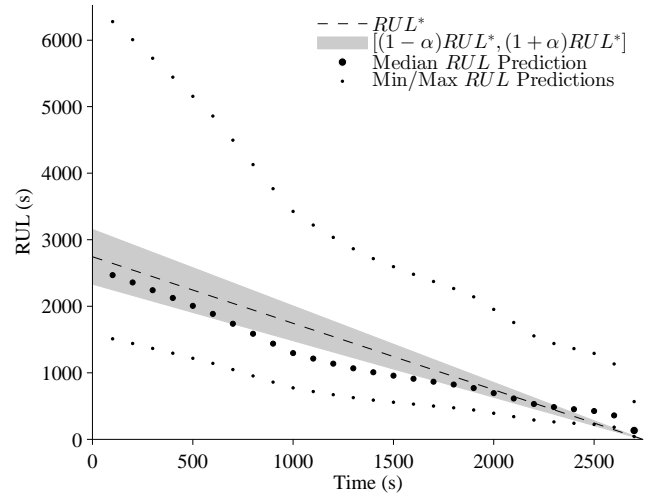


Figure 10. Prediction performance with future input trajectories drawn from a uniform distribution using the unscented transform and assuming no process noise with  $\alpha = 0.15$ .

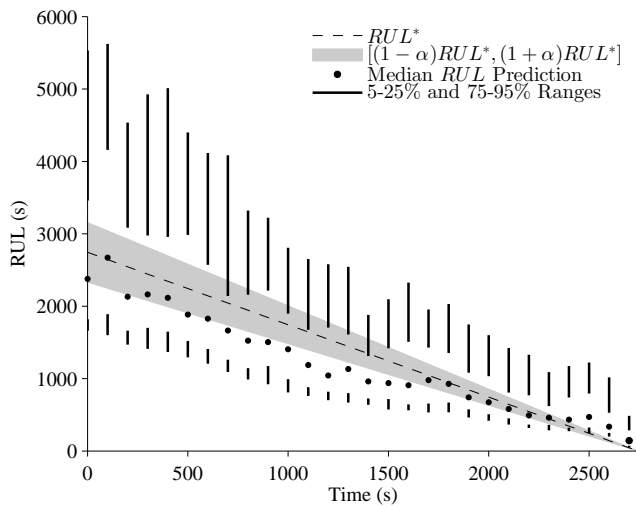


Figure 9. Prediction performance with 100 future input trajectories drawn from a uniform distribution and assuming no process noise with  $\alpha = 0.15$ .

space, is  $\kappa = 2.4$ . With  $\kappa = 2$ , the UT happens to choose the three points as the mean of the distribution and its two endpoints, thus naturally capturing the input bounds. The results are shown in Fig. 10. Comparing to the case where 100 random input trajectories were generated, using the UT we are able to capture approximately the same distribution with only a fraction of the computational effort. The figure shows the results from the three sigma points directly, but the distribution can be reconstructed from this minimal set of samples. We will show in the following subsection that the UT is able to do this accurately. In this case the UT provides both the

<sup>4</sup>A smaller value of  $\kappa$  will bring the sigma points closer together, and a larger value spreads them out.

RUL distribution and its bounds deterministically. Note that for a bounded distribution one may always choose the endpoints to determine the RUL bounds, however, in this case the UT does this automatically with the added benefit of being able to reconstruct the RUL distribution from those two points and the mean.

### 4.3. Simulation Results

For a more careful analysis, we ran a set of comprehensive simulation experiments. In these experiments, by using a simulation model, we eliminate several sources of uncertainty: (i) the state at the time of prediction is assumed to be known exactly, (ii) the system model is known exactly, (iii) the process noise distribution is known exactly, and (iv) the future input trajectory distribution is known exactly. This focuses the resulting uncertainty to only that associated with the process noise and the future input trajectories. In each experiment, the true input trajectory is sampled from the known distribution, while the algorithm knows only the distribution.

To analyze prognostics performance we use the relative accuracy (RA) metric to characterize the accuracy (Saxena, Celaya, Saha, Saha, & Goebel, 2010). For RA we use the median as the measure of central tendency since the RUL distributions are skewed. For spread, we use relative median absolute deviation (RMAD). For each experiment we perform 100 iterations, and average the results over these iterations. We compute also a computation time metric  $T_{cpu}$  which is computed as the fraction of computation time taken for a prediction  $t_{cpu}$  over the true RUL,  $RUL^*$ , i.e.,  $T_{cpu} = t_{cpu}/RUL^*$ .

Results with process noise but without input uncertainty are shown in Table 2. In this case process noise has little effect

$N_v$	$RA_{RUL}$	$RMAD_{RUL}$	$T_{cpu}$
0	99.42	0.00	$5.91 \times 10^{-4}$
10	99.28	1.34	$6.75 \times 10^{-4}$
50	99.15	1.47	$6.97 \times 10^{-4}$
100	99.02	1.39	$7.34 \times 10^{-4}$

Table 2. Prognostics Performance with different values of  $N_v$  and no input uncertainty.

on accuracy, so even when ignoring process noise (indicated by  $N_v = 0$  in the table) accuracy is pretty good, although the RMAD is 0 so the true spread is being underrepresented by the RUL prediction. Even with only 10 samples the prediction spread is quite close to the case with 100 samples (with a difference of only 0.26 in RA and 0.05 in RMAD), so only having a about 10 samples is acceptable for this level of process noise. As process noise increases, more samples will be needed to properly cover that space.

We now focus on the effect of the uncertainty in future inputs, so eliminate process noise from the simulation. The actual input current is drawn from a uniform distribution between 0.75 and 2.00 A. If the prediction algorithm always selects the mean, 1.375 A, as the future input current, then the relative accuracy of the first prediction point varies from about 52-97%. RA is high when the actual current is close to the mean but low when far from the mean. Because no uncertainty is taken into account, when the actual current is far from the mean not only will the accuracy be low, but the predictions will be of high precision and therefore be presented as predictions with high confidence, which would be incorrect and lead to poor decision-making.

If we instead sample randomly from the input distribution, the relative accuracy on average does not really change from when assuming only the mean input. This is because RA is computed based on the median RUL prediction, and when the input distribution is sampled enough, then we will get the correct mean of that distribution, which, because the distribution is symmetric, will correspond to the median RUL prediction. The key difference, however, is that now the RUL spread is more accurately represented by the predicted RUL distribution. The true RMAD is around 27%. Table 3 shows the prediction performance for different values of  $N_u$ . RA is about the same and the prediction spread, as computed by RMAD, approaches the true spread with around 25 samples (with about 7 times the computation as when predicting with only one sample). Using the UT (in which only 3 samples are needed), the performance is similar with only about 2.6 times as much computation needed compared to using only a single sample. Using  $\kappa = 2$  for the UT, it chooses the mean and the endpoints of the assumed uniform distribution, so naturally provides the median and best- and worst-case RULs. Using the weights of the sigma points we can reconstruct the distribution it represents, which is not possible when just choosing these values in an ad hoc manner.

$N_u$	$RA_{RUL}$	$RMAD_{RUL}$	$T_{cpu}$
1	62.98	0.00	$6.31 \times 10^{-4}$
10	74.57	25.96	$2.36 \times 10^{-3}$
25	75.90	27.07	$5.39 \times 10^{-3}$
50	78.54	27.50	$9.61 \times 10^{-3}$
75	74.31	27.55	$1.48 \times 10^{-2}$
100	76.71	27.61	$2.08 \times 10^{-2}$
UT	76.77	26.72	$1.61 \times 10^{-3}$

Table 3. Prognostics Performance with different values of  $N_u$  and no process noise.

## 5. CONCLUSIONS

In this paper, we analyzed the sources of uncertainty in prognostics and developed a general model-based prediction algorithm that incorporates this uncertainty to provide EOL and RUL results that correctly capture the true uncertainty in EOL and RUL. We also introduced the use of the unscented transform for efficiently sampling from the space of possible future input trajectories, which can achieve the same results as random sampling but at a fraction of the computational effort (see Table 3).

We applied the approach to both real and simulated lithium-ion battery data, where end-of-discharge was predicted. Here it was demonstrated that it is important to realize that for any source of uncertainty that is ignored in the prediction, even though in one particular case the result may be accurate, the actual uncertainty is underrepresented and so is not capturing the right information needed for decision-making. If the uncertainty is accounted for, on average the accuracy will be the same as if just average performance is assumed, but the spread will be correctly represented and the actual system behavior will fall within the predicted spread. Since the actual path the system will take is uncertain, it is best practice to capture the uncertainty as accurately as possible. It is important also not to overestimate the uncertainty. For example, when defining the space of possible future input trajectories it is important to constrain these trajectories as much as possible.

In this paper, we have shown that it is possible, by using the unscented transform, to estimate the uncertainty in predictions in a deterministic manner and with reduced computational burden while still keeping the important statistical information intact. Furthermore, since the uncertainty in future inputs happens to be uniformly distributed in our example, the UT method also determines the bounds of the uncertainty, which can be extremely valuable information in making decisions based on prognostic estimates in order to contain the risk. Minimizing the computational cost maintains the real-time application potential of the algorithm without compromising on the quality of prediction. This allows us to move a step closer towards adapting prediction algorithms, which are generally inherently stochastic, to meet the needs of current certification procedures and protocols that require determin-

istic system outputs. This also lays the foundation to start investigating other methods that can be used for generalized situations that do not assume a specific distribution type for future input uncertainty.

In the future, we will further investigate these issues on other systems and with additional experimental data. It is also important to investigate more closely the applicability of the UT for sampling the input trajectory space in a variety of practical applications.

#### ACKNOWLEDGMENT

Funding for this work was provided by the NASA System-wide Safety and Assurance Technologies (SSAT) Project.

#### REFERENCES

- Arulampalam, M. S., Maskell, S., Gordon, N., & Clapp, T. (2002). A tutorial on particle filters for on-line nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2), 174–188.
- Barsali, S., & Ceraolo, M. (2002, March). Dynamical models of lead-acid batteries: Implementation issues. *IEEE Transactions on Energy Conversion*, 17(1), 16–23.
- Ceraolo, M. (2000, November). New dynamical models of lead-acid batteries. *IEEE Transactions on Power Systems*, 15(4), 1184–1190.
- Chen, M., & Rincon-Mora, G. A. (2006, June). Accurate electrical battery model capable of predicting runtime and I-V performance. *IEEE Transactions on Energy Conversion*, 21(2), 504 - 511.
- Daigle, M., & Goebel, K. (2010, October). Improving computational efficiency of prediction in model-based prognostics using the unscented transform. In *Proc. of the annual conference of the prognostics and health management society 2010*.
- Daigle, M., & Goebel, K. (2011, August). A model-based prognostics approach applied to pneumatic valves. *International Journal of Prognostics and Health Management*, 2(2).
- Daigle, M., Saha, B., & Goebel, K. (2012, March). A comparison of filter-based approaches for model-based prognostics. In *Proceedings of the 2012 IEEE aerospace conference*.
- Edwards, D., Orchard, M. E., Tang, L., Goebel, K., & Vachtsevanos, G. (2010, September). Impact in input uncertainty on failure prognostic algorithms: Extending the remaining useful life of nonlinear systems. In *Annual conference of the prognostics and health management society*.
- Julier, S. J. (1998, April). A skewed approach to filtering. In *Proc. aerosense: 12th int. symp. aerospace/defense sensing, simulation and controls* (Vol. 3373, p. 54-65).
- Julier, S. J., & Uhlmann, J. K. (1997). A new extension of the Kalman filter to nonlinear systems. In *Proceedings of the 11th international symposium on aerospace/defense sensing, simulation and controls* (pp. 182–193).
- Julier, S. J., & Uhlmann, J. K. (2004, March). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3), 401–422.
- Luo, J., Pattipati, K. R., Qiao, L., & Chigusa, S. (2008, September). Model-based prognostic techniques applied to a suspension system. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 38(5), 1156 -1168.
- Roychoudhury, I., & Daigle, M. (2011, October). An integrated model-based diagnostic and prognostic framework. In *Proceedings of the 22nd international workshop on principles of diagnosis* (p. 44-51).
- Saha, B., Quach, C. C., & Goebel, K. (2012, March). Optimizing battery life for electric UAVs using a Bayesian framework. In *Proceedings of the 2012 IEEE aerospace conference*.
- Sankararaman, S., Ling, Y., Shantz, C., & Mahadevan, S. (2011). Uncertainty quantification in fatigue crack growth prognosis. *International Journal of Prognostics and Health Management*, 2(1).
- Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2010). Metrics for offline evaluation of prognostic performance. *International Journal of Prognostics and Health Management*, 1(1).

# Lithium-ion Battery Remaining Useful Life Estimation Based on Nonlinear AR Model Combined with Degradation Feature

Datong Liu<sup>1</sup>, Yue Luo<sup>2</sup>, Yu Peng<sup>3</sup>, Xiyuan Peng<sup>4</sup>, Michael Pecht<sup>5</sup>

<sup>1,3,4</sup>*Department of Automatic Test and Control, Harbin Institute of Technology, Harbin, 150080, China*

*liudatong@hit.edu.cn*

*pengyu@hit.edu.cn*

*pxy@hit.edu.cn*

<sup>2</sup>*Beijing System Design Institute of Electro-Mechanic Engineering, Beijing, 100854, China*

*ygxdl@126.com*

<sup>5</sup>*CALCE, University of Maryland, College Park, Maryland, 20742, USA*

*pecht@calce.umd.edu*

## ABSTRACT

Long term prediction such as multi-step time series prediction is a challenging prognostics problem. This paper proposes an improved AR time series model called ND-AR model (Nonlinear Degradation AutoRegression) for Remaining Useful Life (RUL) estimation of lithium-ion batteries. The nonlinear degradation feature of the lithium-ion battery capacity degradation is analyzed and then the non-linear accelerated degradation factor is extracted to improve the linear AR model. In this model, the nonlinear degradation factor can be obtained with curve fitting, and then the ND-AR model can be applied as an adaptive data-driven prognostics method to monitor degradation time series data. Experimental results with CALCE battery data set show that the proposed nonlinear degradation AR model can realize satisfied prognostics for various lithium-ion batteries with low computing complexity.

## 1. INTRODUCTION

With high energy density, high galvanic potential, wide temperature range, low self-discharge rate and long lifetime, the lithium-ion battery has been widely used in mobile communications, electric vehicles, aerospace electronics and almost all of the industrial fields with energy supply etc. The lithium-ion battery has gradually become the key techniques for many important areas and industrial applications (Bhaskar Saha & Kai Goebel, 2009) (Jingliang Zhang & Jay Lee, 2011) (Wei He, Nicholas Williard, Michael Osterman, & Michael Pecht, 2011). Due to the safety management, charging and discharging control,

capacity degradation of the lithium-ion battery, capacity fade and remaining useful life (RUL) estimation of lithium-ion batteries has become a hotspot and challenge problem in the fields of reliability, automatic test, power sources, and electric vehicles, etc. As a result, lithium-ion battery RUL estimation and prediction became the hot issues in electronic prognostics and health management (PHM) (K. Goebel, B. Saha, A. Saxena, J. R. Celaya, & J. P. Christophersen, 2008) (F. Rufus & S. Lee, 2008).

At present, among the various approaches of battery State of Charge (SOC) estimation and RUL prediction, it can be generally classified into two categories: data-driven (or statistical data-driven) and model based (Jingliang Zhang & Jay Lee, 2011) (Xiao-Sheng Si, Wenbin Wang, Chang-Hua Hu, & Dong-Hua Zhou, 2011). There are lots of research work focusing on performance degradation, SOC/SOH assessment, RUL estimation for the lithium-ion battery (Bhaskar Saha, Kai Goebel, & Jon Christophersen, 2009) (Bhaskar Saha, Kai Goebel, Scott Poll, & Jon Christophersen, 2009) (Enrico Zio, & Giovanni Peloni, 2011) (Achmad Widodo, Min-Chan Shim, Wahyu Caesarendra, & Bo-Suk Yang, 2011). Especially for the lithium battery prognostics, the prediction uncertainty, and the applicability of the model-based (physics based model, chemistry model, etc.) and data-driven methods have always been the challenge problems in this area.

Lots of researchers such as Bhaskar Saha and Kai Goebel and others researchers in the Prognostics Center of Excellence (PCoE) of the NASA AMES Center achieved the battery RUL prediction as well as the uncertainty representation and management with particle filter (PF) algorithm (Bhaskar Saha, Kai Goebel, & Jon Christophersen, 2009) (Bhaskar Saha, Kai Goebel, Scott Poll, & Jon Christophersen, 2009). Moreover, the Artificial Neural Networks (ANN) (Jie Liu, Abhinav Saxena, Kai Goebel,

Datong Liu et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Bhaskar Saha, & Wilson Wang, 2010), Extended Kalman Filter (EKF) (Lijun Gao, Shengyi Liu, & Roger A. Dougal, 2002), Support Vector Machine (SVM) (Bhaskar Saha, Kai Goebel, Scott Poll, & Jon Christophersen, 2007), Relevance Vector Machine (RVM) (B. Saha, S. Poll, & K. Goebel, 2007), Gaussian Process Regression (GPR) (Bhaskar Saha, Kai Goebel, & Jon Christophersen, 2009) and other machine learning and statistical algorithms are applied in lithium-ion battery prognostics. At the same time, lots of physical model, chemistry model and other related empirical model are developed or applied in the battery RUL estimation.

All of the RUL estimation framework became an effective and practical approach for lithium battery degradation analysis and estimation. However, in the complicated operating conditions, to obtain the model or to identify the model parameters is very difficult for physical model or chemistry model. The drawback is that it does not consider the varied operation condition for on-line application. In the other hand, most of the data-driven prognostics algorithm are of high computing complexity, it is hard to realize or compute for some simple application.

To develop efficient RUL estimation method for the real-time prognostics of lithium batteries, this paper proposes an improved AR time series prediction model based on analyzing the nonlinear degradation of capacity of lithium batteries. This paper first introduces the basic principle of the AR model, and then with the battery RUL estimation experiment, the “accelerated” degradation factor is extracted based on the experimental result. The improved so-called ND-AR model is described to achieve satisfied long term prediction of the status of lithium battery. Experimental results with the CALCE battery data set show that the algorithms can be effectively applied to RUL prognostics for lithium battery capacity degradation with better performance in both efficiency and accuracy.

## 2. AR TIME SERIES PREDICTION MODEL

Time series analysis and prediction based on stochastic process theory and mathematical statistics has been widely applied in signal processing, intelligent information analysis and PHM etc. In the engineering field, the AR model is used more extensive than the MA model and the ARMA model, because the parameter identification of the AR model is relatively simple, as well as the computing load is small. Furthermore, it has already proved the MA model and ARMA model can be equivalent by higher order AR model (Jianqing Fan, & Qiwei Yao, 2003). The degradation of battery capacity data is based on observations and calculated time-series data, it can take advantage of the AR model in time series analysis techniques to study.

### 2.1. The AR Model

The AR model is first proposed for time series analysis.

For time series  $\{x_t\}$ ,

$$\begin{cases} x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \dots + \phi_p x_{t-p} + a_t \\ \phi_p \neq 0 \\ E(\varepsilon_t) = 0, \text{Var}(\varepsilon_t) = \sigma_\varepsilon^2, E(\varepsilon_t \varepsilon_s) = 0, s \neq t \\ E x_s \varepsilon_t = 0, \forall s < t \end{cases} \quad (1)$$

It is defined as AR model of  $p$  order as  $AR(p)$ . The latter three constraints in equation (1) can be omitted. At this time, the AR model is described as the time series  $\{x_t\}$  can equals to the linear function of the historical value and random noise.

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \dots + \phi_p x_{t-p} + a_t \quad (2)$$

Here the  $\phi$  is the coefficient of autoregression,  $p$  is the order of the model.  $a_t, t=0, \pm 1, \dots$  is the independent white noise sequence with mean zero and variance  $\sigma_a^2$ . In the  $AR(p)$  model, the number of the parameters is  $p+2$ , the parameters are the order  $p$ , coefficients  $\phi_1, \phi_2, \dots, \phi_p$  and  $\sigma_a^2$ .

It can be seen from the equation (2) that the AR model is linear prediction function.

### 2.2. The Order and Parameters Estimation for AR Model

While the AR model is applied to time series prediction, the selection of the order the model is a key factor. Because the coefficients  $\{\phi_p\}$  are relevant to the order  $p$ , to obtain the reasonable coefficients  $\{\phi_p\}$ , we should first select the suitable order for the AR model.

In this paper, the AIC (Akaike Information Criterion, AIC) method (Akaike H, 1974) is applied for the determination of the model order. The AIC method is defined as follows.

$$AIC(p) = N \ln \sigma_p^2 + 2p \quad (3)$$

Here the  $p$  is the determination of the model order,  $n$  is the number of the data sample,  $\sigma_p^2$  is the prediction variance of  $p$  order model.

The methods for parameters estimation of the AR model includes the least square estimation, Maximum Likelihood Estimation, Yule-Walker method (autocorrelation method), the Burg method and covariance method, etc. In this paper, the Burg algorithm that can directly calculate the parameters with the observed time series is applied to realize parameters estimation of the AR model. This algorithm can avoid the priori estimation of the autocorrelation function, as a result, the computing is simple, and the real-time performance is excellent. Especially, the Burg algorithm is suitable for the parameters estimation of short time series, which meets the demand of the battery remaining useful life estimation.

### 3. NONLINEAR DEGRADATION AR MODEL FOR BATTERY REMAINING USEFUL LIFE ESTIMATION

In this part, the AR model is applied to achieve the battery RUL estimation with the long term time series prediction. Based on the prediction result, the unsuitability for the nonlinear degradation of lithium battery RUL estimation is analyzed. At last, the nonlinear degradation AR model for battery RUL estimation is proposed to expect better prognostics results.

#### 3.1. Lithium-ion Battery RUL Estimation based on AR Model

Before the modeling to the Lithium battery capacity data with AR model, the order and the parameters of the AR model should be determined. From the experience perspective, the order value of the AR model should not be more than 10. The order  $p$  value could be optimized and determined using the AIC criterion according to the evaluation precision.

According to equation (3), when the order  $p$  increases from one gradually,  $AIC(p)$  will get the minimum value for certain  $p$  value. The corresponding  $p$  is the suitable order of the AR model.

Here we conduct analysis using the NASA battery data set (B. Saha, & K. Goebel, 2007), the battery No. 05 is selected as modeling data set, the order of the AR model with AIC method. Figure 1 and Table 1 shows the AIC value while the order  $p$  varies from 1 to 10.

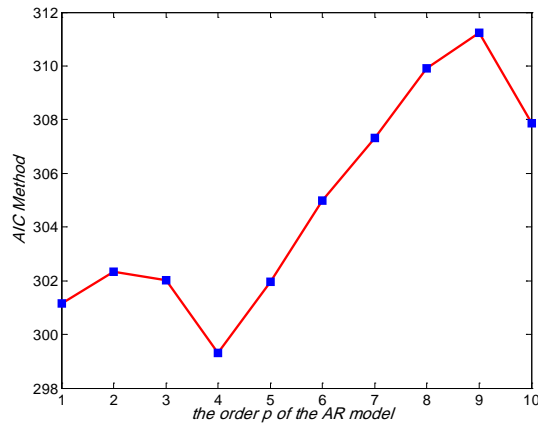


Figure 1. the order of the AR model with AIC method for NASA battery No. 05

From the figure 1 and Table 1, we can conclude that the best order  $p$  value for the battery degradation data is 4 while the corresponding AIC obtains the most optimized value. After the determination of the order  $p$ , if the prediction starting point is  $T$  cycle, we could realize the parameters estimation with Burg algorithm to the observed capacity data Capacity (1:  $T$ ), and then the prediction could be implemented with the AR model.

Order $p$	AIC Value
1	201.6376
2	198.5693
3	200.7816
4	196.6361
5	203.1455
6	205.2109
7	205.4390
8	206.8059
9	207.0218
10	202.0553

Table 1. the order of the AR model with AIC method for NASA battery No. 05

Figure 2 shows the battery remaining useful life estimation result (the battery No. 05 of NASA PCoE Center) with different starting point with 4-order AR model. The detail prediction result is shown as Table 2.

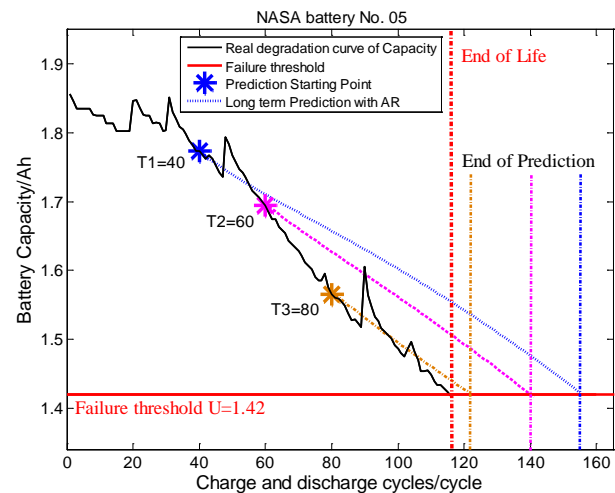


Figure 2. the battery RUL estimation at different starting point with AR model(NASA battery No. 05)

Starting point	End of Prediction (cycle)	RUL prediction result(cycle)	prediction error(cycle)
T1=40	154	114	38
T2=60	140	80	24
T3=80	122	42	6

Table 2. Error comparison of long term prediction with AR model at different starting point

In the experiment, the prediction is fulfilled at three different starting points respectively:  $T1 = 40$ cycle,  $T2 = 60$ cycle and  $T3 = 80$ cycle (marked in the figure 2). The EoL (End of Life) of the battery No. 05 is about 116 cycle as shown in the Fig 2. We defined that when the capacity of

the battery degraded to the 70% of its SOC, the battery reaches its EoL (in the experiment, while the capacity of the battery degraded to 1.42Ah, we define as the End of Life). We can see that different prediction results are obtained at different prediction starting points.

### 3.2. Analysis of Nonlinear Degradation of Lithium-ion Battery

From the Table 2, we can see that, at the early stage (T=40) and medium-term (T=60) the RUL prediction results are not satisfied. Although the AR model could be applied to realize trend prediction in time series analysis, the AR model is still a linear method. To analyze the degradation trend of the lithium battery, we could find that with the degradation process developing, the degradation rate will accelerate with the increasing cycle number.

So we can see that the prediction function could not track the “accelerated” degradation process. It means that with the development of the degradation with the charging and discharging, it shows an accelerated degradation trend from the lifetime monitoring data. Especially, at the early stage and medium stage, the RUL prediction result could not satisfy the real application.

### 3.3. Nonlinear Degradation AR Model

To solve the poor prediction accuracy and improve the RUL estimation performance with AR model, the “accelerated” degradation factor should be considered. To implement more precise degradation trend tracking, an accelerated factor could be modify the un-matching of the AR model. To check the un-matching phenomenon carefully, we can find that with the degradation process develops (the degradation cycle increases), the degree of the un-matching strengthens.

The accelerated degradation characteristics above can be understood as follows. With the charging and discharging cycle, the inner lithium-ion decreases and the resistance increases. With this degradation process developed, the degradation trend will be accelerated with the increasing of the inner resistance. As a result, the power loss will be gradually increased leading the accelerated degradation process.

With this accelerated degradation factor, the linear AR time series prediction value could be supplemented. Moreover, with this idea, the high efficiency of the AR model could be kept well. According to the analysis above, we proposed an improved battery RUL estimation approach with AR model combined with nonlinear degradation process (accelerated degradation process with the cycle increasing). We call this approach as nonlinear degradation AR model (ND-AR model). The ND-AR model is defined as follows.

An “accelerated” factor is add to the AR model output to match the battery degradation process:

$$x_t = K_T \times [\phi_1 x_{t-1} + \phi_2 x_{t-2} + \dots + \phi_p x_{t-p} + a_t] \quad (4)$$

Here the  $K_T$  is the “accelerated” factor. Considering the accelerated factor is correlated to the degradation cycle, we define the  $K_T$  as follows considering the nonlinear degradation process analyzed above.

$$K_T = \frac{1}{1 + a^*(k + b)} \quad (5)$$

In equation (5), k is the prediction step, and the  $K_T$  become the time varied accelerated factor with the prediction process.

While the parameters estimation of the AR model is fulfilled, the parameters in equation (5) could be obtained by curve fitting or least square estimation.

## 4. EXPERIMENTAL AND DISCUSSION

### 4.1. Battery Data set

The battery data set is from the CALCE of University of Maryland. The lithium-ion batteries were tested to discover the degradation of the capacity. The cycling of the batteries was implemented with the Arbin BT2000 battery testing system under room temperature. The 1.1Ah rated capacity of batteries are adopted in the experiment with the discharging current (0.45A that the discharging speeds is 0.5C) (Wei He, Nicholas Williard, Michael Osterman, & Michael Pecht, 2011). The battery capacity degradation of different batteries are shown as Figure 3.

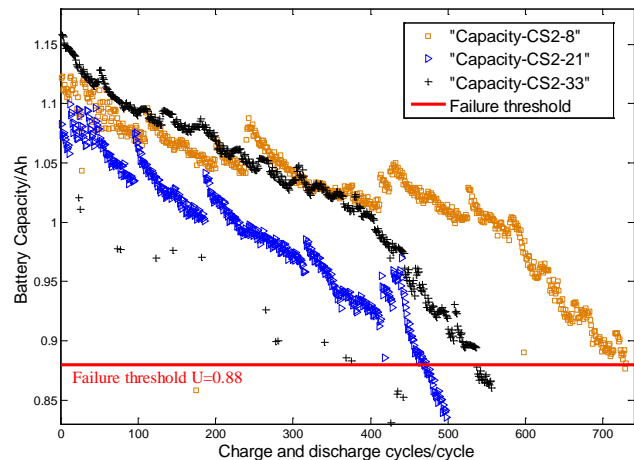


Figure 3. the capacity degradation of different batteries (CALCE battery data)

### 4.2. Parameters Estimation

The parameters of AR model containing the order p and other parameters are determined using the same method as described in section 3.1. The order p value equals to 4 while the model gets best prediction performance.



With the curve fitting method, the parameters a and b in the ND-AR model are estimated,  $a=1.5e-7$ ,  $b=100$ .

The other parameters are obtained with Burg algorithm to the observed capacity degradation BCm (1: T) with the modeling process.

### 4.3. Battery RUL estimation with ND-AR model

Figure 4, Figure5, and Figure 6 show the prediction result for various lithium battery capacity degradation data with the ND-AR model proposed in this paper. From Fig.4, Fig5, and Fig.6, we can conclude that the capacity degradation process under different testing and operating condition is forecasted precisely with the proposed ND-AR model. The prediction and estimation of RUL will be beneficial for the process control and maintenance of the lithium batteries.

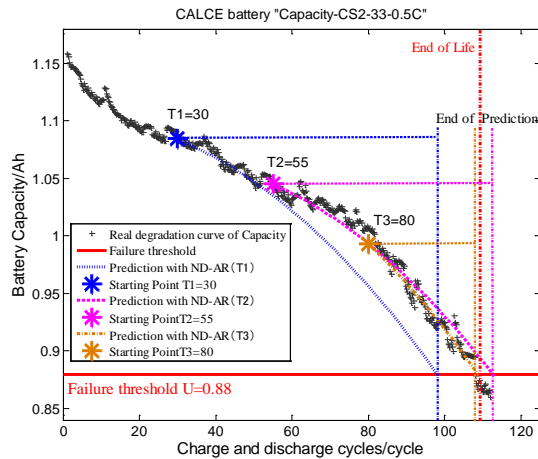


Figure 4. the battery RUL estimation at different starting point with ND-AR model(CALCE battery“Capacity-CS2-33-0.5C”)

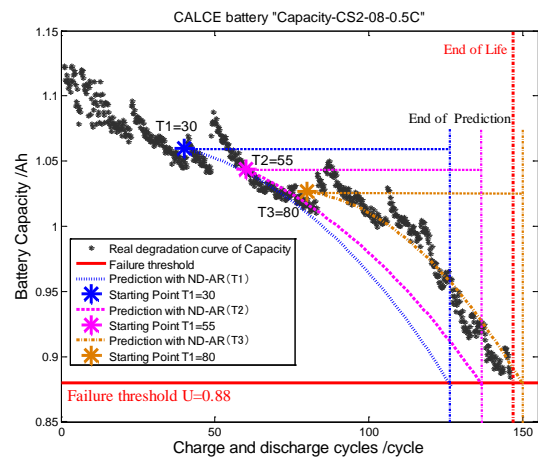


Figure 5. the battery RUL estimation at different starting point with ND-AR model(CALCE battery“Capacity-CS2-08-0.5C”)

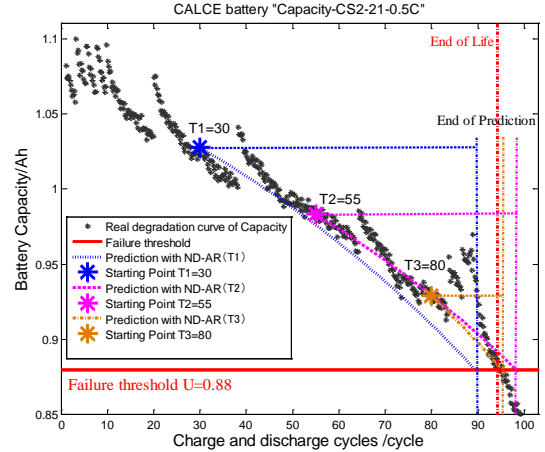


Figure 6. the battery RUL estimation at different starting point with ND-AR model(CALCE battery“Capacity-CS2-21-0.5C”)

### 4.4. Results analysis and comparison

To evaluation the proposed ND=AR model, we compare the battery RUL estimation results of both AR and ND-AR model. The prediction result is shown as figure 6 for one of testing battery.

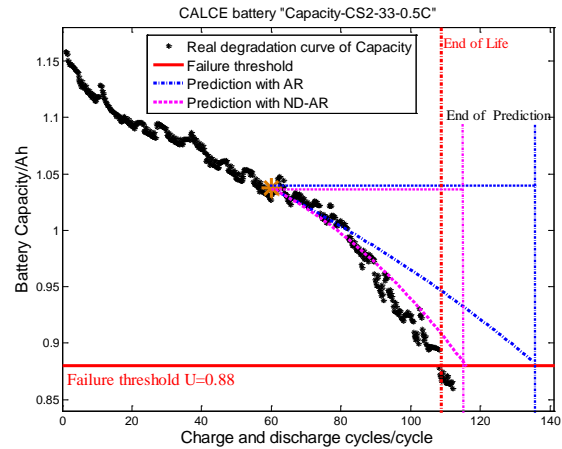


Figure 7.the battery RUL estimation at different starting point with ND-AR model(CALCE battery Capacity-CS2-33-0.5C)

From the figure 7, we can conclude that compared to the basic AR model, the proposed ND-AR model can realize more satisfied prediction result at the same starting point.

To evaluate the comparison result quantitatively, we adopt the Mean Absolute Error(MAE) and Root Mean Square Error (RMSE) and Error of RUL estimation to analyze the prediction results with two methods.

The definition of MAE:

$$MAE = \frac{1}{n} \sum_{i=1}^n |x(i) - \bar{x}(i)| \quad (6)$$

RMSE:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n [x(i) - \bar{x}(i)]^2} \quad (7)$$

Error of RUL:

$$E_{RUL} = |RUL_{real} - RUL_{prediction}| \quad (8)$$

Here  $n$  is the number of prediction data set,  $x(i)$  is the real value of testing and monitoring of battery capacity,  $\bar{x}(i)$  is the prediction value. In the experiment,  $k$  is the prediction steps from the starting point.

The detail result is shown as Table 3.

Index of batteries	CALCE No. 8	CALCE No. 21	CALCE No. 33
MAE of ND-AR	0.0060	0.0057	0.0066
MAE of AR	0.0304	0.0287	0.0317
RMSE of ND-AR	0.0113	0.0105	0.0126
RMSE of AR	0.0349	0.0316	0.0397
$E_{RUL}$ of ND-AR	10	8	7
$E_{RUL}$ of AR	34	27	28

Table 3. Comparison of AR and ND-AR model for battery RUL prediction

From the Table 3, we can find that, the prediction MAE, RMSE and Error of RUL of the ND-AR model are superior than the AR model for various lithium battery.

## 5. CONCLUSIONS

This paper explores an improved nonlinear degradation AR model for battery remaining useful life estimation. The main contribution of this research can be concluded that: (1) low computing complexity AR time series model is applied for battery remaining useful life prediction with the monitoring and testing data, the real-time performance of the proposed method is high. (2) The “accelerated” nonlinear degradation feature of the battery capacity fade is analyzed based on experiment. (3) A nonlinear degradation factor is extracted to combined with standard AR time series prediction model to realize better RUL estimation result and more precisely prediction result could be fulfilled. With the experiment we can conclude that the improved model is suitable for cycle life estimation of the lithium battery as well as low computing application. This proposed NA-AR lithium-ion battery RUL prognostics method shows better prospective in industrial application comparing with RUL prediction based on linear AR model or other time series prediction methods.

## 6. FUTURE WORK

In future, we will consider the uncertainty representation ability of the proposed time series based data-driven method. The dynamic parameters training and models fusion for

battery with complex operating condition should be focused in the future research work.

## ACKNOWLEDGEMENT

This research work is supported by Research Fund for the Doctoral Program of Higher Education of China (20112302120027). The author would also express his sincere thanks to Dr. Wei He at CALCE of University of Maryland and Dr. Eden Ma at PHM Center of City University of Hong Kong for their help on the battery data set.

## REFERENCES

- Bhaskar Saha, Kai Goebel (2009). Modeling li-ion battery capacity depletion in a particle filtering framework. Annual Conference of the Prognostics and Health Management Society.
- Jingliang Zhang, Jay Lee (2011). A review on prognostics and health monitoring of Li-ion battery. Journal of Power Sources, 196, 6007-6014.
- Wei He, Nicholas Williard, Michael Osterman, Michael Pecht (2011). Prognostics of lithium-ion batteries based on Dempster-Shafer theory and the Bayesian Monte Carlo method. Journal of Power Sources, 196, 10314-10321.
- Goebel, B. Saha, A. Saxena, J. R. Celaya, J. P. Christophersen (2008). Prognostics in battery health management, IEEE Instrumentation & Measurement Magazine. 8, 33-40.
- F. Rufus, S. Lee, A. Thakker (2008). Health Monitoring Algorithms for Space Application Batteries. In Proceedings of International Conference on Prognostics and Health Management.
- Xiao-Sheng Si, Wenbin Wang, Chang-Hua Hu, Dong-Hua Zhou (2011). Remaining useful life estimation – A review on the statistical data driven approaches. European Journal of Operational Research, 213(1), 1-14.
- Bhaskar Saha, Kai Goebel, Jon Christophersen (2009). Comparison of prognostic algorithms for estimating remaining useful life of batteries. Transactions of the Institute of Measurement and Control. 31, 293-308.
- Bhaskar Saha, Kai Goebel, Scott Poll, Jon Christophersen (2009). Prognostics Methods for Battery Health Monitoring Using a Bayesian Framework, IEEE Transactions on Instrumentation and Measurement, 58(2), 291- 297.
- Enrico Zio , Giovanni Pelsoni (2011). Particle filtering prognostic estimation of the remaining useful life of nonlinear components, Reliability Engineering and System Safety, 96, 403-409.
- Achmad Widodo, Min-Chan Shim, Wahyu Caesarendra, Bo-Suk Yang (2011). Intelligent prognostics for battery health monitoring based on sample entropy, Procedia Engineering, 14, 2707-2713

- Jie Liu, Abhinav Saxena, Kai Goebel, Bhaskar Saha, and Wilson Wang (2010). An Adaptive Recurrent Neural Network for Remaining Useful Life Prediction of Lithium-ion Batteries. Annual Conference of the Prognostics and Health Management Society.
- Lijun Gao, Shengyi Liu, Roger A. Dougal (2002). Dynamic Lithium-Ion Battery Model for System Simulation. IEEE Transaction on Components and Packaging Technologies, 25(3), 495-505.
- Jianqing Fan, Qiwei Yao (2003) Nonlinear Time Series: Nonparametric and parametric methods. SPRINGER, 12-123.
- Akaike H (1974). A New Look at the Statistical Model Identification. IEEE Transactions on Automatic Control. 19, 716-723
- B. Saha, K. Goebel (2007). Battery Data Set, NASA Ames Prognostics Data Repository, [<http://ti.arc.nasa.gov/project/prognostic-data-repository>], NASA Ames, Moffett Field, CA.

## BIOGRAPHIES

**Datong Liu** received the B.Sc. and M.Sc. degrees in Department of Automatic Test and Control from Harbin Institute of Technology (HIT), Harbin, China in 2003 and 2005, respectively. During 2001 to 2003, he also minored the Computer Science and Technology in HIT. He received the Ph.D. degree in major of measurement and instrumentation from HIT in 2010. He is now an assistant professor in Department of Automatic Test and Control, HIT. His research interests include automatic test and intelligent information processing, time series analysis, Data-driven PHM, Machine Learning, Data Mining, etc. He is currently an IEEE member, ACM member, PHM society member, China Computer Federation member. He has published more than 20 conference and journal papers, and holds 7 invention patents and more than 20 invention patents pending in China. He is now in charge of 5 projects related to PHM that Supported by Research Fund for the Doctoral Program of Higher Education of China, 12.5 government advanced research fund in China, etc.

# Sensor Selection with Grey Correlation Analysis for Remaining Useful Life Evaluation

Peng Yu, Xu Yong, Liu Datong, Peng Xiyuan

*Automatic Test and Control Institute, Harbin Institute of Technology, Harbin, 150001, China*

*pengyu@hit.edu.cn  
xuyong.hit@gmail.com*

## ABSTRACT

Sensor selection in data modeling is an important research topic for prognostics. The performance of prediction model may vary considerably under different variable subset. Hence it is of great important to devise a systematic sensor selection method that offers guidance on choosing the most representative sensors for prognostics. This paper proposes a sensor selection method based on the improved grey correlation analysis. From empirical observation, all the continuous-value sensors with a consistent monotonic trend are firstly selected for data fusion, and a linear regression model is used to convert the multi-dimensional sensor readings into one-dimensional health factor (HF). The correlation between HF and each of the selected sensors is evaluated by calculating the grey correlation degree defined on two time series. The optimal sensor subset with a relatively large correlation degree is selected to execute the final fusion. The effectiveness of the proposed method was verified experimentally on the turbofan engine simulation data supplied by NASA Ames, using instance-based learning methodology, and the experimental results showed that RUL prediction with fewer sensor inputs can obtain a more accurate prognostics performance than using all sensors initially considered relevant.

## 1. INTRODUCTION

Prognostics and health management (PHM) of complex engineered systems has gained increasing attention from the research community worldwide. Prognostic builds the foundation of PHM, and its outcome directly affects the other PHM components such as operations planning, timely maintenance, logistics, etc. Hence prognostics can play an important part in reducing cost, increasing safety, and accomplishing critical missions (Heng et al., 2008). Generally, prognostics can be divided into the detection of failure precursors and the prediction of remaining useful life

(RUL). Comparing to the detection of failure precursors, RUL prediction is mostly irrelevant to the application. The methods for RUL prediction are almost the same to all prognostics applications.

The ultimate aim of most prognostics systems is accurate estimation the RUL of individual systems, and the prediction accuracy relies not only on the prediction models used, but also on the types and number of sensors selected (Cheng et al., 2010). As the performance of system degrades, the monitored parameters tend to change accordingly. These raw multi-dimensional sensor data or features extracted from them may be used to track the degradation behavior of system. Typically, these degradation data can be used as the inputs of data-driven prognostics model to make RUL estimation. Degradation data may consist of sensor readings, such as temperature and pressure, or inferred features, such as model residuals or physics-based model predictions. Commonly, it is beneficial to fuse sensor readings and inferred features into a single health factor, which is considered as a more robust input to the prognostics model (Coble, 2010). However, inclusion of irrelevant or redundant variables during the fusion may lead to over-fitting or less sensitivity of prognostics model, which is adverse to the prediction performance. Hence variable selection is critical to make an accurate RUL estimation. Typically, sensor selection is left to expert knowledge, empirical observation of available data, and intimate knowledge of degradation mechanisms. These methods are time-consuming, and scale with the number of available sensors and possible fault modes (Zhang, 2005).

For many real-world systems, it is almost impossible to fully understand the degradation behavior of the systems and employ the first-principle models for prognostics. Since Instance-based learning (IBL) approach develops prognostics model based on a mass of historical instances, it becomes a preferable choice (Xue et al., 2008). As the rapid development of communication and sensor technology, abundant data collection from complex systems, such as aircraft engines, satellite power system, etc, becomes possible. These massive life-cycle condition data collected

---

Peng Yu et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

from various instances of the equipment further promotes the application of IBL prognostics approach.

The current work focuses on improving the RUL prediction accuracy of complex systems via sensor selection. In real applications, complex system is operated under dynamic operating conditions, and k-means clustering algorithm is employed to cluster the operational conditions into a finite number of operating regimes. According to the empirical observation, those sensors with a consistent monotonic trend under different operating regimes are selected, and a linear regression model is employed to convert the selected multivariate sensor readings from individual regimes into a one-dimensional HF, then all the HFs are merged to form a complete HF time series with the original time stamps. The correlation between HF and each of the selected sensors is evaluated by calculating the grey correlation degree defined on two time series. The optimal sensor subset with a relatively large correlation degree is selected to execute the final fusion. Finally, the obtained HF time series is integrated into an IBL prognostics architecture consisting of model recognition, similarity evaluation and RUL prediction. Moreover, the performance of the sensor selection strategy is verified experimentally on the turbofan engine simulation data supplied by NASA Ames, using the integrated IBL prognostics architecture.

The paper is organized into the following sections. In section 2, the sensor selection scheme based on improved grey correlation analysis is introduced. In section 3, the IBL prognostics architecture is summarized. The turbofan engine application is elaborated in section 4. The experiment results and discussions are presented in section 5. Conclusion is drawn in section 6.

## 2. SENSOR SELECTION SCHEME

Sensor selection is mostly relevant to the application, and it aims at reducing the unnecessary redundancy while maximizing the relevance in the sensor subset (Wang et al, 2008). According to the characteristics of collected data, the sensor selection scheme probably consists of operation condition division, empirical observation, data fusion and grey correlation analysis.

### 2.1. Operation Condition Division

In real applications, the dynamic operation conditions have a great impact on the sensor readings or inferred features from the system and complicate the system degradation behaviors over time. The sensor time series may show little trend. However, if the operation conditions are clustered into several operating regimes by the use of certain clustering algorithm, the sensor data collected from different regimes may exhibit a prominent trend.

The vector  $\mathbf{c}_i$  represents the operation conditions of the system at time  $t_i$ . Suppose the operational conditions  $\mathbf{c}$  can be concentrated into a limited number of operating regimes  $\mathbf{O}=\{O_1, O_2, \dots, O_p\}$  using k-means clustering algorithm  $f$ . The output of  $f$  is defined as:

$$\mathbf{S}=f(\mathbf{c})=(S_1, S_2, \dots, S_p) \quad (1)$$

where  $S_p$  is membership score when  $\mathbf{c} \in O_p$ .

In case of discrete operation conditions, the output of  $f$  can be simplified as:

$$f(\mathbf{c})=\arg \max _{k=1, \dots, P} S_k \quad (2)$$

### 2.2. Empirical Observation

The objective of empirical observation is to eliminate the sensors which are obviously not suitable for prognostics modeling. The visual inspection procedures of sensor data under different operating regimes are as follows:

1. Some sensors with one or multiple discrete values are firstly discarded, from which it is difficult to track the degradation trend of the system.
2. Some other sensors have continuous values, but exhibit non-monotonic trend during the life time of the instances, should also be discarded.
3. All the remaining sensors with continuous values exhibit a monotonic trend, but some of them show inconsistent evolution trend among the different instances, which may represent various fault modes of the system. As it is hard to quantize or identify the fault modes without system relevant information, those sensors with inconsistent trend are eliminated.
4. Only those sensors with a consistent monotonic trend under different operating regimes are selected for data modeling or other processing.

### 2.3. Data Fusion

In this paper, data fusion refers to convert the selected multivariate sensor data from individual regimes into a single HF within a normalized range. Therefore, the HFs obtained from each regime can be merged to form a new one-dimension time series, as described in Fig.1.

For operating regime  $O_p$ , one local regression model is created:

$$z=h(\mathbf{x}; \boldsymbol{\theta}^{(p)}) \quad (3)$$

where  $\mathbf{x}$  represents the multivariate sensor data, and  $\boldsymbol{\theta}^{(p)}$  denotes the local model parameters.

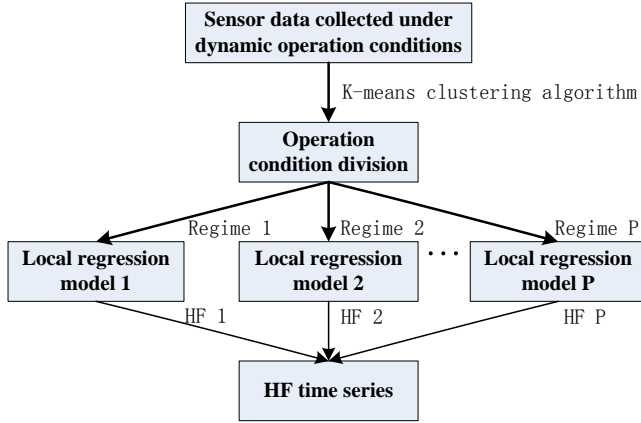


Figure 1. Data processing for RUL modeling through multi-regime health assessment

To make the HF comparable under different operating regime, the obtained HF should be normalized to a range, usually between 0 and 1, hence, the learning procedures of a local regression model can be followed:

- For those samples  $(\mathbf{c}_i, \mathbf{x}_i)$  collected at early stage, that is  $t_i < T_1$ , assign the matching output  $z_i = 1$ ;
- For those samples  $(\mathbf{c}_i, \mathbf{x}_i)$  collected at middle stage, that is  $T_1 < t_i < T_2$ , will not participate model training;
- For those samples  $(\mathbf{c}_i, \mathbf{x}_i)$  collected at late stage, that is  $t_i > T_2$ , assign the matching output  $z_i = 0$ .

The number and parameters of local models vary with the assigned thresholds  $T_1$  and  $T_2$ , which can greatly influence the model performance. Commonly, the parameters are chosen by the rule of thumb, e.g.  $T_1 = t_E * 10\%$  and  $T_2 = t_E * 90\%$ , where  $t_E$  represents the whole lifespan of the instance. Above all, once the sensor data of training instances  $\mathbf{x}_i$  are provided, they will be divided into finite groups in accordance with different operating regimes and applied to learn different local regression models.

#### 2.4. Grey Correlation Analysis

As mentioned, the sensors are preliminary selected by empirical observation, and the sensor fusion method is provided. One major problem lies in whether the sensor subset can be further optimized. Due to large noise and low sensitivity, some sensors exhibit an unclear trend compared with the others. Including them in the data fusion may lower the prediction accuracy. Hence certain analysis method should be adopted to further select the sensors.

Grey correlation analysis is a principle theory of grey system theory, which can be applied in grey system analysis and random variables processing (Zhang & Zhang, 2007). The correlation between factors is represented by the

similarity level of geometry which is called grey correlation degree, and the correlation degree between reference sequences and comparison sequences can be quantitatively estimated. Grey correlation degree describes the relative change between different factors in the process of system evolution, and the larger the correlation degree is, the higher the similarity level is. Thus, the correlation degree can represent the impact of different sensors on the system degradation behaviors. By the calculation of improved correlation degree between HF time series and sensor time series, the sensors with a relatively large correlation degree are selected since they have a larger impact on the HF. The calculation steps of grey correlation degree between HF and sensors are as follows:

1. The HF time series which can represent the system degradation behaviors is set as the reference sequences  $\mathbf{Z} = \{z(k) | k=1,2,\dots,n\}$ , and the sensor time series which can affect the system degradation behaviors are set as the comparison sequences  $\mathbf{X}_i = \{X_i(k) | k=1,2,\dots,n\}, i=1,2,\dots,m$ .
2. Due to the various units of measurements, the dimensions of sensor data are different, which may lead to a wrong correlation analysis result. Thus, the data should be converted into dimensionless form. There are many dimensionless processing methods, such as equalization, initialization, etc. In this paper, initialization method is applied, and the whole data in the original sequences are divided by the initial data, which is shown as follows:

$$x_i(k) = \frac{X_i(k)}{X_i(1)}, k=1,2,\dots,n; i=1,2,\dots,m \quad (4)$$

3. The correlation is substantially the fitting degree of geometry between curves, thus the difference between curves is considered as the performance indicator of correlation. Set  $\Delta_i(k) = |z(k) - x_i(k)|$ , the grey correlation coefficient between  $z(k)$  and  $x_i(k)$  is:

$$\xi_i(k) = \frac{\min_i \min_k \Delta_i(k) + \rho \max_i \max_k \Delta_i(k)}{\Delta_i(k) + \rho \max_i \max_k \Delta_i(k)} \quad (5)$$

where  $\rho \in (0, \infty)$ , and its effect lies in enhancing the significance of difference between the correlation coefficients. The value of  $\rho$  is commonly set as 0.5.

4. The correlation coefficient represents the correlation between reference sequences and comparison sequences at various points, and there is a corresponding correlation coefficient at each point. However, the decentralized information is inconvenient for holistic comparison. To solve the problem, the correlation degree is proposed, which can be represented by the mean of the correlation coefficients:

$$r_i = \frac{1}{n} \sum_{k=1}^n \xi_i(k), k = 1, 2, \dots, n \quad (6)$$

5. The above-mentioned correlation degree is calculated without considering the diversity of correlation coefficients at different point. Therefore, the stability of correlation coefficient sequences is proposed:

$$S(r_i) = \sqrt{\frac{1}{n} \sum_{k=1}^n (\xi_i(k) - r_i)^2} \quad (7)$$

On the basis of the stability, the computing model of grey correlation degree is improved:

$$r^*(i) = \frac{r_i}{1 + S(r_i)} \quad (8)$$

The object of calculating grey correlation degree lies in comparing the impact of different sensors on system degradation behaviors. If  $r_1 > r_2$ , the comparison sequence  $\mathbf{X}_1$  has a greater impact on  $\mathbf{Z}$  than  $\mathbf{X}_2$ .

### 3. IBL PROGNOSTICS ARCHITECTURE

From the above-mentioned sensor selection scheme, an optimal sensor subset is selected, and HF time series is formed by sensor data fusion, which is integrated into IBL prognostics architecture. In this architecture, a number of HF time series extracted from the historical monitoring data of the training instances with known failure times are used to form a library of degradation models. The similarity between a test instance and each of the models are evaluated, and each model can give an individual RUL estimation to the test instance. These RUL estimations can fuse into a final RUL prediction by the similarity-weighted sum. The architecture consists of model recognition, similarity evaluation and RUL prediction.

#### 3.1. Model Recognition

The HF time series extracted from one training instance is available to establish a model depicting the whole performance degradation process of the instance, and the model library  $\{M_i\}$  can be constructed based on the multiple HF time series extracted from training instances.  $M_i$  is commonly a deterministic model that can give a predicted output at a given time:

$$M_i : y = m_i(t) + \varepsilon, -T_i \leq t \leq 0 \quad (9)$$

where  $\varepsilon$  is noisy term,  $T_i$  is the lifetime of the training instance used to establish the model.

The selection of model type is application dependent. For complex engineered systems, the main consideration is commonly focused on the long term degradation trend of the system, and the fluctuations in the degradation process can

be recognized as interference or noise. Hence, a model with smoothing function of the time series can be adopted.

#### 3.2. Similarity Evaluation

The definition of similarity between different instances has a great impact on the performance of IBL prognostics method. In this paper, grey correlation degree and Euclidean distant are adopted to respectively represent the similarity between the test instance  $\mathbf{Z}_T = \{z(k) | k=1, 2, \dots, r\}$  and degradation model  $M_i$ . The calculation steps of grey correlation degree between different data sequences have already been introduced in section 2.4 and will not be reiterated here.

The Euclidean distant between  $\mathbf{Z}_T$  and  $M_i$  is defined as:

$$D(\tau, \mathbf{Z}_T, M_i) = \sum_{j=1}^r (z(j) - m_i(-\tau - r + j))^2 / \sigma_i^2 \quad (10)$$

where  $0 \leq \tau \leq T_i - r + 1$ ,  $\tau$  represents the time span that the time series  $\mathbf{Z}_T$  is moved away from cycle zero of model  $M_i$ , and  $\sigma_i^2$  is the prediction variance provided by  $M_i$ . The smaller the distance is, the higher the similarity is.

Moreover, the similarity degree between  $\mathbf{Z}_T$  and  $M_i$  is defined as:

$$S(\tau, \mathbf{Z}_T, M_i) = \exp(-D(\tau, \mathbf{Z}_T, M_i)) \quad (11)$$

#### 3.3. RUL Prediction

Once the definition of similarity is defined, each model  $M_i$  in the library can give an individual RUL estimation to the test instance:

$$RUL_i = T_i - r - \arg \min_{\tau} D(\tau, \mathbf{Z}_T, M_i) \quad (12)$$

All RUL predictions and corresponding similarity degrees form a set  $\{(RUL_i, S(\tau, \mathbf{Z}_T, M_i)) | i=1, 2, \dots, I\}$ , where  $I$  represents the number of models in the library. The similarity-weighted method is applied to fuse all the RUL predictions in the set to get a final RUL estimation:

$$RUL_r = \frac{\sum_{i=1}^I S(\tau, \mathbf{Z}_T, M_i) \cdot RUL_i}{\sum_{i=1}^I S(\tau, \mathbf{Z}_T, M_i)} \quad (13)$$

### 4. CASE STUDY

In this section, the performance of the sensor selection scheme will be validated experimentally on the turbofan engine simulation data available from NASA Ames Prognostic Data Repository (Saxena & Goebel, 2008), using the integrated IBL prognostics architecture.

	Set 1	Set 2	Set 3	Set 4
Fault modes	1	1	2	2
Operation condition	1	6	1	6
Training units	100	260	100	248
Testing units	100	259	100	249

Table 1. Experiment settings of the data sets

#### 4.1. Data Description

Four data sets with different simulation settings such as the number of operation conditions and fault modes are provided by NASA, and these data sets consist of multivariate time series from multiple instances of the turbofan engine. The collected data for each instance consists of a 24-dimensional time series (3 operation conditions and 21 sensor readings for each flight cycle), and these data can represent the condition of engine throughout its flight history. The experiment settings of the data sets are described in Table 1.

Furthermore, each data set is divided into training and testing subsets. The instances in the training subset have complete run-to-failure data, which can be applied to develop prognostics model, while the instances in the testing subset have up-to-date data and corresponding failure time data, which can be applied to validate the performance of prognostics model. In this paper, data set 2 is selected to evaluate the effectiveness of the proposed sensor selection scheme. Nevertheless, only the training subset is applied for validity assessment, for the testing instances with incomplete run-to-failure data are not suitable for performance evaluation metrics based on successive RUL estimations throughout the whole life. Hence, the first 200 training instances will be applied for training, and 30 out of the 60 remaining training instances will be selected randomly for testing.

#### 4.2. Performance Metrics

In the context of prognostics, the traditional accuracy-based or robustness-based performance metrics are inadequate to fairly assess the performance of prediction algorithms. Hence, four performance metrics proposed by Saxena et al. (2010) are adopted with minor modification. These metrics are on the basis of successive RUL prediction for each instance.

##### 1) Prediction horizon

Prediction horizon (PH) is defined as the RUL estimation that firstly satisfies the  $\alpha$ -bound criteria:

$$PH = t_E - t_\alpha \quad (14)$$

where  $t_E$  is the end-of-life time stamp, and  $t_\alpha$  represents the time stamp of the RUL estimation that firstly satisfies

the  $\alpha$ -bound criteria. At each time stamp  $t_i$ , the corresponding RUL estimation is  $r_i$ .  $t_\alpha$  is defined as:

$$t_\alpha = \min \{ t_i | t_i \in [t_s, t_f], r_i^* - \alpha \cdot t_E \leq r_i \leq r_i^* + \alpha \cdot t_E \} \quad (15)$$

where  $t_s$  is the start time of the RUL estimation, and  $t_f$  denotes the end time of the RUL estimation.

##### 2) Rate of acceptable predictions

Rate of acceptable predictions (AP) is defined as the rate of predictions that fall into an acceptable cone-shape area when  $t_i \geq t_h$ :

$$AP = \text{Mean}(\{ \delta_i | t_h \leq t_i \leq t_f \})$$

$$\delta_i = \begin{cases} 1, & \text{if } (1-\alpha)r_i^* \leq r_i \leq (1+\alpha)r_i^* \\ 0, & \text{otherwise} \end{cases} \quad (16)$$

where  $t_h$  is chose as PH calculated above. Obviously, AP is a stricter metric than the prediction errors.

##### 3) Relative accuracy

Relative accuracy (RA) is defined as the mean absolute percentage errors for all  $t_i \geq t_h$ :

$$RA = 1 - \text{Mean}(\{ \frac{|r_i - r_i^*|}{r_i^*} | t_h \leq t_i \leq t_f \}) \quad (17)$$

RA can give a quantitative metric of the prediction accuracy within the specified RUL, comparing with AR.

##### 4) Convergence

Convergence (CG) evaluate how fast the prediction performance improves when more historical data is available:

$$CG = 1 - \left( \frac{\frac{1}{2} \sum_{i=s}^f (t_{i+1}^2 - t_i^2) E_i}{\sum_{i=s}^f (t_{i+1} - t_i) E_i} - t_s \right) \cdot \frac{1}{t_f - t_s} \quad (18)$$

where performance metric  $E_i = r_i - r_i^*$ . The value of CG is between 0 and 1,  $CG > 0.5$  indicates convergence for the prediction.

##### 5) Performance evaluation

To assess the prediction performance based on multiple series from  $K$  testing instances, the median of four performance metrics is used:

$$\begin{aligned} PH &= \text{Median}(\{ {}^k PH \}_K) \\ AP &= \text{Median}(\{ {}^k AP \}_K) \\ RA &= \text{Median}(\{ {}^k RA \}_K) \\ CG &= \text{Median}(\{ {}^k CG \}_K) \end{aligned} \quad (19)$$



Within the four metrics, PH represents the size of time interval while the others have a value between 0 and 1 (1 means perfect).

### 4.3. Training Stage

From visual inspection, the sensor data in the training instances exhibit no prominent trend, as shown in Fig.2. Hence, k-means clustering algorithm is employed to cluster the 3 operation conditions of all the 200 training instances, and the operation conditions are clustered in 6 discrete operating regimes, labeled by an ID from 1 to 6. By this way, the sensor data from each regime may exhibit a rising trend, as shown in Fig.3

On the basis of the empirical observation mentioned in section 2.2, the 9 sensors with a consistent monotonic degradation trend under all of the 6 operating regimes are selected for further processing, namely #2, #3, #4, #7, #11, #12, #15, #20 and #21. For instance, the readings of sensor #2 from all the training instances under regime 1 are illustrated in Fig.4.

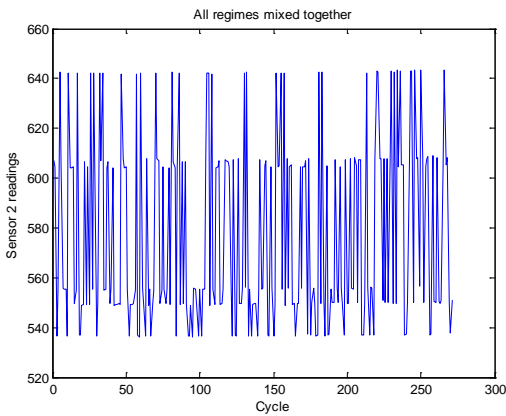


Figure 2. Raw data of sensor 2 from one training instance under all regimes

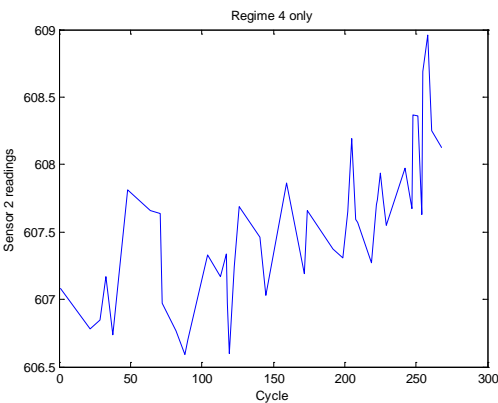


Figure 3. Raw data of sensor 2 from one training instance under regime 4 only

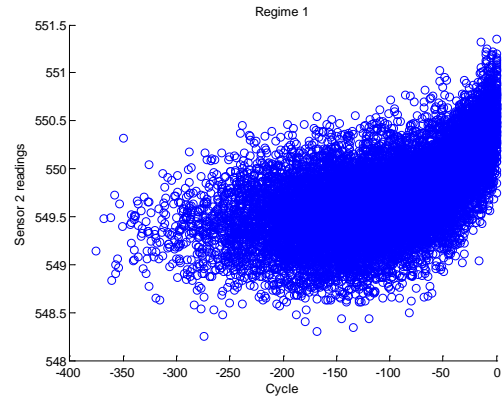


Figure 4. Raw data of sensor 2 from all the training instances under regime 1

Since the sensors have been preliminary selected, HF time series can be obtained through data fusion. It is remarkable that the selected sensors exhibit a consistent monotonic trend under different operating regimes, and a linear model can fit well for data with consistent trend. Hence, a linear regression model is adopted to convert the multi-dimensional sensor data into HF:

$$z = \alpha + \beta^T \cdot \mathbf{x} + \varepsilon = \alpha + \sum_{i=1}^9 \beta_i x_i + \varepsilon \quad (20)$$

where  $\mathbf{x}$  represents the selected 9-dimension sensor data,  $z$  is the health factor, and  $\varepsilon$  is the noise term.

The sample set  $\Omega = \{(\mathbf{x}, z)\}$  is used to learn the linear model:

$$\Omega = \{(\mathbf{x}_i, 0) \mid t_i > T_2\} \cup \{(\mathbf{x}_i, 1) \mid t_i < T_1\} \quad (21)$$

In this paper, the thresholds are set as  $T_1 = -t_E * 90\%$  and  $T_2 = -t_E * 10\%$ , where  $t_E$  represents the whole lifespan of the instance. Finally, 162 HF time series are extracted from all the 200 training instances, one of the time series is showed in Fig. 5.

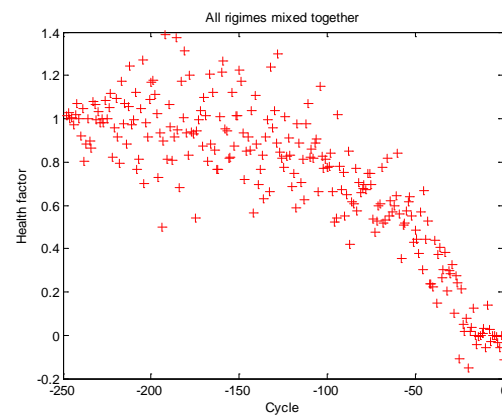


Figure 5. HF time series extracted from one training instance

Ranking	#203	#205	#206	#209
1	1(1)	1(1)	1(1)	1(1)
2	0.73636(6)	0.74473(6)	0.76490(8)	0.78437(8)
3	0.72580(8)	0.74145(8)	0.76392(6)	0.77928(6)
4	0.72313(3)	0.73965(3)	0.75887(3)	0.77731(2)
5	0.72121(2)	0.73890(2)	0.75671(2)	0.77700(3)
6	0.71695(4)	0.73636(4)	0.75041(4)	0.77345(4)
7	0.55575(10)	0.56147(10)	0.55503(9)	0.66692(9)
8	0.55283(9)	0.55972(9)	0.55435(10)	0.66538(10)
9	0.54224(5)	0.55115(5)	0.54571(5)	0.65627(5)
10	0.54137(7)	0.55020(7)	0.54537(7)	0.65556(7)

Table 2. Computing results of correlation degrees in the selected 4 training instances

The HF time series and corresponding 9-dimension sensor time series are labeled by an ID from 1 to 10. The grey correlation degrees between 162 HF time series and their corresponding multivariate sensor time series are calculated, and the results are sorted in descending order. Among them, the computing results of correlation degrees from 4 training instances are shown in Table 2.

From Table 2, the maximal correlation degree equals to 1, namely HF time series has a perfect similarity with itself. The correlation degrees ranked from 2 to 6 are rather close to each other, and so are the correlation degrees ranked from 7 to 10, but the correlation degrees between these two subsets have an obvious difference in value. It means that the first 5 sensor time series have a greater impact on the system degradation behaviors than the latter 4. By now, the first 5 sensors, namely #2, #3, #4, #11 and #15, might be selected to further optimize the sensor selection. To verify the generality of grey correlation analysis, the statistical information related with correlation degrees are given in Table 3.

From Table 3, the correlation degrees of two sensors #15 and #11, are ranked from 2 to 3 in most cases, the between HF and each of the sensors in all training instances correlation degrees of the sensors #2 and #3, are always ranked from 4 to 5, and the correlation degree of sensor #4 is ranked as 6 in most cases. Meanwhile, the correlation degrees of the first 5 sensors are rather close in value. Moreover, the correlation degrees of the remaining 4 sensors have never featured in the top 6 rankings, and the sensors ranked in the top 6 have an obviously higher coefficient degree than the sensor ranked in 7. Hence, the similarity between HF time series and the sensor time series can be well represented by the improved grey correlation degree, and the selected sensor subset includes 5 sensors, #2, #3, #4, #11 and #15. Furthermore, whether the selected optimal sensor subset can be further reduced remains to be discussed.

Ranking	#2	#3	#4	#7	#11	#12	#15	#20	#21
2	0	0	0	0	28	0	134	0	0
3	9	7	0	0	128	0	18	0	0
4	89	66	0	0	6	0	1	0	0
5	64	89	3	0	0	0	6	0	0
6	0	0	159	0	0	0	3	0	0
7	0	0	0	0	0	0	0	94	68
8	0	0	0	0	0	0	0	68	94
9	0	0	0	87	0	75	0	0	0
10	0	0	0	75	0	87	0	0	0

Table 3. Statistical information of correlation degrees

As seen in Fig.5, HF time series demonstrate an exponential degradation trend. Thus, the exponential regression models are adopted to describe the relationship between the HF  $z$  and operating time  $t$ :

$$z = a \cdot \exp(b \cdot t + c) + d + \sigma \quad (22)$$

where  $a$ ,  $b$ ,  $c$ , and  $d$  are the model parameters to be learned from HF time series, and  $\sigma$  is the noise term. The equation  $z=0$  indicates the failure of the instance, which is equal to the constraint  $a \cdot \exp(b \cdot t_E + c) + d = 0$ . After solving the parameter  $d$ , the following model form can be achieved:

$$z = a \cdot (\exp(b \cdot t + c) - \exp(b \cdot t_E + c)) + \sigma \quad (23)$$

The model library  $\{M_i\}$  can be constructed based on the 162 HF time series extracted from all the 200 training instances.

#### 4.4. Testing Stage

The sensor data in the testing instances should be converted into HF time series. For each testing instance, the selected sensor data will be clustered by operating regimes, and transformed by the linear regression models obtained during the training stage, and fused to obtain a HF time series.

In this paper, the similarity between the test instance and each of the degradation models are respectively evaluated by grey correlation degree and Euclidean distance, using Eq. (8) and (11). The final point estimation of RUL is obtained using Eq. (13).

### 5. PERFORMANCE EVALUATION

For each testing instance, multiple RUL estimations will be made at different time stamps along the life of the instance, and each RUL prediction is made based on the up-to-date data till the corresponding time. In this application, the start time of prediction  $t_s$  is set to 50; the end time of prediction  $t_f$  is set to  $t_E - 10$ ; the time interval of prediction is set to 5.

1) Impact of the sensor selection scheme on prediction performance

By empirical observation, the sensors with a consistent monotonic degradation trend under all of the 6 operating regimes are selected, namely #2, #3, #4, #7, #11, #12, #15, #20 and #21, which is represented by sensor subset 1. Furthermore, based on the grey correlation analysis, the optimal sensor subset 2 with a relatively large correlation degree is selected, and the subset includes 5 sensors, labeled by #2, #3, #4, #11 and #15. In this experiment, Euclidean distance is adopted to evaluate the similarity, and the thresholds of linear regression models are set as  $T_1=-t_E*90%$  and  $T_2=-t_E*10%$ . 30 testing instances are selected randomly to validate the impact of different sensor subsets on prediction performance, and the comparison results are described in Table 4.

As seen in Table 4, integrate with the IBL prognostics algorithm, the proposed sensor selection scheme improves the RUL prediction performance significantly. AP, RA and CG are improved by 25.9%, 4.3% and 1.6% while PH is nearly 10 cycles larger than before. Hence, the sensor selection scheme based on improved grey correlation analysis can effectively improve the RUL prediction performance. The sensor subset 2 will be adopted for data fusion in the subsequent experiments

2) Impact of different threshold settings for linear regression models on prediction performance

Not all the multivariate sensor data series can be converted into corresponding HF time series through linear regression. The number and parameters of linear regression models will vary with the different threshold settings. In this experiment, Euclidean distance is adopted to evaluate the similarity, and the thresholds of linear regression models are set to  $T_1=-t_E*90%$  and  $T_2=-t_E*10%$ ,  $T_1=-t_E*95%$  and  $T_2=-t_E*5%$ , and  $T_1=-240$  and  $T_2=-20$  respectively. 30 testing instances selected here is identical to that of the previous experiment, which are used to validate the impact of different threshold settings on prediction performance, and the comparison results are described in Table 5.

As seen in Table 5, when  $T_1=-t_E*90%$  and  $T_2=-t_E*10%$ , AP, RA and CG are much greater than those in the other two threshold settings, except that PH is relatively small. Moreover, it seems as if the setting of threshold parameters using a certain percentage of the total life of the instance can lead to a better prediction effect than the threshold parameters with fixed value. If hard threshold is applied, All the training instances whose lifetime are smaller than  $T_1$ , cannot be used to train the linear regression models, resulting in the decreasing of the model types in the model library.

Performance Metric	Subset 1	Subset 2
PH	134	144.5
AP	0.49515	0.62337
RA	0.78859	0.82270
CG	0.67914	0.68985

Table 4. Prediction performance of IBL prognostics algorithm under different sensor subsets

Performance Metric	$T_1=-t_E*90%$ $T_2=-t_E*10%$	$T_1=-t_E*95%$ $T_2=-t_E*5%$	$T_1=-240$ $T_2=-20$
PH	144.5	145	160.5
AP	0.62337	0.31743	0.29872
RA	0.82270	0.60669	0.58335
CG	0.68985	0.56431	0.53223

Table 5. Prediction performance of IBL prognostics algorithm under various thresholds

3) Impact of different similarity measurements on prediction performance

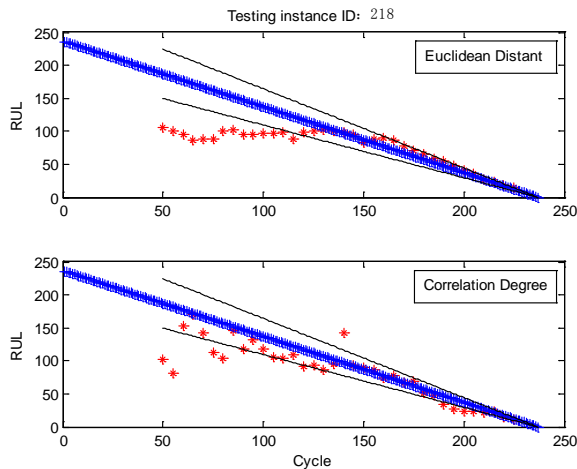
The definition of similarity between different instances has a great impact on the performance of IBL prognostics method. The purpose of this experiment is to figure out either Euclidean distance or grey correlation degree is a preferable similarity measurement. In the experiment, 30 testing instances which have been selected in the previous experiment are used to validate the impact of different similarity measures on prediction performance, and the comparison results are described in Table 6. RUL predictions of certain testing instances using different similarity measures are shown in Fig.6.

As seen in Table 6, combined in the IBL prognostics method, Euclidean distance is a preferable similarity measurement. AP, RA and CG are respectively improved by 19.3%, 8.8% and 3.2%, but PH is relatively smaller.

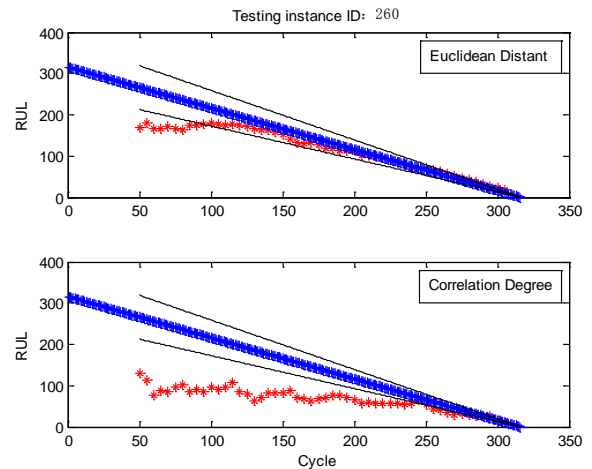
Above all, when sensor selection is executed based on grey correlation analysis, the threshold parameters are set to  $T_1=-t_E*90%$  and  $T_2=-t_E*10%$ , and Euclidean distance is chosen as the similarity measurement, the IBL algorithm can achieve an optimal RUL prediction performance.

Performance Metric	Euclidean Distance	Grey Correlation Degree
PH	144.5	161.5
AP	0.62337	0.52272
RA	0.82270	0.75639
CG	0.68985	0.63457

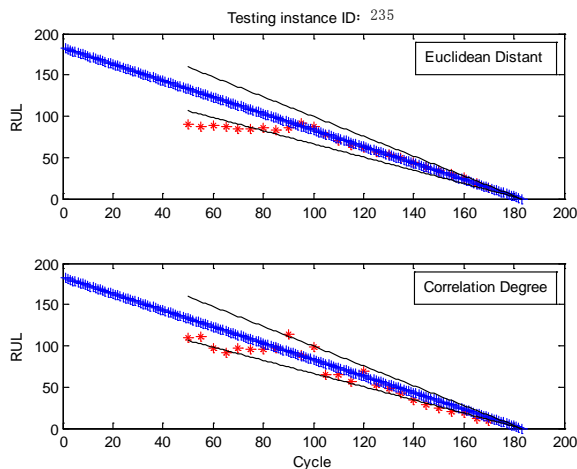
Table 6. Prediction performance of IBL prognostics algorithm under different similarity measurements



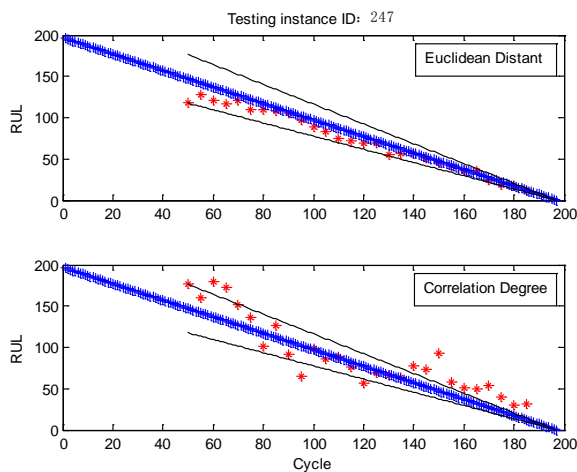
(a) Instance ID: 218



(d) Instance ID: 260



(b) Instance ID: 235



(c) Instance ID: 247

Figure 6. RUL predictions for selected instances using different similarity measures

From Fig.6, the selected instances show a desirable performance, where the RUL predictions converge to the true RUL as time increases. In the final stage, the RUL predictions are almost equal to the corresponding true RULs, indicating that the IBL algorithm has an excellent convergence and prediction effect in this application.

## 6. CONCLUSION

A grey correlation analysis method to selecting the most representative sensors for the RUL prediction of complex engineered systems is developed. The performance of sensor selection scheme integrated with IBL prognostics algorithm was evaluated using four performance metrics designed in the context of PHM. The addition of other sensors not selected by grey correlation analysis to the input sensor subset has led to a decreasing in the prediction performance, confirming the effectiveness of the sensor selection scheme. The scheme presented is expected to gain a similar performance for other complex engineered systems.

## REFERENCES

- Heng, A., Zhang, S., Tan, C. T., & Mathew, J. (2008). Rotating machinery prognostics: State of the art, challenges and opportunities. *Mechanical Systems and Signal Processing*, vol 23, pp. 724-739. doi:10.1016/j.ymssp.2008.06.009
- Cheng, S., Azarian, M. H., & Pecht, M. G. (2010). Sensor systems for prognostics and health management. *Sensors*, vol 10, pp 5774-5797, doi:10.3390/s100605774
- Coble, J. B., (2010). *Merging Data Sources to Predict Remaining Useful Life – An Automated Methods to Identify Prognostic Parameters*, Doctoral dissertation.

University of Tennessee, Knoxville, USA.  
[http://trace.tennessee.edu/utk\\_graddis/683](http://trace.tennessee.edu/utk_graddis/683)

- Zhang, G. F., (2005). *Optimal Sensor Localization/Selection in A Diagnostic/Prognostic Architecture*, Doctoral dissertation. Georgia Institute of Technology, Atlanta, USA.
- Xue, F., Bonissone, P., Varma, A., Yan, W. Z., & Goebel, K. (2008). An Instance-based method for remaining useful life estimation for aircraft engines. *Journal of Failure Analysis and Prevention*, vol 8, pp. 199-206. doi:10.1007/s11668-008-9118-9
- Wang, T. Y., Yu, J. B., Siegel, D., & Lee, J. (2008). A similarity-based prognostics approach for remaining useful life estimation of engineered systems. *International Conference on Prognostics and Health Management*. October 6-9, Denver, CO. doi:10.1109/PHM.2008.4711421
- Zhang, Y. J., & Zhang, X. (2007). Grey correlation analysis between strength of slag cement and particle fractions of slag powder. *Cement and Concrete Composites*, vol 29, pp. 498-504. doi:10.1016/j.cemconcomp.2007.02.004
- Saxena, A., & Goebel, K. (2008). C-MAPSS data set. *NASA Ames Prognostics Data Repository*. <http://ti.arc.nasa.gov/project/prognostic-data-repository>
- Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2010). Metrics for offline evaluation of prognostic performance. *International Journal of Prognostics and Health Management*, vol 1.

## BIOGRAPHIES



recognition, data mining and wireless sensor networks.

**Peng Yu** was born in 1973. He received the Ph.D. in instrument science and technology from Harbin Institute of Technology. Since 2007, he has been a professor in the Department of Instrument science and technology, Harbin Institute of Technology. His research interests include pattern



**Xu Yong** was born in 1985. He is currently working toward the Ph.D. degree from Harbin Institute of Technology. His research interests include wireless sensor networks and prognostics and health management of complex system.

# An Unscented Kalman Filter Based Approach for the Health-Monitoring and Prognostics of a Polymer Electrolyte Membrane Fuel Cell

Xian Zhang<sup>1</sup>, Pierluigi Pisu<sup>2</sup>

<sup>1,2</sup>*Department of Automotive Engineering, Clemson University, Clemson, SC, 29631, USA*

*xianz@g.clemson.edu*

*pisup@clemson.edu*

## ABSTRACT

Poor long-term performance and durability combined with high production and maintenance costs remain the main obstacles for the commercialization of the polymer electrolyte membrane fuel cell (PEMFC). While on-line diagnosis and operating condition optimization play an important role in addressing the durability issue of the fuel cell, health-monitoring and prognosis (or PHM) techniques are of equally great significance in terms of scheduling condition-based maintenance (CBM) to minimize repair and maintenance costs, the associated operational disruptions, and also the risk of unscheduled downtime for the fuel cell systems.

In this paper, an unscented Kalman filter (UKF) approach is proposed for the purpose of damage tracking and remaining useful life (RUL) prediction of a PEMFC. To implement this model-based PHM framework, a physics-based, prognostic-oriented catalyst degradation model is developed to characterize the fuel cell damage that establishes the relationship between the operating conditions and the degradation rate of the electro-chemical surface area (ECSA). The model complexity is kept minimal for on-line prognostic purpose. Simulation is carried out for validation of the proposed algorithm, using a more detailed catalyst degradation model.

## 1. INTRODUCTION

To date long-term performance and durability of the fuel cells are difficult to quantify because not all degradation mechanisms of the various fuel cell components are completely understood. The fuel cell's performance degrades irreversibly throughout its lifetime mainly due to the following components' degradations: 1) catalyst

degradation (catalyst particle coarsening); 2) carbon support degradation (carbon corrosion); and 3) membrane degradation (chemical deterioration and dehydration) (Okada, 2003; Schmittinger & Vahidi, 2008). Depending on the power load among other long-term operating conditions of the fuel cell, the extent of performance and durability degradation varies. General speaking, the longer the fuel cell stack is operated in transient or cycling conditions, or detrimental operating conditions such as flooding, the stronger is the corrosion and therefore the degradation. While on-line diagnosis and operating condition optimization play an important role in addressing the durability issue of the fuel cell, prognostics and health-monitoring (PHM) techniques are of equally great significance in terms of scheduling condition-based maintenance (CBM) to minimize repair and maintenance costs, the associated operational disruptions, and also the risk of unscheduled downtime for the fuel cell systems. However, so far little work has been done in the field of prognostics on fuel cell.

Bayesian based filtering techniques have found more and more use in the prognostic area due to their systematic and elegant treatment of the uncertainty management problem (Daigle, Saha, & Goebel, 2012; Saha & Goebel, 2008; Saha, Goebel, Poll, & Christophersen, 2007, 2009). Among these techniques, particle filtering (PF) has by far gained the most interest for its ability to handle arbitrary distributions and nonlinearities, and ease of implementation. Orchard et al. (M. E Orchard & Vachtsevanos, 2009; M. Orchard, Wu, & Vachtsevanos, 2005; Marcos E. Orchard & Vachtsevanos, 2007) proposed a novel particle filtering based prognostic framework. This approach employs a state dynamic model and a measurement model to predict the posterior probability density function of the state, that is, to predict the time evolution of a fault or fatigue damage. It avoids the linearity and Gaussian noise assumption of Kalman filtering and provides a robust framework for long-term prognosis while accounting effectively for uncertainties. Correction

Xian Zhang et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

terms are estimated in a learning paradigm to improve the accuracy and precision of the algorithm for long-term prediction. The approach is applied to a crack fault, and the results support its robustness and superiority.

Unscented Kalman filter (UKF), on the other hand, can also deal with nonlinearities in the system dynamics by approximating the state probability distribution with deterministic sigma points. While Gaussian distribution has to be assumed for the states and the noises, the UKF method brings great advantages of computational efficiency. In the case study provided in (Daigle et al., 2012), the UKF based prognostics approach outperforms that based on PF both in terms of computational cost and the accuracy measured by  $\alpha - \lambda$  metrics, thus is considered to be a promising on-line prognostic tool.

This paper investigates the application of a UKF-based PHM scheme on the prognostics of a PEMFC. The paper is organized as follows: in Section 2, the UKF based framework for PHM is introduced; Section 3 presents a prognostic-oriented catalyst degradation model for the fuel cell; and in Section 4 we apply the UKF approach to the damage variable tracking and long term damage variable prediction of the fuel cell, as well as the RUL prediction and related performance metrics. The conclusion is presented finally in Section 5.

## 2. UKF-BASED FRAMEWORK FOR PHM

### 2.1. Bayesian Framework for Joint Estimation

We first consider a general joint estimation problem based on the following discrete system.

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{F}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k, \mathbf{w}_k) \\ \mathbf{y}_k &= \mathbf{H}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{n}_k, \mathbf{w}_k) \end{aligned} \quad (1)$$

where  $\mathbf{x}_k$  represent the states of the system,  $\mathbf{y}_k$  the outputs,  $\mathbf{u}_k$  the inputs,  $\mathbf{v}_k, \mathbf{n}_k$  the process and measurement noises, respectively, and  $\mathbf{w}_k$  the time-varying system parameters.

Since the parameters  $\mathbf{w}$  are unknown and time-varying, the state and parameter must be simultaneously/jointly estimated based on the noisy measured output. On the other hand, the dynamics of the time-varying system parameters are usually hard to describe, i.e., there is a lack of descriptive dynamic equation to characterize the parameters. A commonly used method to address this issue is to treat the parameter as a stochastic signal driven by a white noise ( $\mathbf{r}_k$ )

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mathbf{r}_k \quad (2)$$

Then, by concatenating the state and parameter to form an augmented state vector  $\mathbf{x}_k^a = [\mathbf{x}_k^T \quad \mathbf{w}_k^T]^T$ , joint state space

equations (Wan & van der Merwe, 2002) (assuming additive noises) can be obtained as follows

$$\begin{aligned} \begin{bmatrix} \mathbf{x}_{k+1} \\ \mathbf{w}_{k+1} \end{bmatrix} &= \begin{bmatrix} \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) \\ \mathbf{I} \cdot \mathbf{w}_k \end{bmatrix} + \begin{bmatrix} \mathbf{B} \cdot \mathbf{v}_k \\ \mathbf{r}_k \end{bmatrix} = \mathbf{F}(\mathbf{x}_k^a, \mathbf{u}_k) + \begin{bmatrix} \mathbf{B} \cdot \mathbf{v}_k \\ \mathbf{r}_k \end{bmatrix} \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) + \mathbf{n}_k = \mathbf{H}(\mathbf{x}_k^a, \mathbf{u}_k) + \mathbf{n}_k \end{aligned} \quad (3)$$

The joint estimation problem of states and parameters based on observation can be formulated in an optimal recursive estimation framework as given in the following equation

$$\hat{\mathbf{x}}_k^a = E[\mathbf{x}_k^a | \mathbf{Y}_k] = E[\mathbf{x}_k^a | \mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_k] \quad (4)$$

Two step process (recursively) are involved, the first is the measurement correction

$$p(\mathbf{x}_k^a | \mathbf{Y}_k) = \frac{p(\mathbf{x}_k^a | \mathbf{Y}_{k-1}) p(\mathbf{y}_k | \mathbf{x}_k^a)}{p(\mathbf{y}_k | \mathbf{Y}_{k-1})} \quad (5)$$

And the second is the one-step prediction

$$p(\mathbf{x}_{k+1}^a | \mathbf{Y}_k) = \int p(\mathbf{x}_{k+1}^a | \mathbf{x}_k^a) p(\mathbf{x}_k^a | \mathbf{Y}_k) d\mathbf{x}_k^a \quad (6)$$

Various filtering techniques can be implemented in this general recursive estimation framework, including the most widely used extended Kalman filter (EKF), particle filtering (PF), and unscented Kalman filter (UKF). EKF is difficult to tune, and the Jacobian is usually hard to derive, and it can only handle limited amount of nonlinearity, while PF can handle arbitrary distributions and nonlinearities but is computationally very complex. In this paper, we focus on UKF since we believe gives the best tradeoff between PF and EKF.

### 2.2. UKF Implementation

We assume the additive (zero mean) noise case and follow the UKF procedure given in (Wan & van der Merwe, 2002).

First, the augmented state estimation and covariance matrix are initialized with (7).

$$\begin{aligned} \hat{\mathbf{x}}_0^a &= E[\mathbf{x}_0^a] \\ \mathbf{P}_0^a &= E\left[(\mathbf{x}_0^a - \hat{\mathbf{x}}_0^a)(\mathbf{x}_0^a - \hat{\mathbf{x}}_0^a)^T\right] \end{aligned} \quad (7)$$

Then, for each iteration ( $k=1,2,\dots$ ), the sigma points for the state variables in last step are obtained and combined into a matrix as follows

$$\mathbf{X}_{k-1} = \begin{bmatrix} \hat{\mathbf{x}}_{k-1}^a & \hat{\mathbf{x}}_{k-1}^a + c\sqrt{\mathbf{P}_{k-1}^a} & \hat{\mathbf{x}}_{k-1}^a - c\sqrt{\mathbf{P}_{k-1}^a} \end{bmatrix} \quad (8)$$

where  $\lambda = \alpha^2(L + \kappa) - L$ ,  $c = L + \lambda$ , and  $\alpha, \kappa$  and  $\beta$  are all tunable parameters (in this paper,  $\alpha = 10^{-3}$ ,  $\kappa = 0$ ,  $\beta = 2$  is chosen according to (Wan & van der Merwe, 2002)).

These sigma points are then fed to the state equation to generate a set of new sigma points for the state variables in the current step:  $\mathbf{X}_{k|k-1}^{(i)} = \mathbf{F}(\mathbf{X}_{k-1}^{(i)}, \mathbf{u}_k)$ , where the superscript  $(i)$  denotes the  $i$ -th column of the corresponding matrix, i.e., the  $i$ -th sigma point.

The one-step prediction for the augmented state vector in (6), when implemented with UKF approach, can now be expressed in (9)

$$\begin{aligned}\hat{\mathbf{x}}_k^{a-} &= \sum_{i=0}^{2L} w_i^m \cdot \mathbf{X}_{k|k-1}^{(i)} \\ \mathbf{P}_k^{a-} &= \sum_{i=0}^{2L} w_i^c \cdot \left[ \mathbf{X}_{k|k-1}^{(i)} - \hat{\mathbf{x}}_k^{a-} \right] \left[ \mathbf{X}_{k|k-1}^{(i)} - \hat{\mathbf{x}}_k^{a-} \right]^T + \mathbf{Q}\end{aligned}\quad (9)$$

where  $\mathbf{Q}$  is the process noise covariance matrix, and  $w_i^m, w_i^c$  are the weights for the corresponding sigma points.

Measurement correction, on the other hand, is given through (10) – (16)

$$\mathbf{Y}_{k|k-1}^{(i)} = \mathbf{H}(\mathbf{X}_{k|k-1}^{(i)}, \mathbf{u}_k) \quad (10)$$

$$\hat{\mathbf{y}}_k^- = \sum_{i=0}^{2L} w_i^m \cdot \mathbf{Y}_{k|k-1}^{(i)} \quad (11)$$

$$\mathbf{P}_{\bar{\mathbf{y}}_k \bar{\mathbf{y}}_k} = \sum_{i=0}^{2L} w_i^c \cdot \left[ \mathbf{Y}_{k|k-1}^{(i)} - \hat{\mathbf{y}}_k^- \right] \left[ \mathbf{Y}_{k|k-1}^{(i)} - \hat{\mathbf{y}}_k^- \right]^T + \mathbf{R} \quad (12)$$

$$\mathbf{P}_{\bar{\mathbf{x}}_k \bar{\mathbf{y}}_k} = \sum_{i=0}^{2L} w_i^c \cdot \left[ \mathbf{X}_{k|k-1}^{(i)} - \hat{\mathbf{x}}_k^{a-} \right] \left[ \mathbf{Y}_{k|k-1}^{(i)} - \hat{\mathbf{y}}_k^- \right]^T \quad (13)$$

$$\mathbf{K}_k = \mathbf{P}_{\bar{\mathbf{x}}_k \bar{\mathbf{y}}_k} \mathbf{P}_{\bar{\mathbf{y}}_k \bar{\mathbf{y}}_k}^{-1} \quad (14)$$

$$\hat{\mathbf{x}}_k^a = \hat{\mathbf{x}}_k^{a-} + \mathbf{K}_k (\mathbf{y}_k - \hat{\mathbf{y}}_k^-) \quad (15)$$

$$\mathbf{P}_k^a = \mathbf{P}_k^{a-} - \mathbf{K}_k \mathbf{P}_{\bar{\mathbf{x}}_k \bar{\mathbf{y}}_k} \mathbf{K}_k^T \quad (16)$$

where  $\mathbf{R}$  is the measurement noise covariance matrix. The weights typically employed in UKF are given as follows:

$$w_0^m = \lambda/c$$

$$w_0^c = \lambda/c + (1 - \alpha^2 + \beta)$$

$$w_i^m = w_i^c = \lambda/2c$$

### 2.3. UKF Approach for Prediction

The Bayesian method is an iterative method that involves two steps in each iteration, i.e., prediction and measurement correction. In the general Bayesian estimation framework, the one-step prediction in (6) can be extended to  $(m+1)$ -step long-term prediction as follows,

$$\begin{aligned}p(\mathbf{x}_{k+m+1}^a | \mathbf{Y}_k) &= \int p(\mathbf{x}_{k+m+1}^a | \mathbf{x}_{k+m}^a) p(\mathbf{x}_{k+m}^a | \mathbf{Y}_k) d\mathbf{x}_{k+m}^a \\ &= \int p(\mathbf{x}_{k+m+1}^a | \mathbf{x}_{k+m}^a) \left[ \int p(\mathbf{x}_{k+m}^a | \mathbf{x}_{k+m-1}^a) p(\mathbf{x}_{k+m-1}^a | \mathbf{Y}_k) d\mathbf{x}_{k+m-1}^a \right] d\mathbf{x}_{k+m}^a \\ &= \iint p(\mathbf{x}_{k+m+1}^a | \mathbf{x}_{k+m}^a) p(\mathbf{x}_{k+m}^a | \mathbf{x}_{k+m-1}^a) p(\mathbf{x}_{k+m-1}^a | \mathbf{Y}_k) d\mathbf{x}_{k+m-1}^a d\mathbf{x}_{k+m}^a \\ &= \int \cdots \int p(\mathbf{x}_k^a | \mathbf{Y}_k) \prod_{j=k}^{k+m} p(\mathbf{x}_{j+1}^a | \mathbf{x}_j^a) d\mathbf{x}_k^a \cdots d\mathbf{x}_{k+m}^a\end{aligned}\quad (17)$$

In this case of  $(m+1)$ -step long-term prediction, when implemented with UKF approach, the equations (9) can be written as (18),

$$\begin{aligned}\mathbf{X}_{n|n-1}^{(i)} &= \mathbf{F}(\mathbf{X}_{k-1}^{(i)}, \mathbf{u}_n) \\ \hat{\mathbf{x}}_n^a &= \sum_{i=0}^{2L} w_i^m \cdot \mathbf{X}_{n|n-1}^{(i)} \\ \mathbf{P}_n^a &= \sum_{i=0}^{2L} w_i^c \cdot \left[ \mathbf{X}_{n|n-1}^{(i)} - \hat{\mathbf{x}}_n^a \right] \left[ \mathbf{X}_{n|n-1}^{(i)} - \hat{\mathbf{x}}_n^a \right]^T + \mathbf{Q}\end{aligned}\quad (18)$$

As can be seen from (18), the long-term prediction utilizes only the one-step prediction iteratively without measurement correction, since the future measurement is unavailable at the current step when prediction is made. Equivalently, the standard UKF procedure can be used with the Kalman gain in (15) set to 0, i.e.,  $\mathbf{K}_k = \mathbf{0}$ .

Also note that in (18), the system input  $\mathbf{u}_n$  at step  $n$  ( $n > k$ ) is required to generate the new set of sigma points at each iteration. Generally speaking, the future input can be obtained by analyzing its past stochastic feature and then projecting into the future, while at the same time including the uncertainty of the input itself. For simplicity purpose, in this paper, the current input  $\mathbf{u}_k$  is taken as the constant input for all future steps during the prediction, and the uncertainty issue is not considered.

### 3. AGING MODEL FOR CATALYST DEGRADATION

Aging model is essential to the physics-based prognostic and health monitoring approaches. While extensive researches have been carried out on control and dynamical modeling for fuel cell system, modeling work addressing PEMFC degradation and corresponding health-monitoring and prognostic system has been much less reported. Many of the degradation models have little physical basis, and thus have no predictive capability (A.A. Franco & Tembely, 2007). Franco et al. (A. A. Franco, Schott, Jallut, & Maschke, 2007; Alejandro A. Franco et al., 2009; Alejandro A. Franco & Gerard, 2008) have done a series of leading work on developing a multi-scale mechanistic model of the electrochemical aging processes in a PEMFC to describe, in particular, the carbon corrosion at the cathode, the cathodic oxidation/dissolution of platinum and the carbon supported platinum electrochemical ripening. However, its CFD modeling approach makes it unsuitable for the on-line



prognostic purpose due to the computation burden involved. Also, the model needs too many parameters that are hard to obtain. Darling and Meyers (R.M. Darling & Meyers, 2003) proposed a spatially lumped model that treats a single, porous platinum electrode and the ionomeric solution that fills the pores of the electrode. The model includes spherical platinum particles that can grow and shrink as platinum plates and dissolves; a platinum oxide layer; and an ionic platinum species in solution ( $\text{Pt}^{2+}$ ). The kinetic expressions for platinum oxidation and dissolution developed in this work is incorporated by the same authors (Robert M. Darling & Meyers, 2005) in a transient, one-dimensional mathematical model of the cross section of a PEM fuel cell. In this model, each electrode contains two platinum particle sizes, enabling a description of electrochemically driven transfer of platinum between particles of different sizes. That is, platinum can be exchanged between particles by dissolution and crystallization, capturing the underlying principles of the quasi-Ostwald ripening. Based on their work, we proposed a 64-particle catalyst degradation model (Xian, 2012) and, for the prognostic purpose, simplified the detailed model to a second order aging model. For simplicity of demonstration, in this paper, we further simplify that model to a first order dynamic model, as shown in (19), by neglecting the dynamics of the platinum oxide coverage during the load cycling. The geometric catalytic surface area  $A_{\text{geo}}$  can then be described as follows

$$\frac{dA_{\text{geo}}}{dt} = -\frac{k_1}{V_{\text{Pt}}^2} \frac{M_{\text{Pt}}}{\rho_{\text{Pt}}} \frac{F\alpha_1}{RT} \beta_r^2 \cdot \exp\left(\frac{F\alpha_1}{RT} \frac{A_{\text{geo}}}{3V_{\text{Pt}}}\right) \cdot A_{\text{geo}}^3 \eta(\Delta\phi_c) \quad (19)$$

where  $k_1$  is a constant characteristic of the kinetic rate of the reaction, and is usually determined by experiments;  $V_{\text{Pt}}$  is the total volume of the Pt catalyst;  $F$  is the Faraday constant,  $R$  is the universal gas constant (J/mol K) and  $T$  the absolute temperature (K);  $\beta_r$  is a system parameter to be identified and represents the relative distribution of the particles radii;  $\Delta\phi_c$  is the phase potential difference between the electrolyte phase and the cathode phase, and  $\eta(\bullet)$  is a function;  $M_{\text{Pt}}$  and  $\rho_{\text{Pt}}$  are the molecular weight and density of platinum, respectively;  $\alpha_1 = 1.14 \times 10^{-10}$ .

Since  $k_1$  is extremely small in value, we can scale the above equation in time with a scaling coefficient  $\varepsilon$  to denote its slowness explicitly. Also note that  $k_1$  is an implicit function of the temperature  $T$ , thus we can write the degradation rate of the catalytic surface area as a function of the input potential load, the temperature, the parameter  $\beta_r$ , and the catalytic surface area itself.

$$\frac{dA_{\text{geo}}}{dt} = \varepsilon \cdot \mathbf{g}(A_{\text{geo}}, \Delta\phi_c, \beta_r, T) \quad (20)$$

Now, by taking into account the uncertain factors as additive process noise, the following stochastic degradation model can be obtained

$$\begin{aligned} \frac{dA_{\text{geo}}}{dt} &= \varepsilon \cdot \mathbf{g}(A_{\text{geo}}, \Delta\phi_c, \beta_r, T) + w_1(t) \\ \dot{\beta}_r &= w_2(t) \end{aligned} \quad (21)$$

where the unknown time-varying parameter  $\beta_r$  is taken as a state variable, and its derivative as a process noise.

The cell voltage given in (22) is utilized as the measured system output,

$$\begin{aligned} V_{\text{cell}} &= E_0(T_{\text{fc}}) - \frac{RT_{\text{fc}}}{4(1-\alpha)F} \ln(i + i_{\text{leak}}) + \frac{RT_{\text{fc}}}{4(1-\alpha)F} \ln \xi_{\text{cat}} \\ &+ \frac{RT_{\text{fc}}}{2F} \ln p_{\text{H}_2}^* + \frac{RT_{\text{fc}}}{4(1-\alpha)F} \ln p_{\text{O}_2}^* - i \cdot A_{\text{fc}} \cdot R_{\text{ohm}} \end{aligned} \quad (22)$$

where  $E_0(T_{\text{fc}})$  is the voltage component that only depends on the fuel cell temperature for a specific type of fuel cell;  $\alpha$  is the transfer coefficient of the oxygen reduction reaction;  $i$  and  $i_{\text{leak}}$  are the current density and the leak current density;  $p_{\text{H}_2}^*$ ,  $p_{\text{O}_2}^*$  are the hydrogen and oxygen partial pressures at the reaction site; and  $R_{\text{ohm}}$  is the total ohmic resistance of the fuel cell.  $\xi_{\text{cat}} = \frac{A_{\text{geo}}}{A_{\text{geo}}|_{t=0}}$  is the ratio of

current and initial catalytic geometric (or equivalently electrochemical) surface area, the value of which is 1 when the fuel cell is at its fresh state.

#### 4. APPLICATION OF THE UKF APPROACH FOR DAMAGE TRACKING AND PROGNOSTICS OF PEMFC

The performance degradation of the PEMFC, characterized by the voltage drop under the same level of current load, typically goes through two stages. In the first stage, the PEMFC experiences a gradual performance loss caused by the slowly decreasing ECSA that can be described by the degradation model (21) presented in the last section; in contrast, the second stage is characterized by a much more abrupt performance drop that leads to the total loss of the power output capability of the PEMFC, resulting from the mechanical breach (pinholes) formed in the polymer membrane. As it may seem obvious that the second stage, if detected, can be utilized to determine the end-of-life of the PEMFC, the gradual loss of the ECSA in the first stage can also lead to serious performance loss to the extent that failure of the system can be declared. In (Shimoi, Aoyama, & Iiyama, 2009), actual vehicle tests are carried out to collect the PEMFC stack durability data until the performance is deemed low. The post-test analysis of the

membrane-electrolyte-assembly (MEA) reveals no crack is developed in the membrane and suggests that the voltage drop is mainly attributed to the reduced ECSA at the cathode of the fuel cells. Therefore in this paper, as a first step, we only investigate the health monitoring and prognostic problem for the catalyst degradation.

#### 4.1. Data Preprocessing

From (22), it is obvious that the measured voltage (“feature”) is not only a function of the slowly varying damage variables, i.e.,  $i_{leak}$  and  $\xi_{cat}$ , but also a function of the fast varying system variables  $p_{H_2}^*, p_{O_2}^*$ . To obtain an output variable that is solely related to the slowly varying damage variables, i.e., the state variables in the aging model describing the degradation process, we propose the following data preprocessing system, as shown in Figure 1.

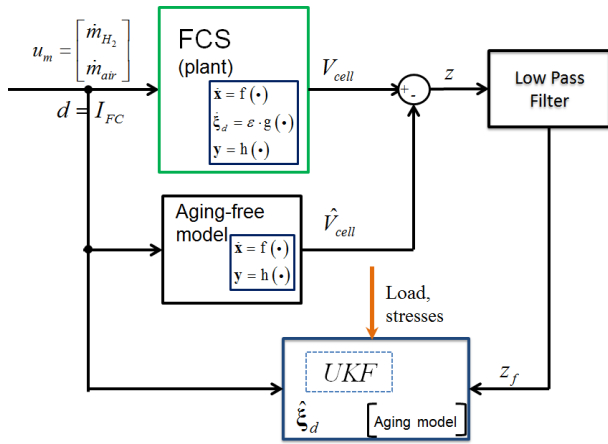


Figure 1 Preprocessing of the output signal for UKF

Basically, the method above utilizes an aging-free dynamic model of the fuel cell system that captures the fast dynamics of the FCS in the normal time scale while assuming constant, non-degrading damage variables in the long time scale. The model output is given in (23).

$$\begin{aligned} \hat{V}_{cell} = & E_0(T_{fc}) + \frac{RT_{fc}}{2F} \ln \hat{p}_{H_2}^* + \frac{RT_{fc}}{4(1-\alpha)F} \ln \hat{p}_{O_2}^* \\ & - i \cdot A_{fc} \cdot R_{ohm} - \frac{RT_{fc}}{4(1-\alpha)F} \ln(i + i_{leak}^0) \end{aligned} \quad (23)$$

where  $i_{leak}^0$  is the membrane leak current density at the initial time, and here we assume that the total ohmic resistance  $R_{ohm}$  is readily available.

The difference of the outputs of the actual FCS and the aging-free dynamic FCS model can then be calculated and written as follows.

$$\begin{aligned} V_{cell} - \hat{V}_{cell} = & \frac{RT_{fc}}{4(1-\alpha)F} \ln \xi_{cat} - \frac{RT_{fc}}{4(1-\alpha)F} \ln \left( \frac{i + i_{leak}}{i + i_{leak}^0} \right) \\ & + \left( \frac{RT_{fc}}{2F} \ln \frac{p_{H_2}^*}{\hat{p}_{H_2}^*} \right) + \left[ \frac{RT_{fc}}{4(1-\alpha)F} \ln \frac{p_{O_2}^*}{\hat{p}_{O_2}^*} \right] \end{aligned} \quad (24)$$

Ideally, given that the model is accurate in capturing the fast dynamics of the system and no disturbance is present, we have  $\frac{p_{H_2}^*}{\hat{p}_{H_2}^*} = 1, \frac{p_{O_2}^*}{\hat{p}_{O_2}^*} = 1$ , and the difference in (24) would depend only on the slowly varying damage variables. However, in practice, the instantaneous voltage degradation ( $V_{cell} - \hat{V}_{cell}$ ) would inevitably be affected by modeling error that may result in model prediction deviation especially in transition, and by disturbances such as the water content in the gas diffusion media that could even result in offset between the two outputs at their steady states. By denoting the two kinds of sources of mismatches with  $n_d$  and  $n_s$  respectively, the difference  $z$  in Figure 1 can be expressed as follows,

$$z = \frac{RT_{fc}}{4(1-\alpha)F} \ln \xi_{cat} - \frac{RT_{fc}}{4(1-\alpha)F} \ln \left( \frac{i + i_{leak}}{i + i_{leak}^0} \right) + n_d + n_s \quad (25)$$

The deviation between the model and the actual plant outputs is then fed to the low pass filter in Figure 1 to remove the fast dynamics mismatch  $n_d$  and to obtain the filtered deviation  $z_f$  as in (26).

$$z_f = \frac{RT_{fc}}{4(1-\alpha)F} \ln \xi_{cat} - \frac{RT_{fc}}{4(1-\alpha)F} \ln \left( \frac{i + i_{leak}}{i + i_{leak}^0} \right) + n_s \quad (26)$$

To apply the UKF approach introduced in the previous section, we need to first discretize the continuous system equations. By denoting the fixed sample time as  $\Delta T$ , the state equations can be written as

$$\begin{aligned} A_{geo,k+1} = & \epsilon \cdot N_\epsilon \cdot \Delta T \cdot \left[ \mathbf{g}(A_{geo,k}, \Delta\phi_{c,k}, \beta_{r,k}, T_{fc}) + w_{1,k} \right] \\ \beta_{r,k+1} = & \beta_{r,k} + \epsilon \cdot N_\epsilon \cdot \Delta T \cdot w_{2,k} \end{aligned} \quad (27)$$

and the output equation is given by (28)

$$z_{f,k} = \frac{RT_{fc}}{4(1-\alpha)F} \ln \xi_{cat,k} - \frac{RT_{fc}}{4(1-\alpha)F} \ln \left( \frac{i_k + i_{leak,k}}{i_k + i_{leak,k}^0} \right) + n_{s,k} \quad (28)$$

where the  $k$  represents the discretization time step. Note that, since the fuel cell catalyst degradation is an extremely slow process (the fresh to failure life cycle typically takes several hundred hours or longer), the sample time  $\Delta T$  for the discretized model employed by the UKF can be chosen in the hour scale, instead of second, to lower the unnecessary computational burden. In this paper, we choose  $\Delta T = 1500s$  (25 min). Under such a slow time scale, disturbances

induced offset between the two voltage outputs  $n_{s,k}$  can then be treated reasonably as a white noise.

As seen in Figure 1,  $z_{f,k}$  in (28) can be practically obtained as

$$z_{f,k} = \text{LPF}(V_{\text{cell},k} - \hat{V}_{\text{cell},k}) \quad (29)$$

where LPF stands for the low pass filter.

Simulation is carried out where we assume a cyclic load profile as given in Figure 2.

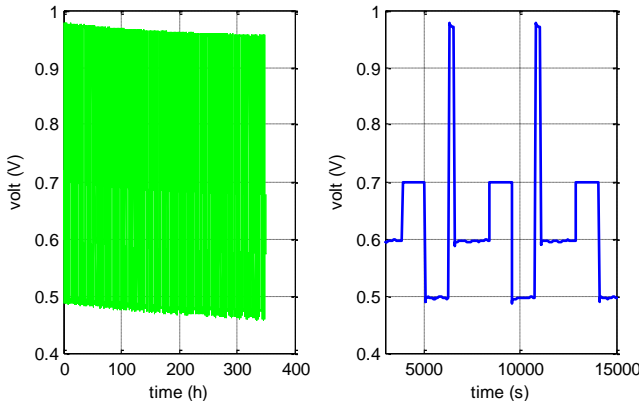


Figure 2 Load profile for PHM scheme simulation validation

#### 4.2. Simulation Results

The simulation result for health monitoring validation is presented in Figure 3. It can be seen that the ECSA has been tracked satisfactorily, while  $\beta_r$  varies between the range [0.03, 0.04], and gradually settles down after 100 hours. In fact, the first 100 hours can be considered as the necessary phase for parameter identification. After this phase, the parameter  $\beta_r$  would stay relatively stable around some constant value. This characteristic feature can also be exploited to perform fault diagnostics for the purpose of early detection of the massive gas crossover.

The parameters used for this simulation are grouped in the following table:

Initial state estimate	$\begin{bmatrix} \hat{A}_{Pt}^0 \\ \hat{\beta}_r^0 \end{bmatrix} = \begin{bmatrix} A_{Pt}^0 \times 1.1 \\ 0.035 \end{bmatrix}$
Initial covariance matrix	$P_0 = \begin{bmatrix} (0.02 \times A_{Pt}^0)^2 & 0 \\ 0 & (0.04 \times 0.15)^2 \end{bmatrix}$

Process noise	$\begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix} = \text{diag} \left( \begin{bmatrix} (A_{Pt}^0 \times 1 \times 10^{-6})^2 \\ (0.035 \times 2 \times 10^{-6})^2 \end{bmatrix} \right)$
Measurement noise	$R = 6.7 \times 10^{-6}$

Table 1 Parameters used in the simulation

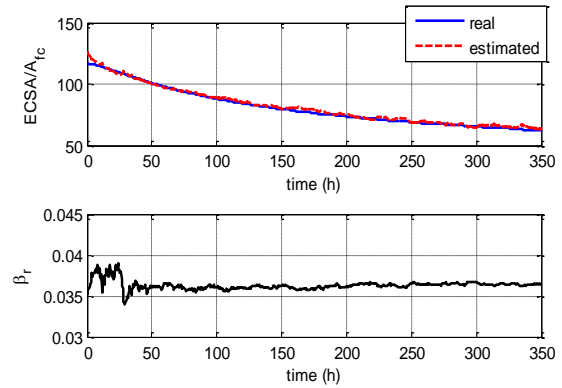


Figure 3 Simulation result for health monitoring

The simulation results for prognostics (long-term prediction of the damage variable) are presented in Figure 4 and Figure 5, at two different prediction time points respectively. For the purpose of demonstration, and without loss of generosity, the threshold for the catalyst failure is set to 63.

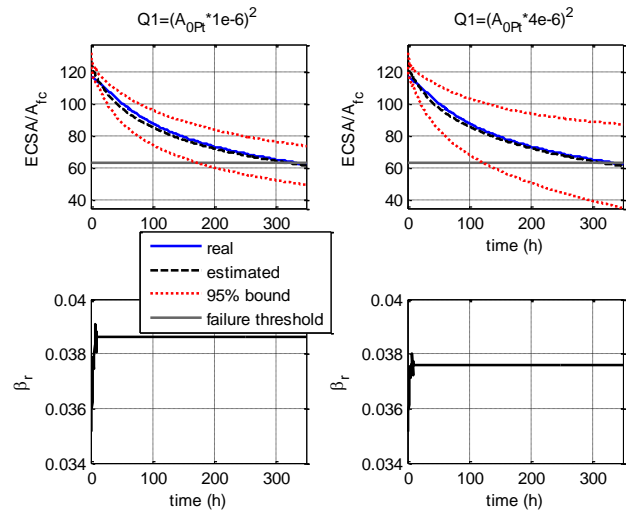


Figure 4 Simulation result for prognostics at the beginning of life (10 hours)

In Figure 4, prognostic is performed at the beginning of life of the fuel cell (10 hours). It can be seen that the 95% error is growing larger with increasing prediction step due to lack

of measurement correction.  $\beta_r$  is seen to be constant all the time with the same reason. Two values for the process noise  $Q_1$  are compared, i.e.,  $(A_{Pt}^0 \times 1 \times 10^{-6})^2$  and  $(A_{Pt}^0 \times 4 \times 10^{-6})^2$ . As expected, the smaller value of the process noise results in narrower 95% bounds, and thus a more accurate prediction.

Figure 5 shows the simulation results with prognostic starting from the middle of the durability test, here at 100 hours. Due to lack of measurement correction,  $\beta_r$  stops updating and is taken as a constant after 100 hours. The 95% error is also seen growing larger with increasing prediction step, but much narrower compared to that in Figure 4.

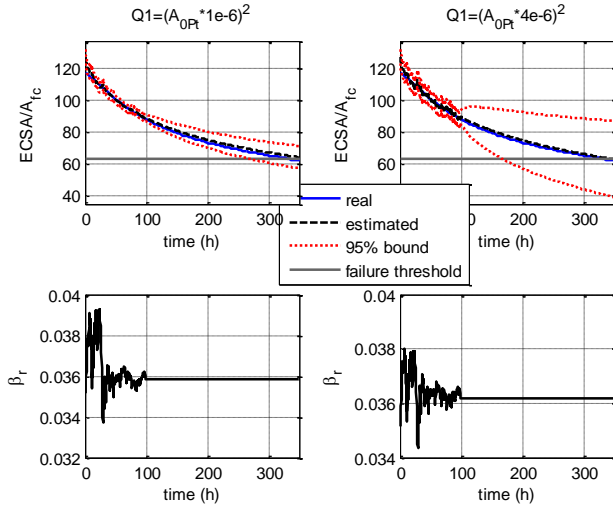


Figure 5 Simulation result for Prognostics at 100h

### 4.3. Performance Evaluation

To apply any prognostics metrics (Saxena et al., 2008; Saxena, Celaya, Saha, Saha, & Goebel, 2009) for the performance evaluation of the proposed algorithm, it is necessary to first obtain the remaining useful life (RUL) prediction w.r.t. the prediction time  $t_{pred}$  based on the information contained in the long-term prediction of the damage variable. By denoting the threshold for the normalized catalytic area to be  $\xi_{th}$ , and the RUL (stochastic variable at  $t_{pred}$ ) to be  $T_{RUL}$ , we can establish the following connection between the probability functions of the damage variable and the RUL

$$\begin{aligned} P\left(\frac{ECSPA}{A_{fc}} < \xi_{th} \mid t\right) \Big|_{t_{pred}} &= P(\text{fail} \mid t) \\ &= P(T_{RUL} \leq t) \Big|_{t_{pred}} = F_{RUL}(t) \Big|_{t_{pred}} \end{aligned} \quad (30)$$

where  $t > t_{pred}$  is any future time,  $F_{RUL}(t)$  is the cumulative probability function of the RUL, with which the RUL probability density function can be derived as

$$f_{RUL}(t) = \frac{\partial F_{RUL}(t)}{\partial t} \quad (31)$$

Here we employ a metrics called  $\alpha$  performance, which is similar to the  $\alpha - \lambda$  performance proposed in (Saxena et al., 2008) except that the time window is not modified along the prediction time axis. The  $\alpha$  performance basically evaluates the probability of the RUL  $T_{RUL}$  falling in the time window  $[(1-\alpha)t_{EOL}, (1+\alpha)t_{EOL}]$  defined by the parameter  $\alpha$ . The probability can be calculated as follows

$$\begin{aligned} P_\alpha &= \int_{(1-\alpha)t_{EOL}}^{(1+\alpha)t_{EOL}} f_{RUL}(t) dt \\ &= F_{RUL}[(1+\alpha)t_{EOL}] - F_{RUL}[(1-\alpha)t_{EOL}] \end{aligned} \quad (32)$$

Figure 6 presents the RUL prediction at selected prediction time, with the following process noise (other simulation parameters are the same as that in Table 1) assumed for the dynamic aging process.  $\alpha$  is chosen to be 15% in this case.

$$\begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix} = \text{diag} \left( \begin{bmatrix} (A_{Pt}^0 \times 8 \times 10^{-7})^2 \\ (0.035 \times 1 \times 10^{-6})^2 \end{bmatrix} \right)$$

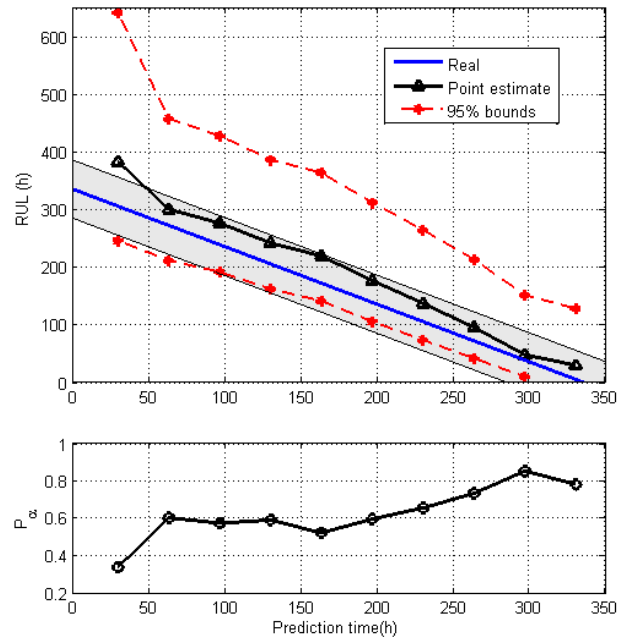


Figure 6 RUL prediction plot ( $\alpha = 15\%$ )

As can be seen from Figure 6, the point estimate of the RUL is contained in the time window (the shaded area in the figure) for the majority of the prognostic process. The

prognostic process is a dynamic process in that the RUL prediction is updated dynamically as the prediction time proceeds. It can also be seen that, with a constant  $\alpha$  time window, the probability  $P_\alpha$  shows an increasing trend along with the prognostic process, indicating improved accuracy of the prediction.

It is noted that  $\alpha$  performance can also be considered as a variant of prognostic horizon (PH) metrics (Saxena et al., 2008) and can be readily converted to it if necessary.

## 5. CONCLUSION

Prognostics problem of the PEMFC is studied in this paper. A prognostic-oriented aging model is presented to describe the slowly-varying dynamics in the fuel cell that characterize the degradation process of the catalyst of the fuel cell. An UKF-based framework is proposed for the health monitoring and prognostic scheme and applied to solve the problems. The outcome of the prognosis scheme provides information about the precision and accuracy of long-term prediction, RUL expectations and 95% confidence intervals. Simulation is carried out for the validation of the proposed scheme with relevant prognostics metrics. The results show that with measurement correction, the health monitoring system can successfully track the damage variable throughout the degradation process; while at any time during the aging process, the remaining useful life can be predicted with satisfactory accuracy given that future load input information is precisely known.

## REFERENCES

- Daigle, M., Saha, B., & Goebel, K. (2012). A comparison of filter-based approaches for model-based prognostics. *2012 IEEE Aerospace Conference* (pp. 1–10). Presented at the 2012 IEEE Aerospace Conference. doi:10.1109/AERO.2012.6187363
- Darling, R.M., & Meyers, J. P. (2003). Kinetic model of platinum dissolution in PEMFCs. *Journal of the Electrochemical Society*, *J. Electrochem. Soc. (USA)*, *150*(11), 1523–7. doi:10.1149/1.1613669
- Darling, Robert M., & Meyers, J. P. (2005). Mathematical model of platinum movement in PEM fuel cells. *Journal of the Electrochemical Society*, *152*(1), A242–A247. doi:10.1149/1.1836156
- Franco, A. A., Schott, P., Jallut, C., & Maschke, B. (2007). A Multi-Scale Dynamic Mechanistic Model for the Transient Analysis of PEFCs. *Fuel Cells*, *7*(2), 99–117. doi:10.1002/fuce.200500204
- Franco, A.A., & Tembely, M. (2007). Transient multiscale modeling of aging mechanisms in a PEFC cathode. *Journal of the Electrochemical Society*, *154*, B712.
- Franco, Alejandro A., Coulon, R., Ferreira de Morais, R., Cheah, S. K., Kachmar, A., & Gabriel, M. A. (2009). Multi-scale Modeling-based Prediction of PEM Fuel Cells MEA Durability under Automotive Operating Conditions (pp. 65–79). ECS. doi:10.1149/1.3210560
- Franco, Alejandro A., & Gerard, M. (2008). Multiscale Model of Carbon Corrosion in a PEFC: Coupling with Electrocatalysis and Impact on Performance Degradation. *Journal of The Electrochemical Society*, *155*(4), B367. doi:10.1149/1.2838165
- Franco, Alejandro A., Gerard, M., Guinard, M., Barthe, B., & Lemaire, O. (2008). Carbon Catalyst-Support Corrosion in Polymer Electrolyte Fuel Cells: Mechanistic Insights (pp. 35–55). ECS. doi:10.1149/1.3002807
- Okada, T. (2003). Effect of Ionic contaminants. *Handbook of Fuel Cells – Fundamentals, Technology and Applications* (pp. 627–646). Wiley & Sons.
- Orchard, M. E., & Vachtsevanos, G. J. (2009). A particle-filtering approach for on-line fault diagnosis and failure prognosis. *Transactions of the Institute of Measurement and Control*, *31*(3-4), 221–246.
- Orchard, M., Wu, B., & Vachtsevanos, G. (2005). A particle filtering framework for failure prognosis. *2005 World Tribology Congress III, September 12, 2005 - September 16, 2005, Proceedings of the World Tribology Congress III - 2005* (pp. 883–884). Washington, D.C., United states: American Society of Mechanical Engineers.
- Orchard, Marcos E., & Vachtsevanos, G. J. (2007). A particle filtering-based framework for real-time fault diagnosis and failure prognosis in a turbine engine. *2007 Mediterranean Conference on Control and Automation, MED, July 27, 2007 - July 29, 2007, 2007 Mediterranean Conference on Control and Automation, MED*. Athens, Greece: Inst. of Elec. and Elec. Eng. Computer Society. doi:10.1109/MED.2007.4433871
- Saha, B., & Goebel, K. (2008). Uncertainty management for diagnostics and prognostics of batteries using Bayesian techniques. *Aerospace Conference, 2008 IEEE* (pp. 1–8).
- Saha, B., Goebel, K., Poll, S., & Christophersen, J. (2007). An integrated approach to battery health monitoring using bayesian regression and state estimation. *Autotestcon, 2007 IEEE* (pp. 646–653).
- Saha, B., Goebel, K., Poll, S., & Christophersen, J. (2009). Prognostics Methods for Battery Health Monitoring Using a Bayesian Framework. *IEEE Transactions on Instrumentation and Measurement*, *58*(2), 291–296. doi:10.1109/TIM.2008.2005965
- Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., & Schwabacher, M. (2008). Metrics for evaluating performance of prognostic techniques. *Prognostics and Health Management, 2008. PHM 2008. International Conference on* (pp. 1–17).
- Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2009). On applying the prognostic performance metrics. *Proceedings of the annual conference of the prognostics and health management society*. Retrieved

from

[http://72.27.231.73/sites/phmsociety.org/files/phm\\_submission/2009/phmc\\_09\\_39.pdf](http://72.27.231.73/sites/phmsociety.org/files/phm_submission/2009/phmc_09_39.pdf)

- Schmittinger, W., & Vahidi, A. (2008). A review of the main parameters influencing long-term performance and durability of PEM fuel cells. *Journal of Power Sources*, J. Power Sources (Switzerland), 180(1), 1–14. doi:10.1016/j.jpowsour.2008.01.070
- Shimoi, R., Aoyama, T., & Iiyama, A. (2009). *Development of Fuel Cell Stack Durability based on Actual Vehicle Test Data: Current Status and Future Work* (No. 2009-01-1014). Warrendale, PA: SAE International. Retrieved from <http://www.sae.org/technical/papers/2009-01-1014>
- Wan, E. A., & van der Merwe, R. (2002). The Unscented Kalman Filter. In S. Haykin (Ed.), *Kalman Filtering and Neural Networks* (pp. 221–280). John Wiley & Sons, Inc. Retrieved from <http://onlinelibrary.wiley.com/doi/10.1002/0471221546.ch7/summary>
- Xian, Z. (2012). *Prognostic and Health-Management Oriented Fuel Cell Modeling and On-line Supervisory System Development*. Clemson University.

Automotive Control, IFAC Technical Committee in Automotive Control, Associate Editor of the Conference Editorial Board of the IEEE Control Systems Society, and Associate Editor of the SAE Journal on Alternative Powertrains.

## BIOGRAPHIES

**Xian Zhang** received his B.S. and M.S. degrees in Automotive Engineering from Tsinghua University, Beijing, China, in 2005 and 2007, respectively, and just received his Ph.D. degree in Automotive Engineering from Clemson University, Clemson, SC in August 2012. His Ph.D. research area was in the control, diagnostics and prognostics of the PEMFC system. From September 2007 to May 2012, he was a Graduate Research Assistant with the International Center for Automotive Research and Department of Automotive Engineering, Clemson University. From June to November 2010, he was an intern with National Transportation Research Center (NTRC), at Oak Ridge National Lab. Xian Zhang is a student member of ASME and SAE.



**Pierluigi Pisu** is an Assistant Professor at the Department of Automotive Engineering and holds a joint appointment with the Department of Electrical and Computer Engineering at Clemson University. He received his Ph.D. in Electrical Engineering from the Ohio State University in 2002. In 2004, Dr. Pisu was granted two US patents in area of model-based fault detection and isolation. Dr. Pisu's research interests are in the area of fault diagnosis and prognosis with application to vehicle systems, energy management control of hybrid electric vehicles, vehicle to grid interaction, and systems aging. He also worked in the area of sliding mode and robust control. Dr. Pisu is member of IEEE, ASME and SAE. He is member of the IEEE Technical Committee in

# Transducer Development for Nonintrusive Load Monitoring of Rotating Machinery

Mateja Putic, Nathan K. Brown, and Paul Muskopf

*Luna Innovations Incorporated, Charlottesville, VA, 22903, USA*

*{puticm, nbrown, muskopfp}@lunainnovations.com*

## ABSTRACT

Monitoring machine runtime health parameters through nonintrusive means can greatly reduce the up-front time and resource barriers to entry of adding instrumentation to existing plant infrastructure. This work presents the design and evaluation of three transducers as part of a nonintrusive load monitoring system for rotating machinery. Data collected using a custom designed, small-scale induction motor test stand shows the dependence of a large air core RF coil, small RF coil array, and Hall effect sensor outputs on applied motor speeds and mechanical loads (estimated based on generator power). Analysis indicates that the large air core RF coil transducer and the presented method for using nonintrusive collection of induction motor speed and stray flux can statistically measure the difference between any two load points with 95% confidence, if their values differ by 6.6% full scale or greater ( $\pm 2\sigma$ ). Additionally, areas of further research toward generalization of the approach are identified.

## 1. INTRODUCTION

Knowing the load profile of rotating machinery is an integral part of determining the useful life of motors and for predicting when motor parts will break down, and for developing maintenance schedules. While many models exist for PHM modeling of motors, the instrumentation that is necessary to collect data to feed into those models can be cumbersome to install and may require machine downtime and skilled technicians to install. Fully capable PHM system installation is especially expensive if it requires motor downtime, but in cases when the application is mission critical, downtime is not feasible. Consequently, facilities may forego the use of PHM systems completely due to the complexity of the install, leaving the benefits of these approaches as yet unrealized.

While hour meters or odometers are widely used, they do not provide indications of when the instrumented machine ran, for how long, or at what load level. For example, a machine may have a low hour count but be highly stressed if much of that time was accumulated over short run cycles. Direct methods of observing motor load include measuring output power by shaft displacement or input power by voltage, current, and power factor. Such approaches require some form of invasive instrumentation, however, such as proximity probes, torque-sensing couplings, and voltage/current interconnections, all of which are non-trivial to install, and usually constitute permanently installed instrumentation. Low-cost, non-invasive methods were desired for the present effort to ease integration effort and permit deployment on a wide range of assets.

Nonintrusive load monitoring of rotating machinery can offer a middle ground between traditional schedule-based maintenance, and fully modern PHM systems. This approach makes available an effective albeit simple indicator as a proxy for the mechanical output power of the motor. Ambient energy emitted by machinery can be tracked, analyzed, and classified to develop a motor load profile that illustrates how long a machine has run and at which load level as a percent of full scale. This type of profile can also be used to determine whether machines are operated within specifications. Finally, load profiling across a fleet of machines can be used to determine whether machine capacity is effectively distributed among plant loads, and to assist in scheduling plant capacity among parallel applications.

In related efforts on stray flux sensing, Adam, Gulez, and Koroglu (2001) provide a strong background on sensing stray flux distributions around electric motors with the objective of "provid[ing] safety regions for humans in the vicinity of these motors". This study documents methods of determining stray flux using x-axis, y-axis, and z-axis air coils mounted to the exterior of the test motor. Results from this paper provide general application rules for stray flux measurement, and this finding directed the placement of our prototype search coil on the test stand motor. An application

---

Mateja Putic et. al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

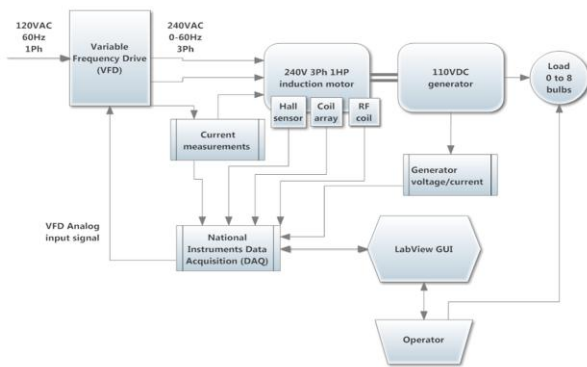


Figure 1. Motor test stand overview, instrumented for measurement of input power and output stray flux

note from the DOE Motor Challenge program (DOE, n.d.) documents traditional methods of determining motor loading and efficiency, input power or line current measurement, and a method based on slip. Additionally, Tumanski (2007) is an excellent reference for air core RF coil transducer and analog front end electronics design. Bin Lu, Habetler, and Harley (2006) provide descriptions of more than 20 efficiency estimation methods for in-service induction motors, and document the required tests and measurements, intrusion level, and average accuracy.

Our research sought to create a solution that is substantially easier to install than alternative approaches, to lower the up-front time and resource barriers to entry toward understanding plant capacity distribution. The broad vision of this system is to collect start/stop times and load levels on a widely distributed set of plant machinery, so that plant maintainers can effectively schedule maintenance resources and determine where machine capacity is underused or overused. The system presented is also designed to be compatible with energy harvesting from a variety of available sources which are abundant in the plant environment, identifying low power requirements as a factor of interest. The scope of this study details the research and development of a transducer that can reliably be used to estimate load nonintrusively through the monitoring of ambient emitted machine energy. Section 2 describes the motor test platform that was built for this study, and the design and preliminary evaluation of stray flux transducers. Section 3 details the characterization of transducer outputs over versus motor operating parameters, and describes the development of stray flux characterization models. Section 4 provides an analysis and statistical evaluation of experimental results. Section 5 contains conclusions and opportunities for further research.

## 2. MOTOR PLATFORM AND TRANSDUCERS

The central parameter of interest to be measured nonintrusively within this study was real-time mechanical

load on an induction motor. As part of system development, several different transducers for detecting stray flux were selected for evaluation to determine the most reliable method. Initially, a motor test stand was built with controllable shaft speed and mechanical load, to facilitate evaluation of experimental transducers.

### 2.1. Motor Test Stand

A motor test stand diagrammed in Figure 1 was built and was used as a basis for transducer evaluations. This stand consists of a variable frequency drive (VFD), a 1 HP 240 VAC three phase AC induction motor, a 110 VDC generator mechanically coupled to the drive motor shaft, and a switchable resistive load bank of eight light bulbs. The VFD accepts an input of single phase 120 VAC and converts it to a 240 VAC three phase output. By using the VFD, speed of the motor was controlled by varying the three phase output frequency.

To correlate stray flux transducer output to motor load, a motor power estimate was obtained by measuring the output voltage and current from the DC generator. Although this approach does not take into account motor losses, generator efficiency, and frictional losses, it achieved sufficient accuracy of relative motor load for transducer characterization.

The test stand was instrumented to control motor drive speed and electrical load, measure input power, and to measure outputs from three different sensors, all simultaneously. Two Tamura L10Z050S05 Hall effect sensors that output a voltage proportional to the current flowing through the center of the sensor enabled acquisition of the current from one phase of the VFD output, as well as the current produced by the generator. The actual output frequency from the VFD was obtained through tonal analysis of the signal from the Tamura Hall effect sensor, implemented using National Instruments LabVIEW. Furthermore, the frequency of the voltage output from any installed stray flux transducers was also analyzed to verify if those measurements agree with the tone analysis of the motor current sensor voltage. These signals, as well as the voltage output from the generator, were connected to a National Instruments Multifunction Data Acquisition (DAQ) platform, which was used to directly sample these signals and import them into a LabVIEW development environment.

To provide a visual display of data and a control interface to the VFD and the DAQ, a GUI was also developed using LabVIEW. This interface allows automatic control of the VFD output frequency, load level prompting, signal acquisition and processing, and data logging, for semi-automated stimulus and test data collection.



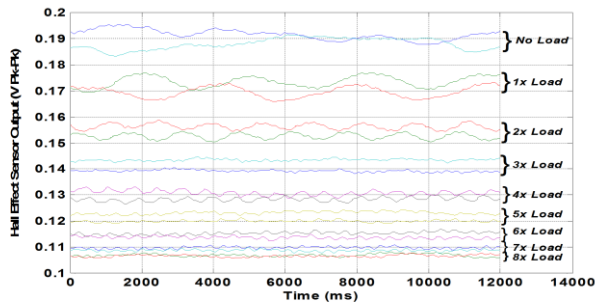


Figure 2. Hall Effect sensor amplitude for varying motor loads

## 2.2. Transducer Selection

The technique employed for nonintrusive load monitoring is to utilize the stray magnetic flux emitted from the motor to infer its output power. This approach relies on the fact that the finite permeability of a motor’s magnetic core cannot fully contain the generated flux from the windings, and a small portion of the core flux will ultimately stray to the surrounding environment. Although significantly reduced in scale, the stray flux density correlates with the magnitude of the motor core flux density, which is in turn a function of motor output power. Therefore, measuring stray flux density can provide an indirect, but reliable means of estimating of mechanical motor load. The following section details evaluation of a COTS Hall effect sensor, a small air core RF coil array, and large air core RF coil transducers for this purpose.

### 2.2.1. COTS Hall Effect Sensor

Hall effect sensors are predominantly employed to measure magnetic flux density for a variety of applications. For the present effort, the Allegro A1395 low-power linear Hall effect sensor IC was selected due to its small form factor, high sensitivity (10 mV/Gauss), and low power requirements. An initial survey of the test motor’s external magnetic field distribution was performed to determine a suitable location to mount the Hall effect sensor for load monitoring. The highest response was achieved at the edge of the stator core stack, above the endturn windings.

With the Hall effect sensor in place, the stray flux density was characterized over a range of applied motor load at a constant operating speed. For each load level, recorded flux density readings display a typical AC waveform at operating frequency. The peak-to-peak amplitude of this stray flux signal is plotted for each motor load level in Figure 2 while stepping through loading and unloading cycles. For each motor load, the signal falls within a discernible voltage band, the amplitude of which decreases steadily as motor power increases. Given the rated 10 mV/Gauss sensitivity

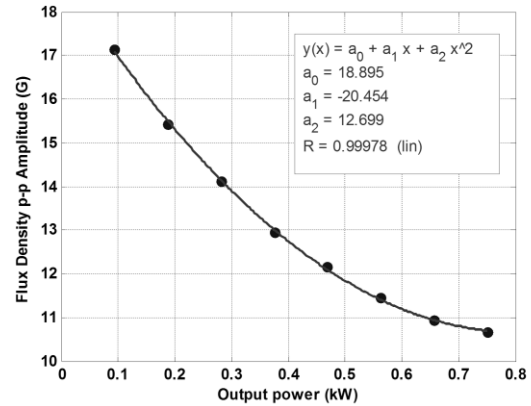


Figure 3. Relationship between RMS Hall effect sensor output and motor load

level of this transducer, the stray flux amplitudes range between 10 and 20 Gauss, or around 0.1% of the internal 10 kG flux density level within the motor.

Interestingly, the low frequency carrier wave of these peak-to-peak ripples is related to the motor power as well, likely due to the increasing rotor slip speed with increasing torque, as the induction motor rotor spins slightly slower than the alternating stator field with increasing mechanical load. The difference in rotational speed inducing current in the rotor conductors generate the corresponding torque. The low frequency ripple in these signals is established as a beat frequency between the non-identical induction rotor and stator field rotation speeds. However, this beat frequency was detectable only at low load levels, making it difficult to use this parameter as an additional indicator of machine load.

Figure 3 shows the average field amplitudes plotted against the power load level, which reveals a clear trend in both loading and unloading directions. For an induction motor, increased torque is produced by increasing rotor conductor current, resulting in a lower net current in the ‘transformer’ loop and consequently lower core flux. The unloaded condition is analogous to a transformer with an open secondary winding (i.e. high core flux), whereas the maximum power case represents the impedance matched condition for the transformer with good coupling between the windings, such that the core flux is considerably lower. The reduction of stray flux density with increasing motor load observed here is the expected result.

The observed repeatable trend of the Hall effect sensor output correlated with mechanical motor load demonstrates the ability of this sensor to nonintrusively monitor motor load level. Calibration in a field application would be required, following a similar surveying procedure as in this study. A calibration curve could be generated by exercising the motor of the operating range of interest while monitoring the output of the sensor. An important caveat here is that this calibration between monitored flux and

motor power will provide a reliable indication of motor load only if the sensor remains in the same mounting location.

**2.2.2. Small Air Core RF Coil Array**

The second transducer chosen for this effort was made up of an array of antennas salvaged from COTS RFID tags. The antenna on these tags is a small air core inductor made of 30 gauge wire with 25 mm OD and 15 mm ID and 1 mm thickness. As a preliminary survey step, a single inductive coil was adhered to the motor casing and its output RMS voltage was read with the DAQ. A clear step change was seen with each change in motor load. To increase the voltage response and measurement area, six of these coils were wired in series and bound together side by side. Though the absolute voltage output does differ with coil circumferential position, longitudinal position, distance from the coil to the motor case, and orientation, the relative voltage response to change in load did not change considerably.

**2.2.3. Large Air Core RF Coil**

Background research led to the choice of a single large air core RF coil as a potentially useful transducer for measuring stray flux around a motor. The RF coil was selected based on its high sensitivity, stability, and passive operation. Though an air core coil will have a lower voltage output compared to a magnetic core coil of the same dimensions, the air core was selected due to its linear response.

$$V = 0.5 \cdot 2\pi \cdot f \cdot n \cdot D^2 \cdot B \tag{1}$$

Equation 1 is the fundamental expression for induction coil output voltage in the presence of a changing magnetic field, according to Faraday’s Law (Tumanski, 2007). In this equation, *V* is the voltage output, *f* is the frequency, *n* is the number of turns, *D* is the mean diameter of the coil, and *B* is the flux density to be measured.

The design parameters of the coil were chosen based on a desired 1 V output at 60 Hz based at an anticipated maximum flux density output of 10 G measured during experimental testing conducted with the Hall effect sensor. The outer diameter of the coil was chosen to be less than 10 cm to accommodate the size of the motor. The final design is a coil of 540 turns wound from 30 gauge magnet wire, and the length of the coil is 18 wire diameters. Once the parameters for the coil design were chosen, a spool for the coil was machined from Type II PVC. The wire was manually wound onto the coil, and the outer edge of the coil was bound to the spool with Kapton tape. The shape of the spool was designed so that the wire coil was held tangent to the motor case.

As with the small coil array, the RMS voltage output from the coil was initially surveyed at various locations and orientations on the motor. The final location was chosen that such that the crosstalk between phases at the output of the

Parameter	Value
Motor Speed	1800, 2400, 2700, 3000, 3300, 3600 RPM
Generator Power Range	0-430 Watts (0-8 bulbs)
Step dwell time	30 seconds
Replicates	4x (216 total steps)
Transducers	Single Induction Coil Induction Coil Array Hall Effect IC
Data sampling rate	10 kHz
Motor Load Metric	Generator Output Power

Table 1. Stray Flux Transducer Characterization Study Parameters

transducer was minimized. Although the output voltage from the coil was of a higher magnitude than the voltage from the small array, the relative change in voltage with respect to motor load was very similar for nearly all drive frequencies compared to the small coil array.

**3. TRANSDUCER CHARACTERIZATION**

Preliminary sensor characterization revealed that the amplitude of the stray flux of the test motor varied not only with load, but with motor speed as well. Since the ultimate nonintrusive load monitoring system must operate under uncertain speed conditions, testing was conducted to characterize transducers across a wide range of loads and speeds. Outputs of the three stray flux transducers were characterized over the operational range of the motor test stand described in previous sections.

Table 1 provides the general parameters used for this set of experimental procedures. The operational range covers six motor speeds and nine power levels, where speed is automatically controlled by the LabVIEW application according to a user-defined profile and power is controlled manually by actuating individual light switches at the designated time. The system dwells at each load/speed parameter pair for 30 seconds before proceeding to the next step.

To provide an improved basis for statistics calculations, the speed and load profiles were replicated four times, and then randomized to develop the final profile. Input randomization during characterization and calibration minimizes the impact of interference, breaks up hysteresis errors, and ensures that each application of input values is independent from the previous. Generally, such random variation will more closely simulate the actual operating conditions (“Some Notes on Device Calibration,” n.d.). Figure 4(a) illustrates the randomized motor speed parameters generated for the test, showing the first 1500 seconds for improved visualization.

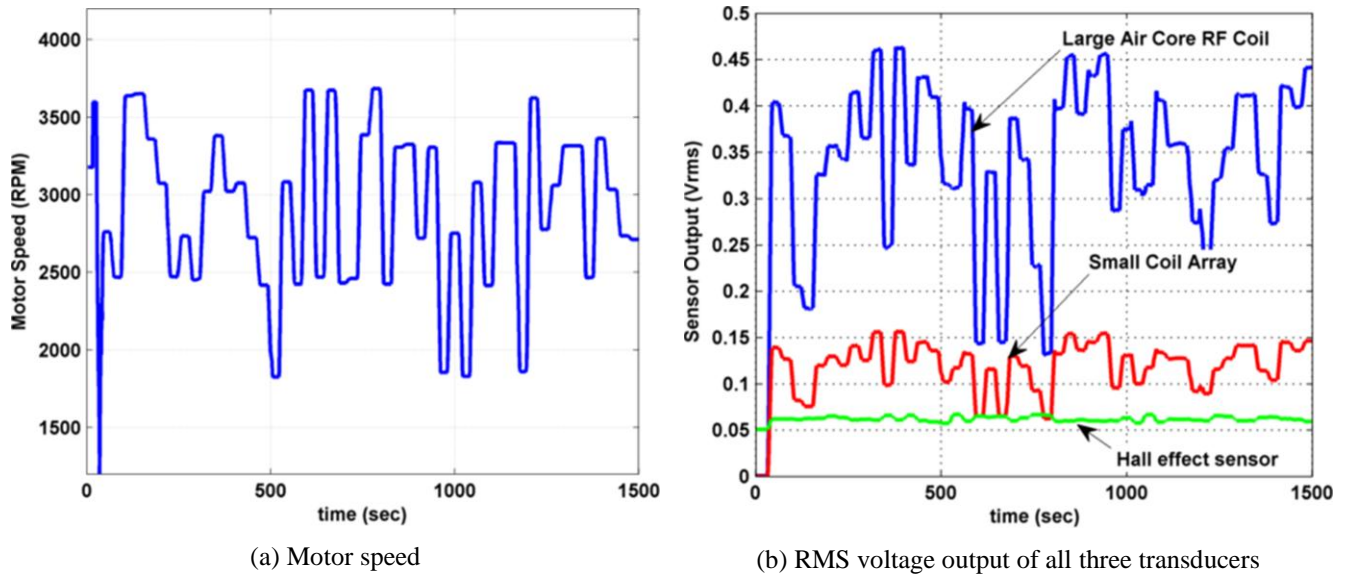


Figure 4. Representative time domain instrumentation data

As stated previously, generator output power was used as the means of monitoring motor load. It is notable that generator output voltage scales roughly with rotor speed, therefore the maximum output power is also speed dependent. The data presented in Figure 4(b) are the RMS values of the AC waveform for each sensor observed within one second windows of time, all showing correlation to the generator output power. The single and arrayed induction coil outputs trend inversely with power output, whereas the Hall effect sensor exhibits the opposite behavior. Relocating the Hall effect sensor around the same radius of the motor showed an inverse trend in the output, possibly due to the small size of the sensor compared to the size of the coil windings. The polarity of the outputs of the induction coils were not found to be sensitive to their location.

All of the signal streams were acquired at a rate of 10 kHz at one second intervals. The respective resulting 10,000 data points for each signal were then processed to provide a single RMS value for each one second time step, essentially reducing the data logging burden by several orders of magnitude. All subsequent analysis is performed using these RMS values. After the initial survey to validate that all transducer outputs were within expected intervals, a more rigorous analysis was performed to develop the sensor output trends over the speed and load operating envelope, discussed in the following section.

#### 4. DISCUSSION

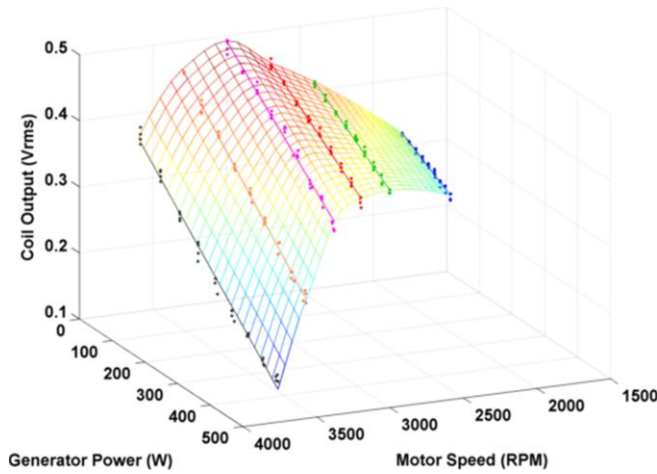
Figure 5 illustrates the dependence of sensor output on the applied motor speed and load (based on generator power) for the large coil (a), coil array (b), and Hall effect sensor (c). The colored dot markers indicate raw sensor data collected for each load and speed test point, with the marker colors indicating the different motor speeds. Figure 5(d)

shows the relative response of the three sensors plotted on the same scale, highlighting the very high sensitivity of the large coil relative to the other two transducers. The raw data collected during the experiment described above was post-processed to provide straightforward visualization of the sensor response over the operating envelope, and to develop some statistical indicators of each sensor's ability to accurately monitor the motor load.

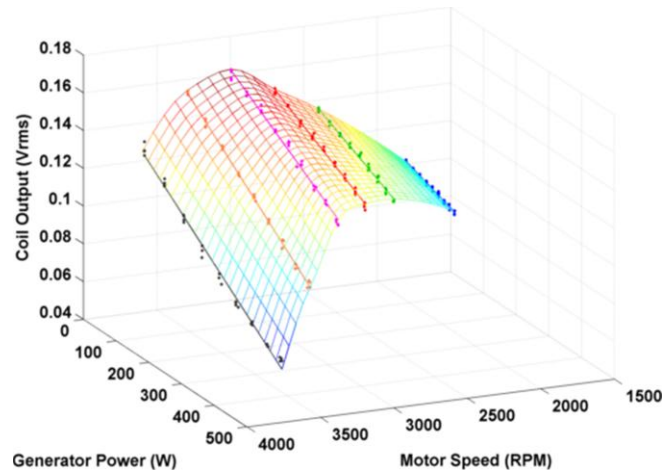
In all cases, the trend between the transducer output and output power is very linear and repeatable across motor speeds, as shown by the linear best-fit lines through the data. It is immediately evident, however, that transducer outputs vary with motor speed non-linearly. This fact warrants the measurement of speed by the nonintrusive sensor, in addition to stray flux. In the interest of maintaining the nonintrusive nature of sensor, the motor speed can be estimated by monitoring the fundamental frequency of the AC stray flux, instead of requiring a dedicated speed sensor. However, slip in induction motors under load may present an issue to the feasibility of this method, warranting further study.

As shown in Figure 5(d), collected data was interpolated to form a surface map that fully covers the motor's operating envelope. In practice, however, it is desirable to perform the load calibration with a minimum number of load and speed inputs given the likely limited ability to effect these changes in a plant environment. By inverting the axes so that motor power is on the z-axis, this data can be used to assemble a transfer function with two inputs (speed, RMS sensor output) that can be evaluated using inputs measured from a single coil mounted to the exterior of the motor.

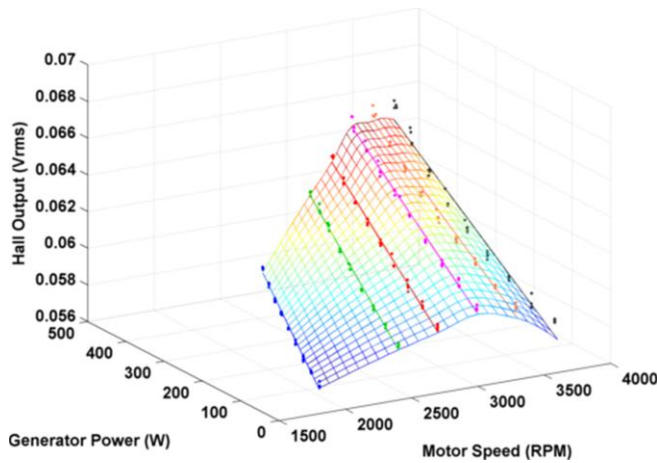
Using the calibration maps established above, it is straightforward to determine how well the measured data fit



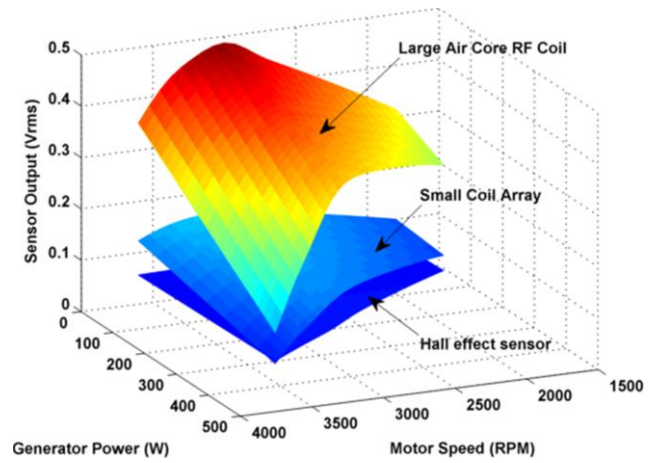
(a) Large coil output vs. motor speed, load



(b) Coil array output vs. motor speed, load



(c) Hall effect output vs. motor speed, load; axes reversed



(d) Relative response of the three sensors

Figure 5. Dependence of sensor output on the applied motor speed and load

to this surface and thereby produce an indication of the accuracy of the load estimation given the spread in the sensor data. The difference between actual measured load and the model estimate was calculated, forming a residual error for each data point. Figure 6 presents the residual errors between the best-fit model and the measured data for each motor speed and load parameter pair, relative to the full scale load. The standard deviation of these residuals provides an indication of how well the data fit the calibration map. Assuming the standard deviation of these values as  $1\sigma$ , the 95% confidence interval around the mean is established at  $\pm 2\sigma$ . In the case of the large coil, where  $\sigma = 3.3\%$  full scale, this analysis indicates that the model can statistically determine the difference between any two load points with 95% confidence if their values differ by 6.6% full scale or greater ( $\pm 2\sigma$ ).

The Hall effect sensor performs rather well from the standpoint that it produces repeatable data, as indicated by reasonable standard deviations on the error, despite its very

low sensitivity relative to the induction coils (10 vs. 400 mV full scale output). The coil array was less sensitive than the large coil and also produces a greater relative spread in the output, indicating a lower relative signal-to-noise ratio. From a sensor robustness standpoint, the large coil displays higher output, greater noise immunity, and the greatest precision among the three transducers.

Finally, Figure 7 illustrates the large coil response surfaces for two independent tests, revealing excellent agreement between the two runs. Using the large coil sensor response to compare two independent data collection runs provides a qualitative evaluation of repeatability of the proposed method.

## 5. CONCLUSIONS AND FUTURE RESEARCH

This work detailed the development of three transducers for stray flux detection as part of a system for nonintrusive load monitoring of rotating machinery. The large RF coil

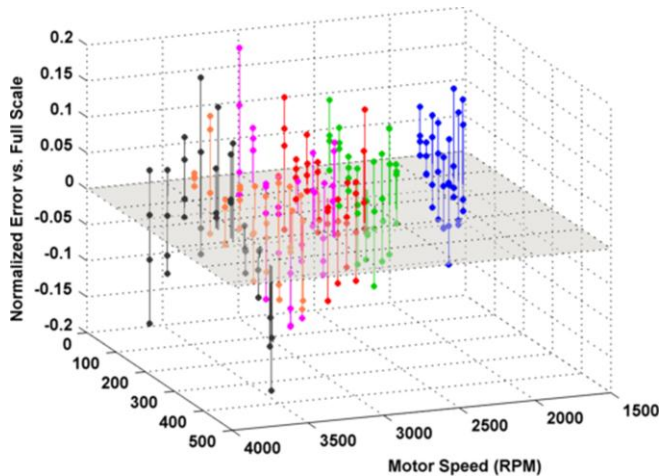


Figure 6. Normalized load estimation error for large air core RF coil, which was the best overall performer

designed in this study was found to be the most effective transducer for measuring stray flux of an induction motor for the purpose of nonintrusively determining its mechanical load. Additionally, the carrier frequency of the same signal was found to be an accurate representation of the drive frequency of the motor. This suggests that a single transducer could be used to generate inputs to a two input (speed, RMS sensor output) transfer function to motor load. A statistical analysis of the transfer function showed that the calibration model can be used to determine the difference between any two load points with 95% confidence if their values differ by 6.6% full scale or greater ( $\pm 2\sigma$ ). This demonstrates the validity of the approach to be used as a model that can predict mechanical motor load without any intrusive measurements.

Calibration of all studied transducers was determined to be necessary between placement and orientation cycles. However, in an active plant facility, placement and orientation of the nonintrusive sensor may be changed during maintenance or other activities. For practical purposes, it would be very convenient to develop a transducer or sensing method that would not require calibration between placement, allowing more flexible placement options. Further testing should be performed to measure the effects of changing position, orientation, and distance from the motor case on output voltage with respect to load.

Finally, further effort is warranted toward generalizing the method for application to a variety of machine types, including synchronous motor architectures. It remains to be determined whether the presented method would scale to larger motors, which may emit less stray flux due to higher efficiencies. Further experimental study will seek to identify common stray flux density signatures from different sizes of motors and motors based on various operating principles, such that these surfaces can be parameterized according to

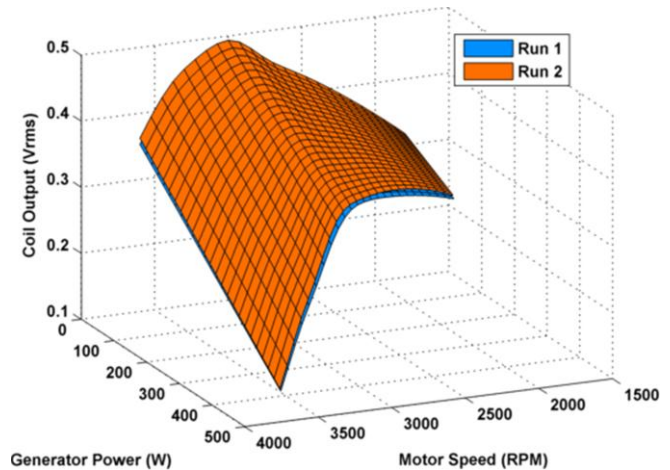


Figure 7. Comparing large coil response between two test runs as a preliminary assessment of repeatability

its expected characteristics, and require fewer calibration points to refine the calibration model once installed in the field. Determining similarities of the stray flux signatures emitted by various types of machines would facilitate the development of a parameterized model that could then be configured with motor specifications or a small number of calibration points, that would scale the model to fit the particular machine of interest. This and other development is needed to minimize the calibration effort needed to enable this approach to work in a practical setting.

#### ACKNOWLEDGEMENT

This material is based upon work supported by the United States Air Force under Contract No. FA9101-11-C-0032. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force.

#### REFERENCES

- Adam, A. A., Gulez, K., & Koroglu, S. (2001). Stray magnetic field distributed around a PMSM. *Turk J Elec Eng & Comp Sci*, 19(1).
- Bin Lu, Habetler, T. G., & Harley, R. G. (2006). A survey of efficiency-estimation methods for in-service induction motors. *IEEE Transactions on Industry Applications*, 42(4), 924–933. doi:10.1109/TIA.2006.876065
- Determining Electric Motor Load And Efficiency. (n.d.). U.S. Department of Energy.
- Some Notes on Device Calibration. (n.d.). Retrieved January 18, 2012, from <http://www.iceweb.com.au/Test&Calibration/NoteDeviceCalibration.pdf>
- Tumanski, S. (2007). Induction Coil Sensors - A Review.pdf. *Meas. Sci. Technol.*, (18), R31–R46. doi:10.1088/0957-0233/18/3/R01

# A Reasoning Architecture for Expert Troubleshooting of Complex Processes

Abdul Naveed<sup>1</sup>, Jiaming Li<sup>2</sup>, Bhaskar Saha<sup>3</sup>, Abhinav Saxena<sup>4</sup>, and George Vachtsevanos<sup>5</sup>

<sup>1,2,5</sup> *Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332, USA*

*naveed@gatech.edu*

*jli339@gatech.edu*

*gv@ece.gatech.edu*

<sup>3</sup> *Palo Alto Research Center, Palo Alto, CA, 94304, USA*

*bhaskar.saha@parc.com*

<sup>4</sup> *SGT Inc., NASA Ames Research Center, Moffett Field, CA, 94035, USA*

*abhinav.saxena@nasa.gov*

## ABSTRACT

This paper introduces a novel reasoning methodology, in combination with appropriate models and measurements (data) to perform accurately and expeditiously expert troubleshooting for complex military and industrial processes. This automated troubleshooting tool is designed to support the maintainer/ repairman by identifying and locating faulty system components. The enabling technologies build upon a Model Based Reasoning paradigm and a Dynamic Case Based Reasoning method acting as the intelligent database. A case study employs a helicopter Intermediate Gearbox as the application domain to illustrate the efficacy of the approach.

## 1. INTRODUCTION

Complex military and industrial systems (machines, aircraft, etc.) experience fault/failure modes that must be diagnosed accurately and rapidly in order to sustain the operational availability of these systems as high as possible. Considerable downtime translates into loss of productivity and increased maintenance costs. In a manufacturing or industrial setting, problems reported by the machine's internal checks (fault indicator) or an operator alert the maintainer of possible problem areas that must be addressed. Unfortunately, in these situations, the alert or advisory does not offer sufficient information to the

maintainer that may permit the accurate and rapid diagnosis of the problem.

An "expert" observes the faulty system, determines the root cause of the problem and composes a work order to schedule people, tools/equipment, or materials for repair and maintenance. If the problem is not diagnosed correctly and corrected, the troubleshooting and repair task is passed on to other technical personnel until a successful solution to the problem is reached. Early efforts to assist the maintainer with troubleshooting and early diagnosis have ranged from built-in test (BIT), and built-in test equipment (BITE) to interactive electronic technical manuals (IETM's). More recently, several organizations have employed rule-based expert systems to provide the maintainer with more systematic tools and methods for fault/failure diagnosis. Although significant improvements have been reported in the troubleshooting arena, the challenge of minimizing the diagnosis/repair cycle and returning critical equipment to service as soon as possible is still remaining and new methods/tools are sought to address it.

This paper addresses the development, testing and evaluation of an intelligent and systematic methodology to troubleshooting that aims to improve current practice and expedite the diagnosis/repair cycle. An expert decision support system is designed to assist the maintainer in navigating through complex system interconnections while reducing the variability of coupled components/subsystems into a well understood series of steps. The enabling technologies borrow from the domains of data mining, modeling and such reasoning paradigms as Model Based Reasoning and Case Based Reasoning. The objective is to

---

Abdul Naveed et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

integrate trouble-shooting practice with technologies from Condition Based Maintenance as many military and industrial installations are adopting these new and emerging technologies. The vision is the eventual development of a rigorous and automated diagnosis and troubleshooting architecture that will optimize maintenance, repair, and overhaul of complex assets.

## 2. THE TROUBLESHOOTING METHODOLOGY

Diagnostic reasoning algorithms use troubleshooting

information modeled in a specific way to select the appropriate troubleshooting procedure, recommend tests and corrective actions as new evidence is acquired, and determine the root cause of the system failure. As shown in Figure 1, we suggest a novel combination of Model-Based Reasoning (MBR), a reasoning tool about fault modes, and Dynamic Case Based Reasoning (DCBR), a ‘smart’ knowledge-base. We describe each of these two tools and their contribution to expert troubleshooting in the following sections.

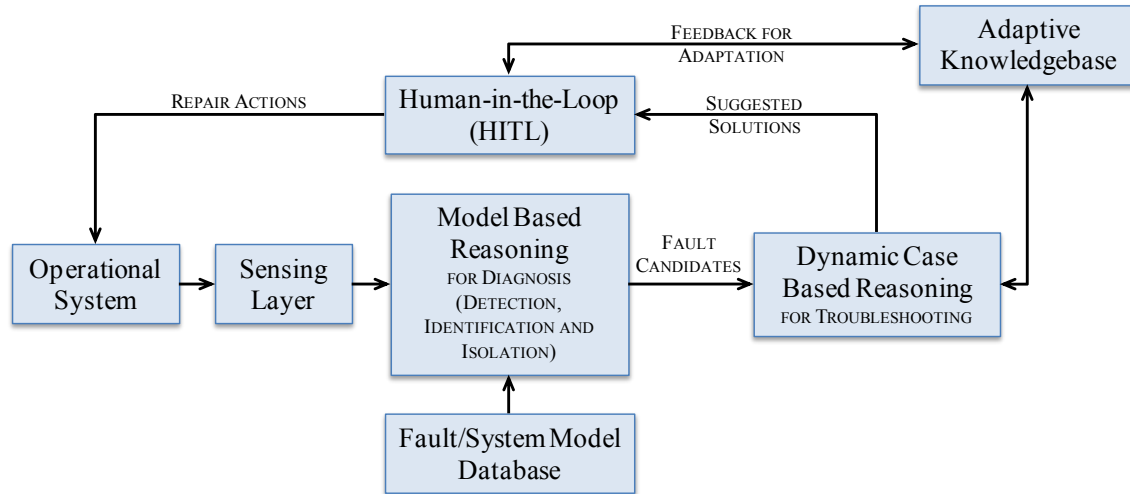


Figure 1. Schematic showing troubleshooting methodology based on reasoners.

## 3. DIAGNOSTIC REASONING

Critical components that are prone to failure are usually constituent elements of a larger subsystem or system. For example, a gearbox, found on most land, sea and aerial vehicles, is an assembly of various components like shafts, bearings, gears, etc. Such subsystems are instrumented with accelerometers mounted on their frame, which respond to vibration signals that might originate from anyone of the gearbox components. It is crucial, therefore, to devise methodologies that are capable of determining which component is defective and, additionally, how specific components, faults/failures may propagate from faulty to healthy components (the domino effect!) Model-based reasoning (MBR) belongs to this methodological category. It utilizes all the system sensors to enhance and confirm fault isolation. It is an intuitive multi-level modeling approach with inherent cross checking for false alarm mitigation and multi-level correlation for accurate failure isolation (Davis, 1984; De Kleer & Williams, 1987).

### 3.1 Model-based Reasoning-The Knowledge Database

The starting point of this diagnostic reasoning methodology is the creation of the *a priori* knowledge database. In the decomposition of a system into its components, often satisfactory results can be achieved by looking only as deep as the macro level where we have commercially available

replaceable components whose behaviors have been studied in detail. Once the component units have been identified, their behavior is studied and stored in the database in the form of functional descriptions aided by a FMECA study. The functional description of the system is described in terms of these behaviors. Associated with these functions the components also exhibit specific fault modes. These anomalies, represented in the form of features extracted from signals gathered from the system, are called *fault features* or *condition indicators* (CI's). The features are also stored in the database along with the functional description for each component. The basic unit of reasoning here is the fault mode  $j$  of component  $i$ , represented logically as:

$$F_{ij} = \bigwedge_k f_{ijk} \quad (1)$$

where,  $f$  denotes feature threshold being exceeded,  $k$  is the symptom index and  $\wedge$  denotes logical AND.

#### 3.1.1 System Model Abstraction

Given a system, we start out by analyzing its structural links. The different component parts are identified and their specific structural organization is stored in the form of a *structural model*. Information about the location of each component and its relative proximity to other neighboring components is crucial in predicting how a local failure in one component may propagate through the entire system. In

logical terms, this information is represented as:

$$\begin{aligned}
 & \text{Set of components, } V = \{v_1, v_2, \dots, v_n\} \\
 & \text{Interconnection, } E = \{<v_a, v_b>, \dots, <v_i, v_j>, \dots\} \\
 & \text{Structural Model, } M_S \equiv V \cup E. \quad (2)
 \end{aligned}$$

where,  $\cup$  denotes the union of two sets.

The *functional model* of the system is constructed by traversing the partially connected graph represented by the structural model and substituting the corresponding function for each component from the database. Any anomaly in the system response is then reasoned about and expressed in terms of faulty operational mode(s) of one or more components. This idea is formalized as:

$$\text{Functional Model, } M_F \equiv \bigwedge_i \bigwedge_j \neg F_{ij} \quad (3)$$

i.e., the nominal system behavior is described as the absence of all known fault modes.

### 3.1.2 The Fault-Symptom Matrix

The construction of the *fault-symptom matrix* is the main reasoning step regarding overall system behavior. Each unit in the functional model is associated with a number of fault modes, with each fault mode corresponding to one or more condition indicators. A system fault is defined as:

$$F \equiv \neg M_F = \neg \bigwedge_i \bigwedge_j \neg F_{ij} = \bigvee_i \bigvee_j F_{ij} \quad (4)$$

i.e., at least one fault mode has been excited. Here  $\neg$  denotes logical NOT and  $\vee$  denotes logical OR.

A matrix tabulating the various fault modes and their symptoms is generated by traversing all units of the functional model and extracting their features from the database. If sufficient data from seeded fault testing is available from a FMECA study, then the fault-symptom matrix can be enhanced with criticality metrics like severity and frequency of occurrence.

As an example, Table 1 gives the fault-symptom matrix of a generic case where we consider 3 symptoms spread over 3 faults. The fault modes can be written as:

$$\begin{aligned}
 F1 &= S1 \wedge S2 = (S1 \wedge S2 \wedge S3) \vee (S1 \wedge S2 \wedge \neg S3) \\
 F2 &= S1 = (S1 \wedge S2 \wedge S3) \vee (S1 \wedge S2 \wedge \neg S3) \vee \\
 & \quad (S1 \wedge \neg S2 \wedge \neg S3) \vee (S1 \wedge \neg S2 \wedge S3) \\
 F3 &= S3 = (S1 \wedge S2 \wedge S3) \vee (S1 \wedge \neg S2 \wedge S3) \vee \\
 & \quad (\neg S1 \wedge \neg S2 \wedge S3) \vee (\neg S1 \wedge S2 \wedge S3)
 \end{aligned} \quad (5)$$

	Symptom 1 (S1)	Symptom 2 (S2)	Symptom 3 (S3)
Fault Mode 1 (F1)	X	X	
Fault Mode 2 (F2)	X		
Fault Mode 3 (F3)			X

Table 1: A generic fault-symptom matrix (X denotes a valid fault-symptom relation).

Using baseline data for the system under study, we calibrate this matrix for acceptable levels of the fault mode symptoms. The choice of thresholds on these symptoms is arbitrary. In the real world, maintenance personnel pick these thresholds from operational experience. In the absence of such expert knowledge we assume that the data is *normally distributed* (equation 5), and we construct this distribution based on the mean  $\mu$  and standard deviation  $\sigma$  of the given baseline data. We use the upper 3-sigma limit as our threshold.

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/(2\sigma^2)} \quad (6)$$

A multi-branched *diagnostic tree* is then constructed from the measurements of the overall system behavior. The nodes of the tree represent monitored system variables, whereas the leaves denote the components that are fault candidates. For the example above, the diagnostic decision tree is shown in Figure 2, where HIGH branches correspond to the X's in Table 1.

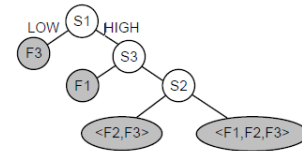


Figure 2. Diagnostic decision tree for the fault-symptom matrix in Table 1.

The leaf nodes below  $<F_i, \dots, F_n>$  in Figure 2 denote that any possible combination of the included faults may occur. Thus,  $<F1, F2, F3>$  includes the fault modes  $F1, F2, F3, F1 \wedge F2, F1 \wedge F3, F2 \wedge F3,$  and  $F1 \wedge F2 \wedge F3$ .

The overall MBR-derived diagnostic procedure can thus be represented by the schematic shown in Figure 3. More details of this architecture have been published by Saha & Vachtsevanos (2006).



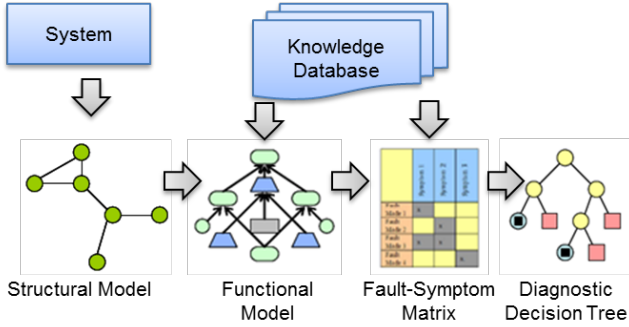


Figure 3. Schematic of MBR diagnostic reasoning architecture.

### 3.2 MBR Enhancements – Confidence Metrics

In this section we briefly outline an MBR enhancement that incorporates confidence metrics in the reasoning procedure. The architecture is elaborated by substituting the crisp Boolean logic implicitly inherent in the reasoning process, with *probabilistic* values that express the *confidence* of a given symptom. The confidence metric expresses the deviation of an observed symptom measurement from an expected baseline value.

The reasoning method is suggested above is a crisp Boolean logic. The occurrence of a behavioral anomaly appears as the deviation of the feature (or CI) value from a fixed threshold. When this deviation becomes distinct, the feature  $f$  is set from 0 to 1. Unfortunately, such a crisp distinction does not reveal the intermediate levels of confidence that are associated with the observation of feature values. In many cases the certainty of a behavioral anomaly based on the observation of a feature value is best described by a value within the  $(0,1)$  interval.

In order to insert the concept of confidence in the MBR architecture we propose that a Particle Filter (PF) module be used to generate the confidence that a feature exhibits anomalous behavior. Denote by  $\phi_i$  the feature measurement that corresponds to the symptom  $s_i$ . The output of the PF algorithm is a *sequence of pdf's*, denoted by  $P_{\phi_i}(k)$ . At each instant  $k$ , the pdf  $P_{\phi_i}(k)$  represents the probability distribution of the feature  $\phi_i$ . This is a function of the time instant  $k$ . Historical data are used to determine the baseline pdf denoted by  $P_B$ , which of course remains constant with time.

The confidence that the symptom  $s_i$  exhibits anomalous behavior based on the measurements of the feature  $\phi_i$  is defined by the overlap area of the two pdf's,  $P_{\phi_i}(k)$  and  $P_B$ . This overlap area  $P_{\phi_i}(k) \cap P_B$  of the two pdf's is the confidence that the symptom  $s_i$  indicates *healthy* behavior based on feature  $\phi_i$ , given by  $C_{healthy}(\phi_i; k) = P_{\phi_i}(k) \cap P_B$ .

Therefore, the confidence that the symptom exhibits *anomalous* behavior is  $C_{anomalous}(\phi_i, k) = 1 -$

$$C_{healthy}(\phi_i, k) = 1 - P_{\phi_i}(k) \cap P_B$$

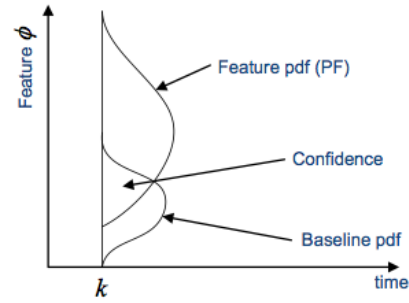


Figure 4. The confidence estimation using PF.

Figure 4 illustrates how the PF module is used to determine the confidence that a feature exhibits anomalous behavior.

For example, Figure 5 shows the time-series progression of Feature 1 denoting the energy in the gear natural frequency in dB, in a gearbox diagnostics scenario (explained in detail in section 4). The baseline pdf indicates the 3-sigma fault threshold at -30dB. From the graph it is clear from the PF tracking output (shown in red circles) that the anomalous behavior confidence  $C_{anomalous}(\phi_i; k)$  increases with  $k$  for this feature.

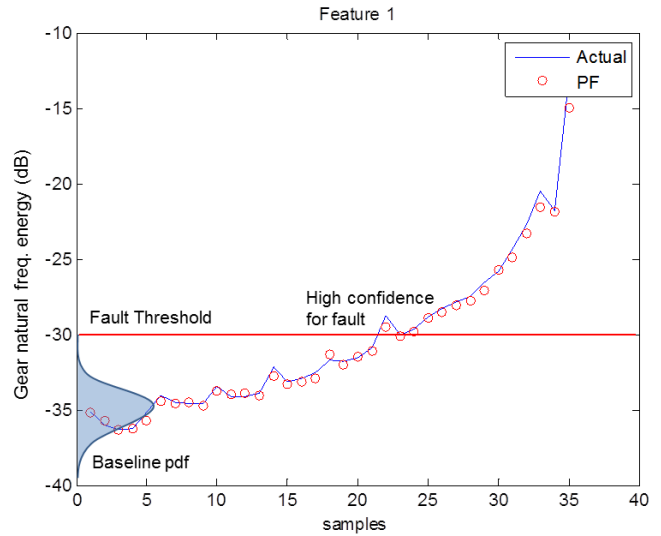


Figure 5: Feature 1 actual and estimated signal from the PF algorithm.

### 3.3 Dynamic Case-based Reasoning (DCBR)

Having determined the fault with sufficient confidence, DCBR constitutes the main system level reasoning for troubleshooting options and incorporates essential elements of a learning strategy. Case-Based Reasoning (CBR) was founded on the belief that human memory is episodic in nature, which comprises human knowledge accumulated from past experience. Faced with a new problem, a human

often relates the problem to one or more memory episodes and composes a solution from these episodes. CBR is a computer program to simulate this human recognition process and has been applied to a variety of process operation support systems. The CBR application domain usually involves problem solving, i.e. identify similar cases for better understanding, assessing and/or comparing with the current situation. The CBR architecture entails five basic steps:

- **indexing** – given a new situation, generate appropriate semantic indices that will allow its classification and categorization;
- **retrieval** – given a new, indexed problem, retrieve the best past cases from memory;
- **adaptation** – modify the old solution to conform to the new situation, resulting in a proposed solution;
- **testing** – determine whether the proposed solution is successful; and
- **learning** – if the solution fails, explain the failure and learn how to avoid repeating it; if the solution succeeds, incorporate it into the case memory as a successful solution.

The proposed software framework imposes requirements that classical (static) CBR methods are not capable of addressing due to the dynamic nature of the systems of interest and the temporal dependence of problem solutions typically found in troubleshooting. To circumvent such issues a new reasoning paradigm called Dynamic Case Based Reasoning (DCBR) was introduced in (Saxena, 2007; Saxena, Wu, & Vachtsevanos, 2005). DCBR enhances the advantages of conventional CBR systems by interpreting from sensor data not only static features but also dynamic and composite ones. Instead of one indexing path, the DCBR applies two—the abnormal symptom (AS) path, i.e. problem situation detected, and the problem description (PD) one. Furthermore, it entails functions to support case similarity evaluation and situational prediction through temporal reasoning and time tagged indexes. The remembrance calculation module updates the remembrance factors of existing cases. Figure 6 shows the major modules of the DCBR architecture.

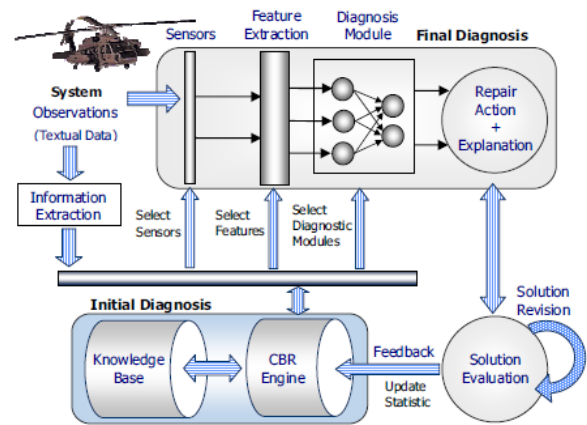


Figure 6. DCBR system for integrated diagnosis of industrial systems (Saxena et al., 2005).

Here, DCBR module depends on MBR, to provide probable fault candidates. Once a set of fault candidates are available, DCBR carries out a targeted search within the knowledgebase to identify if any additional measurements are required to confirm the and isolate the fault modes. The next step is selection of the cases from the case base on which reasoning will be based. In this step the notion of similarity between the cases in memory and the new episodic evidence is applied and the performance relies on the quality of similarity calculation algorithms. Depending on the information type contained in the case base the similarity function can take various forms. Several similarity scoring functions have been suggested in literature. As an example we present one such function here. Let  $Ent_e$  be a new case presented to the system. A “similarity by proximity” notion may be calculated by the following scoring function (Bichindaritz, 1995):

$$sim(Ent_e, Ent_j) = \frac{\sum_{k=1}^n \alpha \times sim(El_{i,k}, El_{j,k}) + \sum_{k=1}^n n_{k_i, pred} \times n_{i, pert} \times sim(El_{i,k}, El_{j,k})}{\alpha \times n + \sum_{k=1}^n n_{k_i, pred} \times n_{i, pert}} \quad (7)$$

Where  $Ent_j$  are cases previously presented,  $El_i$  is a feature or an (attribute, value) pair,  $n_{i, pert}$  is a pertinence weighted variable associated with the description element  $El_i$ ,  $n_{k_i, pred}$  is a prediction weighted variable associated with each case in memory. The similarity measure can be defined as:

$$sim(X, Y) = \sum_i \frac{1 - |x_i - y_i|}{1 + \alpha |x_i - y_i|} \quad (8)$$

### 3.3.1 Learning

Learning from the performance feedback is the key differentiator here with other knowledge based systems. Specific learning methods are applied according to the troubleshooting data environment and reasoner type. In our formulation there are two key learning mechanisms.

### 3.3.2 Human in the Loop Learning (HITLL)

As originally proposed in (Saxena, 2007) we adopt a Human-in-the-Loop Learning (HITLL) framework where not only the experts are kept in the loop for continuous performance monitoring, but also the system learning activity is accelerated. The human operator who actually performs the maintenance and repair operations (MRO) has the best knowledge about the effectiveness and the adjustments that were needed. He closes the loop between sensing and decision support and provides an assessment of results as well as a confirmation of the success or failure of control actions. The operator's expertise will assist to define and verify the completeness and correctness of the selected features for a particular problem set. It is then the task of system developers to devise appropriate mechanisms in the software to invite such feedback and incorporate the information by adjusting the weights in the knowledgebase.

### 3.3.3 Incremental Learning in DCBR

Incremental learning occurs whenever a new case is processed and its results are identified. Thus, the memory keeps track of each of its experiences, whether success or failure, in a declarative way; it is then ready to take advantage of future experiences. It must be pointed out that there is no guarantee that the matching process will lead to the most similar case (Kolodner, 1993). Incremental learning is pursued using Q-Learning, a popular reinforcement learning scheme for agents learning to behave in a game-like environment. Q-Learning is highly adaptive for on-line learning since it can easily incorporate new data as part of its stored database. An attractive feature in a game-like situation is that the player is learning to choose the best action for each particular game setting. In this framework, the expected reward or "cost-to-go" is stated as:

$$Q(s, a) \leftarrow Q(s, a) + \alpha [r + \gamma \max_{a'} Q(s', a') - Q(s, a)]$$

(9)

## 4. A CASE STUDY: THE IGB

We now discuss how we apply the proposed troubleshooting approach to a helicopter intermediate gearbox (IGB). A schematic view of the H-60 IGB is shown in Figure 7 followed by the functional diagram in Figure 8.

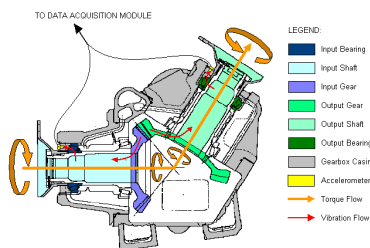


Figure 7. The IGB structure

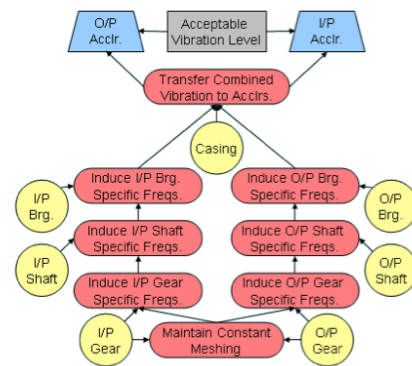


Figure 8. Functional diagram of the intermediate gearbox.

The three basic types of fundamental mechanical units of the IGB are gears, shafts, and bearings. A collection of features that are characteristic of anomalies in the IGBT functions provides a reliable platform for diagnostic reasoning.

The fault-symptom matrix (Table 2) shows that other than the gear natural frequency the remaining symptoms are different for the input and the output sides. To exploit this feature two diagnostic decision trees are constructed for each of the input and output accelerometer data. Taking advantage of the fact that most faults have only a single symptom we devise a simpler linear decision tree. This tree is traversed in a *depth-first* manner, visiting all the branches, starting from the root. This tree traversal is performed every iteration. Each iteration corresponds to a new dataset recorded from the sensors. When any one of these nodes returns a fault, then the same traversal process is applied to the right sub-tree of that node. It is to be noted that the search space is not narrowed down after initial diagnosis since a fault initiation in one component does not rule out the possibility of other components going bad.

Symptoms	Faults						
	Sub-synchronous	1 x shaft speed	2x shaft speed	(n/2)x shaft speed	Higher harmonics	Gear Natural freq.	g.n.f. sideband spacing
Input Gear Crack						820-825	65-70
Output Gear Crack						820-825	82-87
Input Bearing Race Defect				200-350			
Output Bearing Race Defect				250-435			
Excessive Clearance I/P Brg	20-50						
Excessive Clearance O/P Brg	25-60						
Input Shaft Imbalance		50-100					
Output Shaft Imbalance		60-125					
Input Shaft Misalignment			100-150				
Output Shaft Misalignment			125-185				
Input Shaft Looseness					350-450		
Output Shaft Looseness					435-560		

Table 2. Fault-Symptom Matrix for the IGB (the symptoms are represented by associated frequency bands in Hz).

The IGB has multiple fault modes depending upon which components have failed or are failing. For example, a bearing may develop a defect in the inner/outer race, a shaft or a gear pinion might develop a crack, or even a gear tooth may get worn out or chipped. All of these fault modes give rise to characteristic vibration signatures. By looking at the corresponding frequency bands in the accelerometer readings we can classify the fault modes present. Some common IGB fault classification heuristics are given below (Vance, 1988):

- Rotor Imbalance – 1 x shaft speed
- Shaft Misalignment – 2 x shaft speed, high axial vibration
- Mechanical Looseness – higher harmonics of shaft speed
- Excessive Bearing Clearance – sub-synchronous whirl instability
- Bearing Race Defect –  $(n/2)$  x shaft speed,  $n$  is the number of balls
- Gear Bevel Defect – gear natural freq., sidebands spaced at the running speed of the bad gear.

The fault-symptom matrix can be readily constructed and is shown in Table 2. Figure 9 depicts the computer flowchart.

The choice of thresholds for these symptoms is arbitrary. In this application, for all the symptoms, except for the side band spacing, the energy thresholds are set at the upper 3 sigma limit as determined from the baseline data. For the sideband spacing feature the symptom can take the value of either the input shaft or the output shaft speed. The system parameters required, as input to the MBR program for full IGB diagnostics, are given as 68.6 Hz input shaft speed, 25:31 gear ratio, and gear natural frequency around 822 Hz for both input and output gears.

Since the fault symptoms described are primarily energy metrics, the energy content of each data sample received is used to determine whether the system is in nominal mode or not. Once a fault is diagnosed at the system level we evaluate the diagnostic decision tree to isolate and identify the fault.

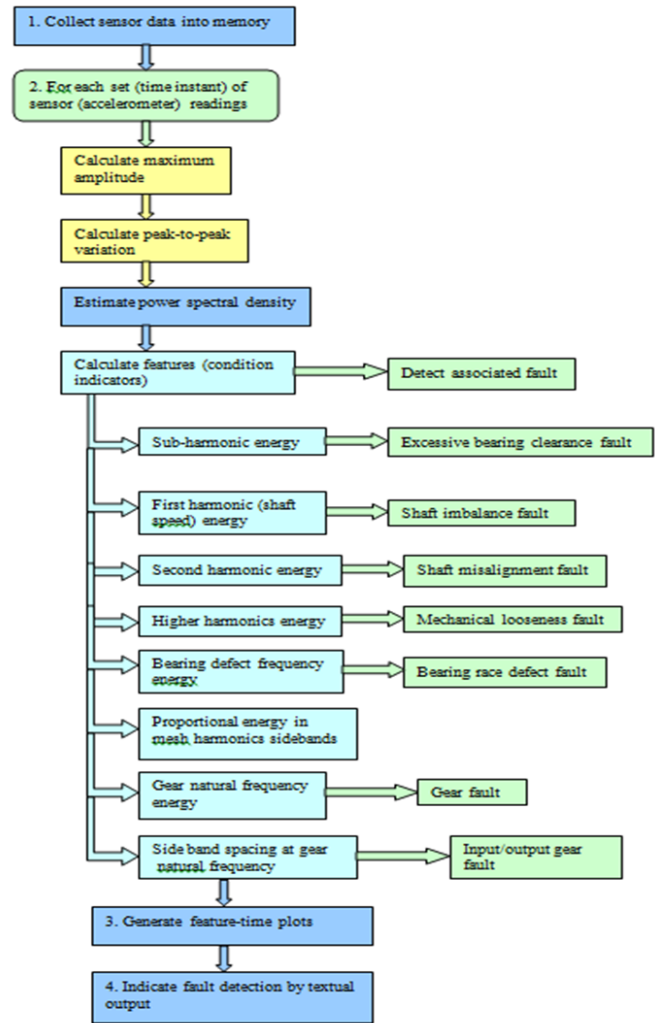


Figure 9. MBR Computer Flowchart for the Helicopter IGB Application.

### 5. RESULTS

For fatigue crack analysis an IGB pinion gear made of 9310 steel was used in a cyclic crack growth test. It was seeded with faults to initiate cracks. These faults consisted of notches made by an electric discharge machine (EDM), and were located at, and parallel to, the root of one of the gear teeth as shown in Figure 10. The crack growth test consisted of rotation in a spin pit, at a constant high speed with a varying load cycle, to simulate flight conditions. Data collection was done using 2 stud mounted accelerometers at the input and the output of the IGB.

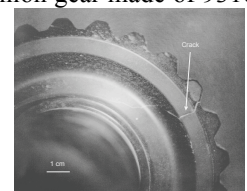


Figure 10. IGB crack. The crack growth test consisted of rotation in a spin pit, at a constant high speed with a varying load cycle, to simulate flight conditions. Data collection was done using 2 stud mounted accelerometers at the input and the output of the IGB.

The data consists of 36 sets of accelerometer readings from both the input and the output ends of the gearbox. A number of features were extracted by frequency domain

demodulation (the Fourier Transform of the vibration signals). The observed behavior from the data was compared with the predicted behavior from the functional model of the system. Any discrepancy in the two patterns was used to detect a fault condition, while the heuristics presented above were used to classify the fault.

A healthy gearbox exhibits certain characteristic frequencies. The primary component of the spectrum is the gear mesh frequency, which is 1714.5 Hz in our case. The harmonics of the mesh frequency are also present but their energy content varies with load conditions. When a crack initiates in the gear, the power spectrum starts to show a clear peak at the natural frequency of the gear. There are also characteristic sidebands spaced at the running speed of the bad gear. Thus, the major energy component of the signal shifts from the mesh frequency to the gear natural frequency and its sidebands.

For the frequency domain analysis of the given data we primarily look at the power spectral density. First, we eliminate the mesh frequency and its prominent harmonics. Then we measure the normalized energy at the gear natural frequency (822 Hz). This is a prime indicator of a growing fault condition. The spacing between the sidebands is then analyzed to see if it matches either the input or the output shaft speed. This provides a means of classifying which pinion gear is faulty. The photograph of the input pinion crack, shown in Figure 10, provides validation for our approach.

## 6. CONCLUSIONS

Current troubleshooting tools rely on fault tree analysis, extensive electronic manuals or expert system methods to assist the maintainer in identifying faulty system components and take corrective action. There is a need to expand the technology base with tools and methods that will assure the accurate and expedient identification of fault modes while providing means to learn from similar cases. This paper introduced a framework for an intelligent approach to the troubleshooting problem. More work is needed to improve the mathematical rigor of the algorithm and ascertain through case studies that appropriate performance and effectiveness metrics are met.

## REFERENCES

- Bichindaritz, I. (1995). Incremental concept learning and case-based reasoning: For a co-operative approach. Paper presented at the Progress in Case-Based Reasoning.
- Davis, R. (1984). "Diagnostic Reasoning Based on Structure and Behavior", *Artificial Intelligence*, Vol. 24, 1984, pp 347-410.
- De Kleer, J. & Williams, B.C. (1987). "Diagnosing Multiple Faults", *Artificial Intelligence*, Vol 32, 1987, pp 97-130.
- Kolodner, J. L. (1993). *Case-based Reasoning*. San Mateo, CA: Morgan Kaufmann Publishers.
- Saha, B. & Vachtsevanos, G. (2006). "A Model-Based Reasoning Approach to System Fault Diagnosis, WSEAS Transactions on Systems, Issue 8, Vol. 5, pp. 1997 – 2004, August 2006.
- Saxena, A. (2007). *Knowledge-Based Architecture for Integrated Condition Based Maintenance of Engineering System*. PhD Dissertation, Georgia Institute of Technology, Atlanta.
- Saxena, A., Wu, B., & Vachtsevanos, G. (2005). *Integrated Diagnosis and Prognosis Architecture for Fleet Vehicles Using Dynamic Case Based Reasoning*. Paper presented at the IEEE Autotestcon '05 Conference, Orlando, FL.
- Vance, J.M. (1988). *Rotordynamics of Turbomachinery*, John Wiley & Sons, Inc., 1988.

# Integrated fatigue damage diagnosis and prognosis under uncertainties

Tishun Peng<sup>1</sup>, Jingjing He<sup>1</sup>, Yongming Liu<sup>1</sup>, Abhinav Saxena<sup>2</sup>, Jose Celaya<sup>2</sup>, Kai Goebel<sup>3</sup>

<sup>1</sup>*Department of Civil & Environmental Engineering, Clarkson University, Potsdam, NY, 13699-5725, USA*

*pengt@clarkson.edu*

*jihe@clarkson.edu*

*yliu@clarkson.edu*

<sup>2</sup>*SGT, NASA Ames Research Center, Moffett Field, CA, 94035, USA*

*abhinav.saxena@nasa.gov*

*Jose.r.celaya@nasa.gov*

<sup>3</sup>*NASA Ames Research Center, Moffett Field, CA, 94035, USA*

*kai.goebel@nasa.gov*

## ABSTRACT

An integrated fatigue damage diagnosis and prognosis framework is proposed in this paper. The proposed methodology integrates a Lamb wave-based damage detection technique and a Bayesian updating method for remaining useful life (RUL) prediction. First, a piezoelectric sensor network is used to detect the fatigue crack size near the rivet holes in fuselage lap joints. Advanced signal processing and feature fusion is then used to quantitatively estimate the crack size. Following this, a small time scale model is introduced and used as the mechanism model to predict the crack propagation for a given future loading and an estimate of initial crack length. Next, a Bayesian updating algorithm is implemented incorporating the damage diagnostic result for the fatigue crack growth prediction. Probability distributions of model parameters and final RUL are updated considering various uncertainties in the damage prognosis process. Finally, the proposed methodology is demonstrated using data from fatigue testing of realistic fuselage lap joints and the model predictions are validated using prognostics metrics.

## 1. INTRODUCTION

For aerospace and civil engineering structures, various types of damage such as fatigue cracks, delamination, debonding, and corrosion have significant impact on service life. To reasonably assess their current health condition (Chong 1999; Masri, Sheng et al. 2004), a sensitive and precise

diagnostic method is needed. By analyzing continuous monitoring data, prognostics can provide valuable information for decision making within an effective structural health management (SHM) framework. Fatigue is

the root cause for a variety of failure modes in aircraft and rotorcraft. There are many non-destructive evaluation (NDE) methods available for fatigue damage diagnostics, such as, thermograph (Hung 1996; Koruk 2009), ultrasonics (Kazys and Svilainis 1997), X-ray CT (Nicoletto, Anzelotti et al.), etc. With the development of Lamb wave based damage detection methods, piezoelectric sensors have been widely used (Ward and Buttry 1990; Monkhouse, Wilcox et al. 1997; Lemistre and Balageas 2001; Giurgiutiu, Zagrai et al. 2002; Giurgiutiu 2003; Giurgiutiu 2005; Santoni, Yu et al. 2007) for structural health monitoring because of their low cost and high efficiency (Constantin, Sorohan et al. 2011). Lamb waves can propagate in thin plate without too much dispersion in certain modes (Scalea, Francesco et al. 2002). Using a proper mode selection, piezoelectric sensor networks have also been widely used for damage inspection of composite plate structures (Wang, Rose et al. 2004) and health monitoring of composite lap joints in aircraft wing. The built-in piezoelectric sensors are an ideal way to detect invisible fatigue crack growth in metal structures (Ihn and Chang 2004) and internal delamination in composite beam (JR and Chang 1993). However, most of the existing literatures focus on the plate-like components without more complex geometrical features. Damage is typically investigated away from the specimen boundaries to avoid complex wave reflections. Extending these detection methods to more complex structural components and prognostic enhancements requires further attention.

Tishun Peng et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

In order to predict fatigue crack propagation, various physics-based models have been developed. Paris' law (Paris and Erdogan 1963) is one of the most acknowledged fracture mechanics-based model for fatigue life prediction. However, Paris's law does not take stress ratio effects into consideration. Several modified versions of Paris' law have been reported in the literature, such as the modification for the near threshold crack growth (Forman 1967; Laird 1979), small/short cracks growth (Kitagawa and Takahashi 1979; Ritchie and Lank 1996), and crack closure (Elber 1970). However, most of these studies employ cycle-based methods and their application to randomly varying spectrums may be difficult. A time-based fatigue crack growth model (Lu and Liu 2010) was suggested that can overcome some of these inherent shortcomings in the classical cycle-based fatigue theories.

Furthermore, fatigue damage accumulation is a stochastic process and an appropriate inclusion and management of various uncertainties is critical for prognostics and decision-making. Bayesian updating algorithm has been shown to make use of condition monitoring data to improve model based prognosis and incorporate effects of different types of uncertainties (Guan, Jha et al. 2011). The uncertainty bounds for life prediction are contained (reduced) by updating the parameter distribution using the detected crack length through periodic measurements. Bayesian updating method has been used in this paper for integrated diagnostics and prognostics under uncertainties for a realistic lap joint geometry.

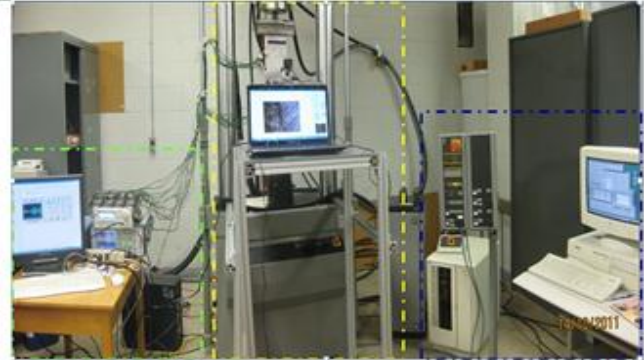
This paper is organized as follows. First, piezoelectric sensor based damage diagnostic method is proposed to detect the crack growth in a lap joint structure. Following this, A Bayes theorem based prognostic method is proposed that makes combined use of the damage detection method and crack propagation models. It is shown that uncertainties involved in the model parameters and detected measurement data are handled through Bayesian updating. Next, experimental data from lap joint coupons are used to demonstrate the applicability of this integrated method followed by model validation through prognostic metrics. The paper ends with conclusions and an outline of future work based on the current effort.

## 2. DAMAGE DIAGNOSIS USING A PIEZOELECTRIC SENSOR NETWORK

### 2.1. Experiment Setup

The experimental setup for damage prognosis of riveted lap joint coupons consists of three major modules: sensing and data acquisition system, fatigue crack ground truth measurement system, and fatigue cycling system (Fig. 1). Sensing and data acquisition system generates a 3.5 cycles tone burst lamb wave from Piezoelectric (PZT) actuators and records the corresponding signal received by PZT

sensors. Crack measurement system uses a traveling microscope to measure the crack length after a certain number of loading cycles at regular intervals. The specimen is subjected to tensile cyclic loading using the fatigue cycling system. The experiment setup is shown in Fig. 1. The coupons were subjected to two types of loading spectrums; constant block loading and variable block loading as shown in Fig. 2.



**Sensing and data acquisition system      Fatigue crack ground truth measurement system      Fatigue cycling system**

Figure 1. Systematic flow chart for the damage diagnosis system for lap joint

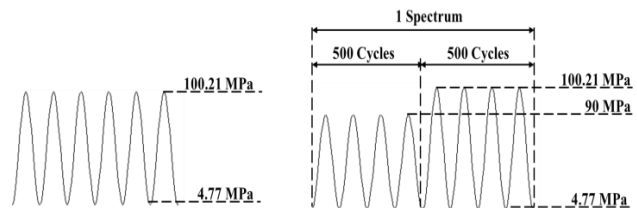


Figure 2. Loading spectrum. a), Constant loading. b), Variable loading

### 2.2. Specimen Geometry and Sensor Layout Design

The riveted panels are made of 0.063 inch Al 2024-T3 sheets that were originally provided by NRC, Canada. For repeatability, additional coupons were manufactured at NASA. Three rows of rivets are embedded in the panels. The detailed geometry is shown in Fig. 3 and corresponding mechanical properties of the material are shown in Table 1. To employ a pitch-catch method (Raghavan and Cesnik 2007), PZTs acting as actuators and sensors are glued on the two sides of the rivet holes.

The experimental results have shown that the major crack, which results in the final failure of the specimen, always appears at the countersunk hole in the first row. Therefore, the first row is considered as the target region for damage detection. In this study, our efforts are focused on detecting damage in the target region and accordingly, the actuators and the sensors are placed on opposite sides of the first row. This ensures the crack would be on the direct wave path of

the sensor-actuator pairs whenever it appears. The corresponding sensor network configuration is shown in Fig. 4. Red dots represent actuators away from the target region and the green dots represent sensors near the target region. Each pair of sensors can interrogate the damage information on their direct wave path.

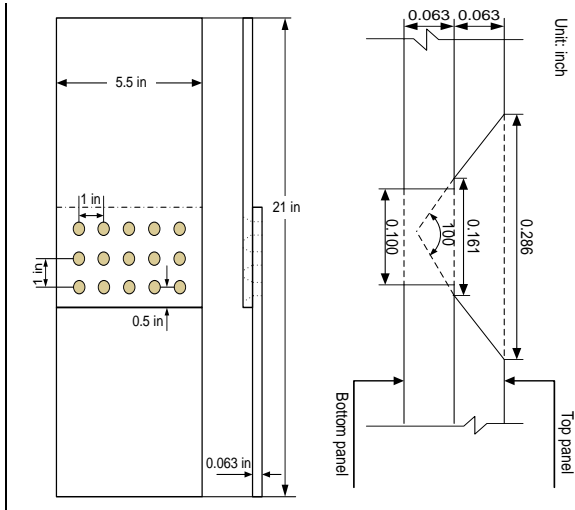


Figure 3. Geometry of the lap joint and detailed dimension of the connection part

Material	Yield strength (Mpa)	Modulus of elasticity (Mpa)	$\sigma_u$ (Mpa)	$\Delta K_{th}$ (Mpa $m^{0.5}$ )
Al2024-T3	360	72000	490	1.1164

Table 1. Mechanical properties of Al2024-T3

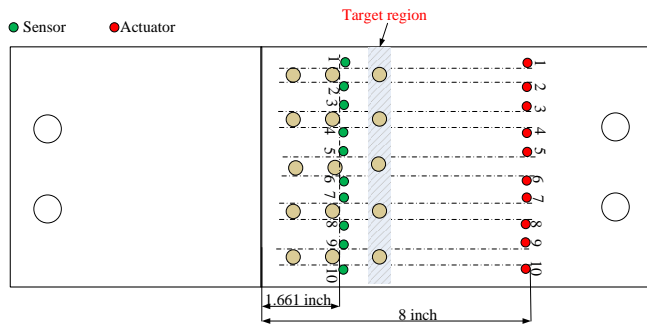


Figure 4. Sensor network layout for damage detection on riveted lap joint

### 2.3. Experimental Result and Data Processing

In this study, a Hamming-windowed sinusoidal tone burst with 3.5 cycles was used as the actuating signal. The central frequency of this signal was set at 200kHz, as shown in Fig. 5. After installing the specimen on the hydraulic machine, first a baseline signal for normal condition is collected.

Further data are collected periodically and the traveling microscope is used to measure the surface crack length at regular intervals. Fig. 6 illustrates a typical signal obtained for specimen T11 subjected to constant loading. With further signal processing, changes in selected features (phase change, correlation coefficient, and amplitude change) are calculated. The amplitude change reflects the energy dissipation due to the crack. The phase angle change is due to the traveling distance change due to the crack. The correlation coefficient change reflects the signal perturbation due to the new waves generated at the crack surfaces (Raghavan and Cesnik 2007). The detailed physical explanation of the chosen features is discussed in a companion papers and is under preparation. All of these feature changes can be obtained by comparing the received signals under normal and defect condition. Detailed description is given in literatures (Zhao, Gao et al. 2007; SU and Ye 2009; Wang and Yuan 2009). The relationship between trends in features and crack length is shown in Fig. 7.

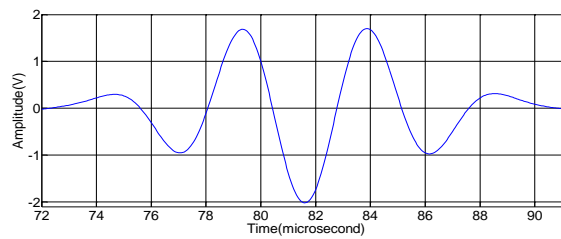


Figure 5. A tone burst signal of 3.5 cycles with 200kHz central frequency

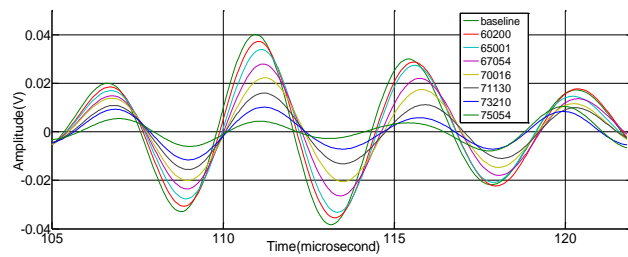


Figure 6. Received signal for specimen T11

Cycle#	Correlation coefficient	Phase change	Amplitude
55900	1	0	0.0201
60200	0.9895	0	0.0185
65001	0.9777	0.05	0.017
67054	0.9481	0.1	0.0149
70016	0.9233	0.1	0.0139
71130	0.9058	0.15	0.0109
73210	0.8717	0.15	0.0093
75045	0.7762	0.3	0.0055

Table 2. The corresponding signal interpreting result for T11 under constant loading



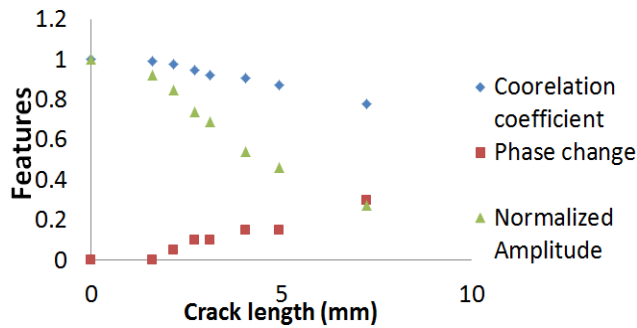


Figure 7. The relationship between signal features and crack length

As shown in Table 2. and Fig. 7, all features exhibit monotonic relationships with increasing crack length, but it is not trivial to predict the crack length using individual feature. Therefore, using data from several experiments, a multivariable regression is carried out. The corresponding formulation is shown in Eq. (1), which is trained using data from five different specimens (T1, T2, T3, T5 and T6). Table 2 lists the tuned regression coefficients for this formulation.

$$a = A + \alpha_1x + \alpha_2y + \alpha_3z + \alpha_4xy + \alpha_5xz + \alpha_6yz + \alpha_7x^2 + \alpha_8y^2 + \alpha_9z^2 \quad (1)$$

where  $A$  is the detected crack length,  $x$  is the correlation coefficient variable,  $y$  is the phase change variable, and  $z$  is the amplitude variable.

Coefficients	Value
A	7.92
$\alpha_1$	-2.77
$\alpha_2$	-2.69
$\alpha_3$	-9.41
$\alpha_4$	0.529
$\alpha_5$	-5.19
$\alpha_6$	10.0
$\alpha_7$	6.21
$\alpha_8$	0.670
$\alpha_9$	3.50

Table 4. Coefficients for second order multivariate regression

Using the diagnostic model described above, detected crack length for various specimen can be obtained. Table 3. shows experimentally measured crack lengths and the detection results for the two specimens (T11 and T9). It can be observed that detection results show good agreement with ground truth crack length measurements in general. However, there are uncertainties in these measurements and diagnostic estimation which show up as differences in the two estimates. A similar behavior was observed for all other specimens. .

T11		T9	
Constant loading		Variable loading	
Measured Crack length(mm)	Detected Crack length(mm)	Measured Crack length(mm)	Detected Crack length(mm)
0	0.257	0	0.257
1.61	0.834	1.94	2.77
2.17	1.71	2.5	3.01
2.74	2.71	3.71	3.79
3.13	2.99	3.88	4.40
4.06	4.38	4.61	4.79
4.96	4.90	4.96	4.75
7.24	6.30	5.52	5.01

Table 3. The experimentally measured crack length and detected result for the two specimens

### 3. DAMAGE PROGNOSIS USING BAYESIAN UPDATING

Following the development of the damage diagnosis model, the detected crack growth data are combined with crack propagation law for the predicting remaining life. A small time scale model (Lu 2010) is briefly discussed next, which was used as the mechanism model for fatigue crack propagation. Bayes theorem based prognostic method is then employed to incorporate diagnostic estimation and fatigue crack growth model.

#### 3.1. Small Time Scale Model

This model defines the fatigue crack kinetics at any arbitrary time instant ( $t+dt$ ) within a loading cycle. As shown in Fig. 8, the crack face will extend by  $da$  for a small temporal increment  $dt$  during the loading cycle (Lu and Liu 2010). The crack growth after a given life span ( $\Delta t$ ) can be obtained through direct integration over time. This model was proposed based on the geometric relationship between the crack tip opening displacement (CTOD) and the instantaneous crack growth kinetics. An illustration of this geometric relationship is shown in Fig. 9. For a given time increment, the relationship between the CTOD increment  $d\delta$  and crack increment  $da$  can be expressed as

$$\frac{da}{dt} = \frac{ctg\theta}{2} \frac{d\delta}{dt} \quad (2)$$

where  $\theta$  is the crack tip opening angle (CTOA). However, this formulation does not consider material property changes at the crack tip during previous loading cycles. To incorporate that, the relationship between  $da$  and  $d\delta$  was modified by (Zhang and Liu 2012). This model is also

applicable to random amplitude loading in general and is given as

$$\frac{da}{dt} = \frac{AK_{\max}\Delta K d\delta}{2\sqrt{\delta} dt} \quad (3)$$

where A is a calibration parameter for a given specimen,  $\Delta K = \Delta\sigma\sqrt{\pi a}Y$  and Y is the geometry correction function. A detailed derivation of this formulation can be found in (Lu and Liu 2010).

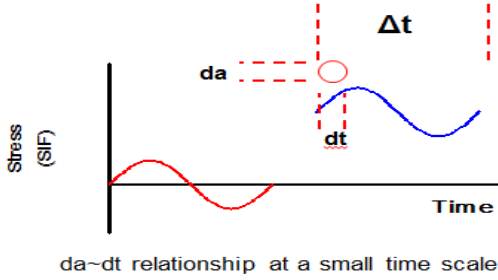


Figure 8. da~dt relationship at a small time scale

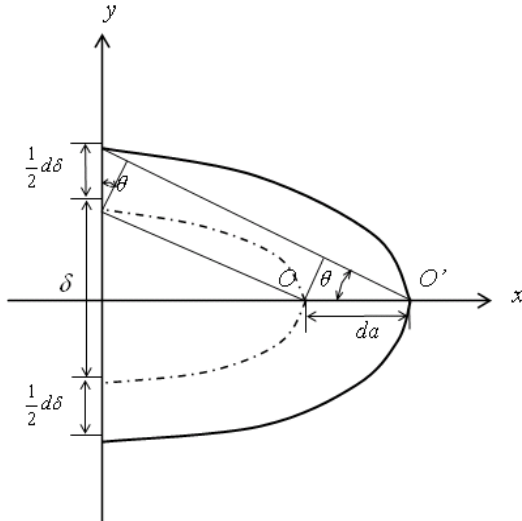


Figure 9. A schematic illustration of the geometric relationship for crack tip

### 3.2. Bayesian updating and uncertainties quantification

Even though the introduced small time scale model can express the fatigue crack growth under variable amplitude loading, this deterministic model cannot handle the uncertainties involved in the fatigue crack growth. Maximum Relative Entropy (MRE) was first applied by (Guan, Jha et al. 2009) in fatigue problems to describe the stochastic process of the fatigue crack growth mechanism. Model parameters are considered as random variables, which can be updated using the in-situ measurements. The posterior distribution of the parameters can be expressed as

$$q(\theta) \propto p(\theta)p(x'|\theta)\exp(\beta g(\theta)) \quad (4)$$

where  $p(\theta)$  is the prior distribution of the model parameter, in which  $\theta$  can be vector,  $p(x'|\theta)$  is the likelihood function,  $q(\theta)$  is the posterior distribution of model parameter,  $\exp(\beta g(\theta))$  is the term introduced by the moment constraint. The detailed derivation of Eq.(4) can be found in al(Guan, Jha et al. 2009). It can be seen that, if there is no moment constraint (i.e.,  $\beta = 0$ ), MRE method is identical with traditional Bayesian updating. In Bayesian updating, model uncertainties and measurements uncertainties are incorporated in the construction of the likelihood function. Let  $x'$  be a system measurement,  $M(\theta)$  is the prediction value based on physic-model. If there is no measurement noise and model parameter uncertainties, the measurement would be identical with the model prediction, i.e.  $x' = M(\theta)$ . However, this is not the case here. If the measurement noise  $\epsilon$  and model uncertainties  $\tau$  are considered, the relationship between  $x'$  and  $M(\theta)$  can be expressed as

$$x' = M(\theta) + \epsilon + \tau \quad (5)$$

Assume the two error term  $\epsilon$  and  $\tau$  are independent zero mean normal variables (Bell 2001; Adams 2002), the sum of them can be expressed as a new random variable  $e = (\epsilon + \tau) \sim N(0, \sigma_e)$ . Therefore, the likelihood function  $p(x'|\theta)$  can be expressed as

$$p(x'_1, x'_2, \dots, x'_n | \theta) = \frac{1}{(\sqrt{2\pi}\sigma_e)^n} \exp\left(-\frac{1}{2}\sum_{i=1}^n \left(\frac{x'_i - M(\theta)}{\sigma_e}\right)^2\right) \quad (6)$$

Where n is the number of available measurements. Substituting Eq.(6) into Eq.(4), the posterior distribution of parameter  $\theta$  is expressed as

$$p(\theta|x'_1, x'_2, \dots, x'_n) \propto p(\theta) \frac{1}{(\sqrt{2\pi}\sigma_e)^n} \exp\left(-\frac{1}{2}\sum_{i=1}^n \left(\frac{x'_i - M(\theta)}{\sigma_e}\right)^2\right) \quad (7)$$

In Eq. (7),  $M(\theta)$  can be replaced by the small time scale model introduced earlier. The crack length can be calculated using given parameter inputs. It is not easy to get an analytical solution for Eq. (7) if the parameter vector is high dimension. Therefore, a sample technique, such as Markov chain Monte Carlo (MCMC) is adopted. Following the Bayesian updating, the uncertainties in model parameters are reduced and result in improved remaining useful life prediction performance.

## 4. DEMONSTRATION EXAMPLES

### 4.1. Constant Loading Case

To precisely predict the fatigue life, an estimate of initial crack length is needed as input to the crack growth model. However, the initial crack length for the lap joint is often undetectable initially and therefore an equivalent initial flaw size (EIFS) concept (Liu and Mahadevan 2009) is applied to calculate the crack length for a given number of load cycles.

Ideally EIFS should be properly chosen to match as close as possible the true crack growth area. We consider both EIFS and A in Eq. (3) as the parameters to be updated. The initial guess about A is obtained from training data from other specimens (T1, T2, T3, T5 and T6 ) and the initial estimate of EIFS is obtained by the model proposed in (El, Topper et al. 1979). The equation estimating the EIFS is given as

$$a = \frac{1}{\pi} \left( \frac{\Delta K_{th}}{\Delta \sigma_f Y} \right)^2 \quad (8)$$

where Y is a geometry correction factor depending on specimen geometry and crack configuration,  $\Delta \sigma_f$  is fatigue limit, and  $\Delta K_{th}$  is the threshold stress intensity factor. For lap joints, Y should incorporate the multiple crack condition, because of other minor cracks that may appear randomly near other holes. Corresponding detailed solution for Y is given in (hijazi, Smith et al. 2004).

Based on the prognostic method described above and detected crack length from the diagnosis module in Table 3, Bayesian updating is performed to update the prediction results and the model parameters. The experimentally measured crack length is considered as ground truth data. Fig. 10 shows the prior belief and two set of data and Fig. 11 shows the updated results from the proposed prognosis method. Blue solid line is the median prediction using the prior distribution. Hollow rectangular points are the optically measured crack length and are considered the true crack length. Black solid points are the crack estimation from the Lamb wave-based damage detection method. It can be seen that measurement uncertainties exist and influence the estimated crack length, which is different from the true crack length.

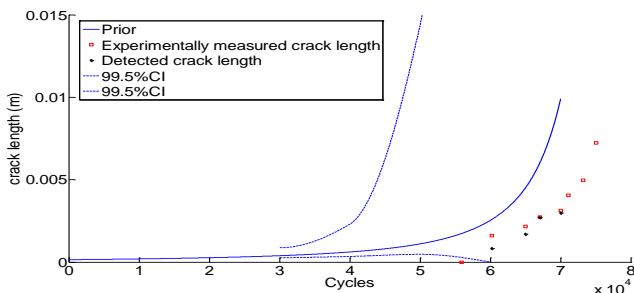
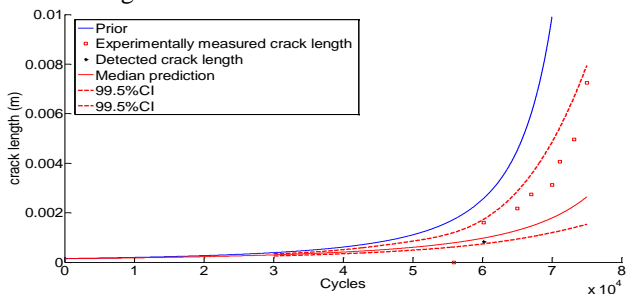
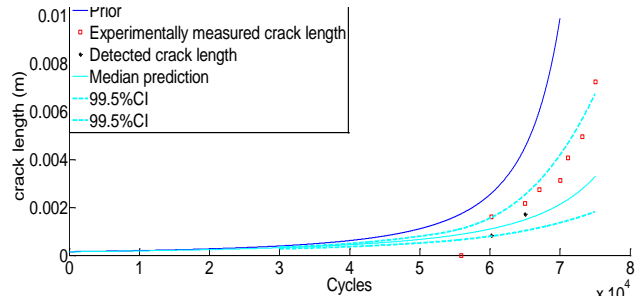


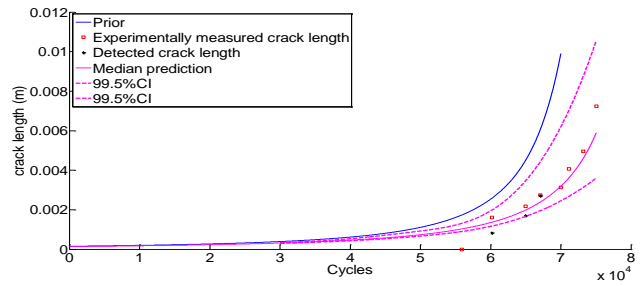
Figure 10. Prior belief and dataset for T11



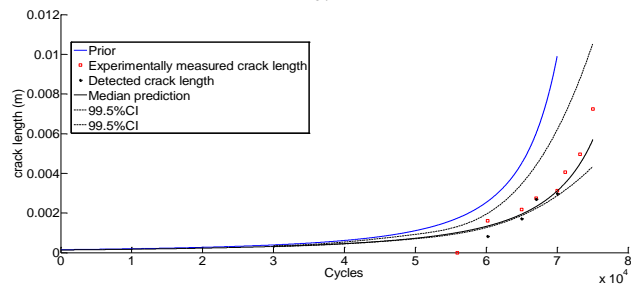
a.



b.



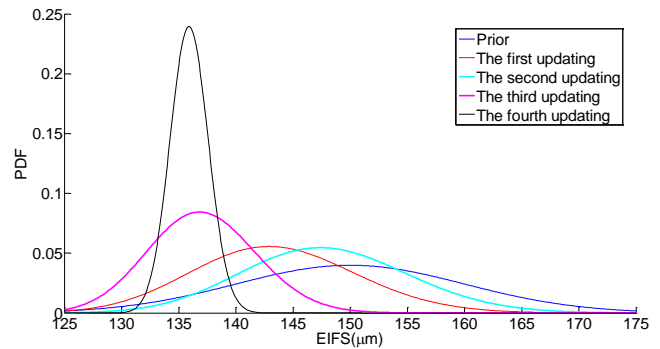
c.



d.

Figure 11. Bayesian updating result. a), One detected data. b), Two detected data. c), Three detected data. d), Four detected data.

From Fig. 11, it can also be seen that the median prediction gets closer to the true crack length with incremental updates from detection data. The uncertainty bounds become narrower with additional updating, which indicate the effectiveness of the Bayesian updating method in reducing prognostic uncertainties. This trend can also be observed in the updated parameter distribution, shown in Fig. 12.



a.

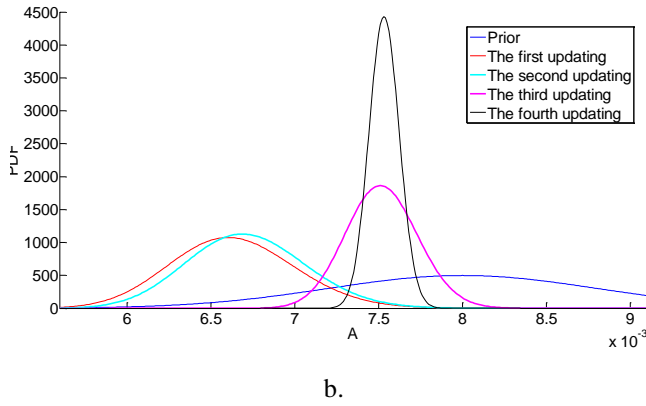


Figure 12. Parameter updating result. a), Updated EIFS. b), Updated A

### 4.2. Variable Loading Case

Following the same procedure as described in Section 4.1, the prognosis is carried out for variable amplitude loading. Fig. 13 shows the prior belief and two sets of data (i.e., crack length measurement from optical microscope and those from Lamb wave detection). Fig. 14 shows the updated results using the proposed prognosis method.

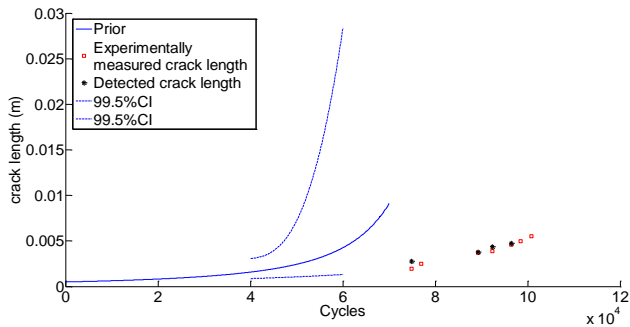


Figure 13. Prior belief and dataset for T9

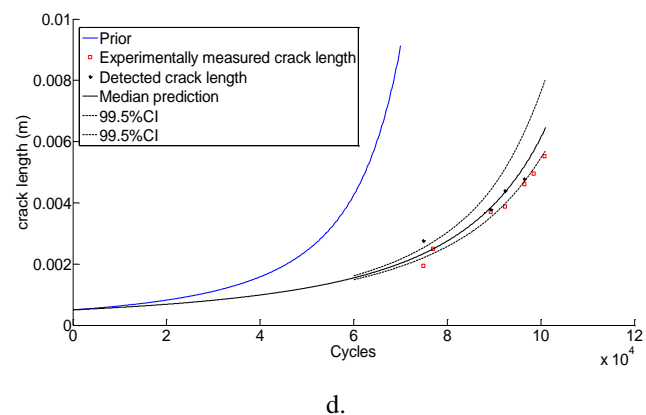
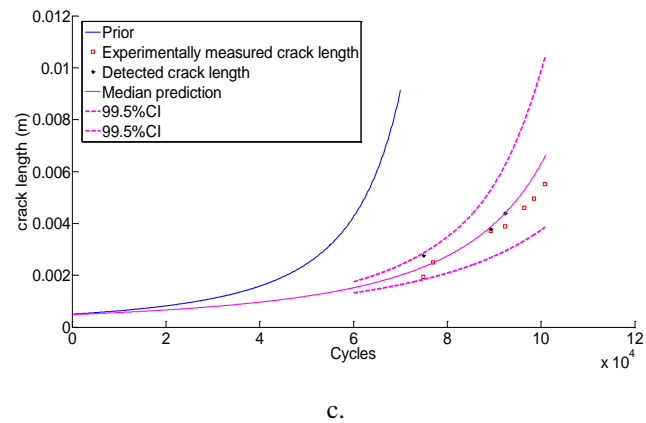
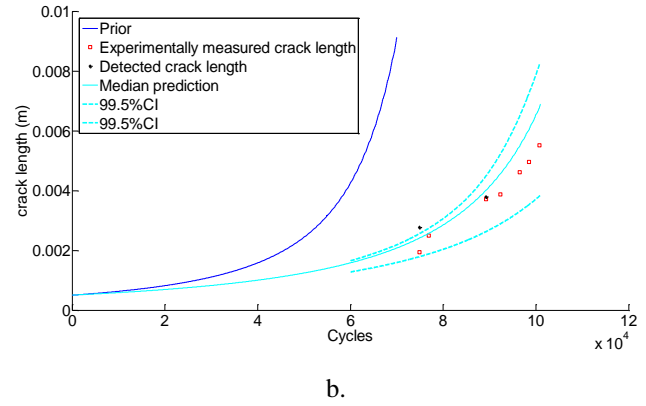
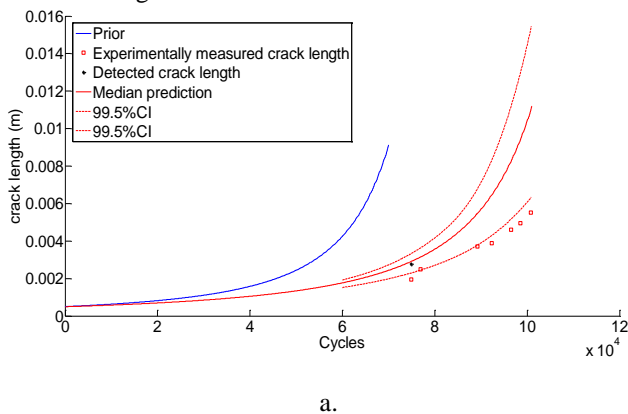
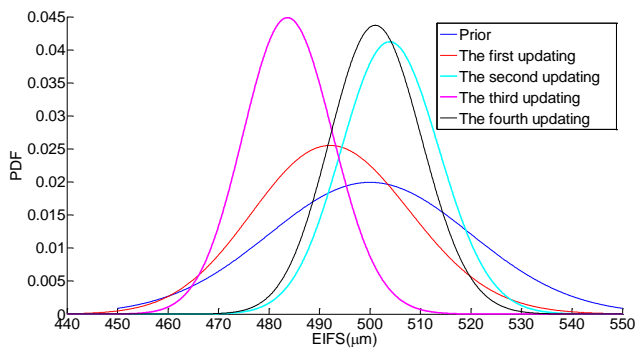


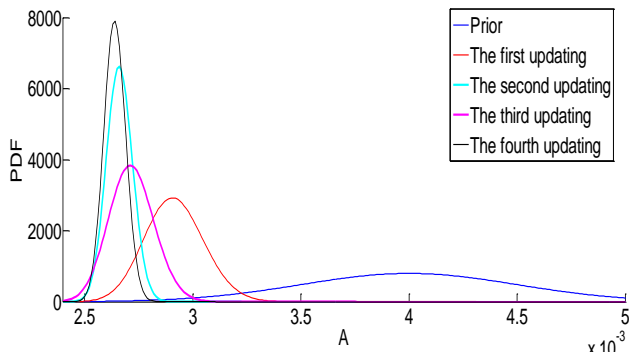
Figure 14. Bayesian updating results with. a), One detected data point. b), Two detected data points. c), Three detected data points. d), Four detected data points.

From the figure above, a similar trend can be observed for the updated crack growth trajectory as well as for the constant amplitude loading case. Fig. 15 shows the updated parameter distribution. In Fig.15a, the shift between the third and fourth updating is because of the uncertainty of the detected crack size, i.e. measurement error. A further shift is possible for the fifth updating, but the uncertainty of updated parameter distribution won't change too much. The shift is also important as it represents the mean prediction. The tightening of curves reflects the uncertainty decrease. It

is possible to move from inside of 90% to outside of 90% due to the variable loading.



a.



b.

Figure 15. Parameter updating result. a), Updated EIFS. b), updated A

### 4.3. Prognostic Performance Evaluation

To evaluate the performance of the prognostic model, prognostic metrics are employed. A detailed discussion of metrics-based model validation can be found in (Saxena, Celaya et al. 2008). Several relevant metrics, such as Prognostic Horizon (PH),  $\alpha - \lambda$  Accuracy, Relative Accuracy (RA), Cumulative Relative Accuracy (CRA), and Convergence are discussed in that publication. In this paper here, Prognostic Horizon and  $\alpha - \lambda$  Accuracy are used to assess prognostic algorithms performance. The Prognostic Horizon describes the length of time before end-of-life (EoL) when a prognostic algorithm starts predicting with desired accuracy limits. The limit is expressed using an  $\alpha$ -bound given by  $\pm \alpha \cdot t_{EoF}$ . In contrast,  $\alpha - \lambda$  Accuracy determines whether prediction accuracy is within desired accuracy levels (specified by  $\alpha$ ) at a given time. We present the results for constant loading and variable loading in the next figures.

- Constant loading prediction performance assessment

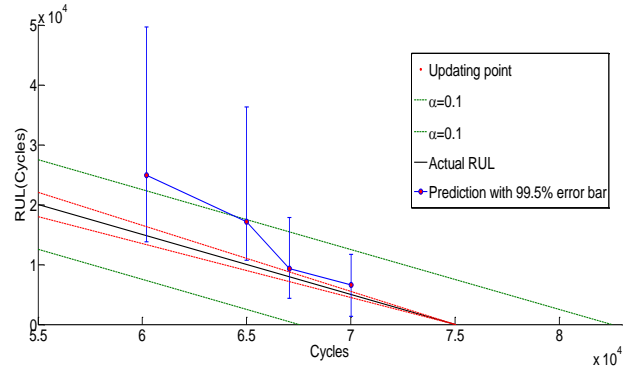


Figure 16. Prognostic performance for PH and  $\alpha - \lambda$  accuracy

Fig. 16 shows that 99.5% RUL interval prediction enters the 90% accuracy zone at the third updating, so the proposed prognostic method can provide a satisfactory prediction of RUL.

- Variable loading prognostic assessment

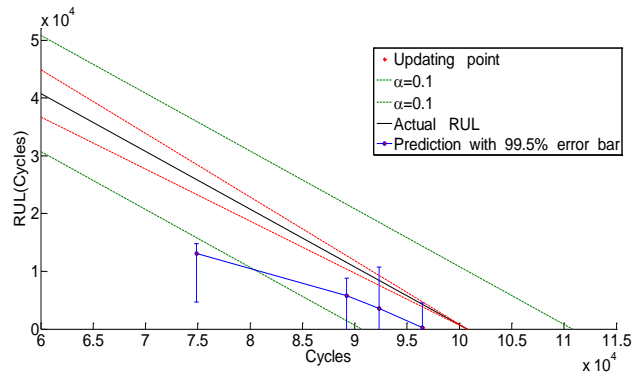


Figure 17. Prognostic performance for PH and  $\alpha - \lambda$  accuracy

From Fig. 17, the same trend can be observed. The 99.5% RUL interval prediction enters the 90% accuracy zone at the fourth updating. Satisfactory result is obtained for the variable loading case.

With additional updating, the median prediction gets closer to the true RUL and the confidence bound is reduced as given more information. Therefore, the updating result can provide an informative RUL prediction at earlier stage of the whole lifecycle.

### 5. CONCLUSION

In this paper, an integrated fatigue damage diagnostics and prognostics methodology is proposed, which combines a piezoelectric sensor network-based damage detection method, physics-based fatigue crack propagation model, and a Bayesian updating framework. The proposed method is demonstrated and validated using fuselage lap joints test

data. Based on the results obtained, several conclusions can be drawn:

1. Individual feature is not adequate for accurate extrapolation from training data to validation data and therefore, feature fusion among different features can be used to reduce the uncertainties in the damage detection;
2. By properly interpreting the features changes of the received signal, the proposed diagnostic model provides a reasonable estimate of the crack length in lap joints. There is some uncertainties inherent in the crack length detection that must be incorporated in predictions.
3. The small time scale model can be successfully applied to both constant loading and variable amplitude loading cases.
4. Bayesian updating can represent and manage the uncertainties introduced by model parameters and measurement noise. The accuracy of model prediction increases as more data are included in the updating. The performance of the proposed prognostic method is assessed by the given prognostics metrics and satisfactory results are obtained.

The current work focused on the damage size estimation for a known damage location identified based on hot spot identification (and consequent sensor placement). Further efforts are required to provide capability for more general crack location methods and to extend the proposed methodology to other structural components. Last but not the least, this work assumed presence of a single major crack for predictions. Further studies are required to consider multiple site cracks and their interactions.

#### ACKNOWLEDGEMENT

The research reported in this paper was supported by the NASA ARMD/AvSP IVHM and SSAT projects under NRANX09AY54A. The support is gratefully acknowledged.

The authors also thank the in-kind support for testing specimens and technical discussions with Dr. Min Liao at NRC Canada and Mr. Mike Venti of NASA Dryden Flight Research Center for providing and manufacturing additional test specimen.

#### REFERENCES

Adams, M. Thomas (2002) "G104-A2L Guide for estimation of measurement uncertainty in testing" American Association of Laboratory Accreditation Manual: 10-18.

Bell, Stephanie (2001) "A Beginner's Guide to Uncertainty of Measurement" The National Physical Laboratory (2): 9-16.

Chong, K. P. (1999). "Health monitoring of Civil structures " *J. Intell. Mater. Syst. Struct* 9(11): 892-898.

Constantin, N., S. Sorohan, et al. (2011). "Efficient and low cost PZT network for detection and localization of damage in low curvature panels." *Journal of Theoretical and Applied Mechanics* 49(3): 685-704.

El, H. M., T. Topper, et al. (1979). "Prediction of nonpropagating cracks." *Eng. Fract. Mech.* 11: 573-584.

Elber, W. (1970). "Fatigue crack closure under cyclic tension." *Eng. Fract. Mech.* 21: 37-45.

Forman, R. (1967). "Numerical analysis of crack propagation in cyclic-loaded structures." *J Basic Eng* 89: 459-464.

Giurgiutiu, V. (2003). "Lamb wave generation with piezoelectric wafer active sensors for structural health monitoring." *Smart Structures and Materials* 5056: 111-122.

Giurgiutiu, V. (2005). "Tuned Lamb wave excitation and detection with piezoelectric wafer active sensors for structural health monitoring." *Journal of Intelligent Material Systems and Structures* 16(4): 291.

Giurgiutiu, V., A. Zagari, et al. (2002). "Piezoelectric wafer embedded active sensors for aging aircraft structural health monitoring." *Structural Health Monitoring* 1(1): 41-61.

Guan, X., R. Jha, et al. (2009). "Probabilistic fatigue damage prognosis using maximum entropy approach." *Journal of Intelligent Manufacturing*: 1-9.

Guan, X., R. Jha, et al. (2011). "Model Selection, Updating and Averaging for Probabilistic Fatigue Damage Prognosis." *Structural Safety* 33(3): 242-249.

hijazi, A. L., B. L. Smith, et al. (2004). "Linkup strength of 2024-T3 bolted lap joint panels with multiple-site damage." *Journal of Aircraft* 41(2): 359-364.

Ihn, J.-B. and F.-K. Chang (2004). "Detection and monitoring of hidden fatigue crack growth using a built-in piezoelectric sensor/actuator network: I. Diagnostics." *Smart Mater. Struct.* 13: 609-620.

JR, C. H. K. and F.-K. Chang (1993). " Identifying delamination in composite beams using built-in piezoelectrics." *J. Intell. Mater. Syst. Struct* 6: 649-672.

Kazys, R. and L. Svilainis (1997). "Ultrasonic detection and characterization of delaminations in thin composite plates using signal processing technique." *Ultrasonics* 35: 367-383.

Kitagawa, H. and S. Takahashi (1979). "Applicability of fracture mechanics to very small cracks or cracks in the early stage.ASM." In: *Proceedings of the second international conference on mechanical behaviour of materials*: 627-631.

Koruk, M., & Kilic, M. (2009). "The usage of IR thermography for the temperature measurements inside an automobile cabin." *International Communication in Heat and Mass Transfer* 36(872-877).

- Laird, C. (1979). "Mechanisms and theories of fatigue." *Fatigue Microstruct.*: 149-203.
- Lemistre, M. and D. Balageas (2001). "Structural health monitoring system based on diffracted Lamb wave analysis by multiresolution processing." *Smart materials and structures* 10: 504.
- Liu, Y. and S. Mahadevan (2009). "Probabilistic fatigue life prediction using an equivalent initial flaw size distribution." *International Journal of Fatigue* 31(3): 476-687.
- Lu, Z. and Y. Liu (2010). "Small time scale fatigue crack growth analysis." *International Journal of Fatigue* 32(8): 1306-1321.
- Masri, S. F., L. H. Sheng, et al. (2004). "Application of a web-enabled real-time structural health monitoring system for civil infrastructure systems." *Smart MATER. STRUCT.* 13(6): 1269-1283.
- Monkhouse, R., P. Wilcox, et al. (1997). "Flexible interdigital PVDF transducers for the generation of Lamb waves in structures." *Ultrasonics* 35(7): 489-498.
- Nicoletto, G., G. Anzelotti, et al. "X-ray computed tomography vs. metallography for pore sizing and fatigue of cast Al-alloys." *Pocedia Engineering* 2: 547-554.
- Paris, P. and F. Erdogan (1963). "A critical analysis of crack propagation laws." *Journal of Basic Engineering, Transactions of the American Society of Mechanical Engineers*: 528-534.
- Raghavan, A. and C. E. S. Cesnik (2007). "Review of guided-wave structural health monitoring." *Shock and Vibration Digest* 39(2): 91-116.
- Ritchie, R. and J. Lank (1996). "Small fatigue cracks: a tatement of the problem and potential solutions." *Mater. Sci. Eng.* 84: 11-16.
- Santoni, G. B., L. Yu, et al. (2007). "Lamb wave-mode tuning of piezoelectric wafer active sensors for structural health monitoring." *Journal of Vibration and Acoustics* 129: 752.
- Saxena, A., J. Celaya, et al. (2008). "Metrics for evaluating performance of prognostic techniques." In *Aerospace conference, 2009 IEEE*: 1-13.
- Scalea, d., L. Francesco, et al. (2002). "Guided wave ultrasonics for NDE of aging aircraft components " *Proc. SPIE* 4704: 123-132.
- SU, Z. and L. Ye (2009). "Identification of damage using Lamb waves." *Springer LNACM* 48: 195-254.
- Wang, C. H., J. T. Rose, et al. (2004). "A synthetic time-reversal imaging method for structural health monitoring." *J. of smart mater. Struct.* 13: 413-423.
- Wang, Q. and S. Yuan (2009). "Baeline-free imaging method based on new pzt sensor arrangements." *Journal of Intelligent Material Systems and Structures* 20(1663-1673).
- Ward, M. D. and D. A. Buttry (1990). "In situ interfacial mass detection with piezoelectric transducers." *Science* 249(4972): 1000.
- Zhang, W. and Y. Liu (2012). "Investigation of incremental fatigue crack growth mechanisms using in situ SEM testing." *Internatinal Journal of Fatigue* 42: 14-23.
- Zhao, X., H. Gao, et al. (2007). "Active health monitoring of an aircraft wing with embedded piezoelectric sensor/actuator network.I.Defect detection, localization and growth monitoring " *Smart MATER. STRUCT.* 16(1208-17).

## BIOGRAPHIES

**Tishun Peng** is a Ph.D. student in the department of civil & environmental engineering at Clarkson University. He received his Bachelor's degree in civil and environmental engineering at Harbin Institute of Technology, China in 2011. His research interests are Bayesian updating and applications, structure diagnostic and probability prognostic, Markov Chain Monte Carlo simulation and reliability.

**Jingjing He** is a graduate research assistant in the department of civil and environmental engineering at Clarkson University. Her research interests are fatigue analysis, structural dynamics, diagnosis and prognosis. She received her B.S. degree in reliability engineering and M.S. degree in aerospace system engineering from Beihang University in China in 2005 and 2008, respectively.

**Yongming Liu** is an associate Professor in the department of civil & environmental engineering. He completed his Bachelors` and Maters` degrees from Tongji University in China and obtained his Ph.D. at Vanderbilt University. His research interests include fatigue and fracture analysis of metals and composite materials, probabilistic methods, computational mechanics, and risk management. Dr. Liu is a member of ASCE and AIAA and serves on several technical committees on probabilistic methods and advanced materials.

**Abhinav Saxena** is a Research Scientist with Stinger Ghaffarian Technologies at the Prognostics Center of Excellence NASA Ames Research Center, Moffet Field CA. His research focus lies in developing and evaluating prognostic algorithms for engineering systems using soft computing techniques. He is a PhD in Electrical and Computer Engineering from Georgia Institute of Technology, Atlanta. He earned his B. Tech in 2001 from Indian Institute of Technology (IIT) Delhi, and Masters` Degree in 2003 from Georgia Tech. Abhinav has been a GM manufacturing scholar and is also a member of IEEE, AAAI and ASME.

**Jose R. Celaya** is a staff scientist with Stinger Ghaffarian Technologies at the Prognostics Center of Excellence, NASA Ames Research Center. He received a Ph.D. degree in Decision Sciences and Engineering Systems in 2008, a

M. E. degree in Operations Research and Statistics in 2008, a M. S. degree in Electrical Engineering in 2003, all from Rensselaer Polytechnic Institute, Troy New York; and a B.S. in Cybernetics Engineering in 2001 from CETYS University, Mexico.

**Kai Goebel** received the degree of Diplom-Ingenieur from the Technische University Munchen, Germany in 1990. He received the M.S. and Ph.D. from the University of California at Berkeley in 1993 and 1996, respectively. Dr. Goebel is a senior scientist at NASA Ames Research Center where he is the deputy area lead for Discovery and Systems Health in the Intelligent Systems division. In addition, he directs the Prognostics Center of Excellence and he is the Associate Principal Investigator for Prognostics of NASA's Integrated Vehicle Health Management Program. He worked at General Electric's Corporate Research Center in Niskayuna, NY from 1997 to 2006 as a senior research scientist. He has carried out applied research in the areas of artificial intelligence, soft computing, and information fusion. His research interest lies in advancing these techniques for real time monitoring, diagnostics, and prognostics. He holds fifteen patents and has published more than 200 papers in the area of systems health management.



# A Cost-Benefit Approach to Evaluating Engine Health Monitoring Systems

Grant Gordon<sup>1</sup>, Chris Hickenbottom<sup>2</sup>, and Dinkar Mylaraswamy<sup>3</sup>

<sup>1,2</sup> *Honeywell International Inc., 111 S. 34th St., P.O. Box 52181, Phoenix, AZ 85072-2181, U.S.A.  
Grant.Gordon@Honeywell.com  
Chris.Hickenbottom@Honeywell.com*

<sup>3</sup> *Honeywell International Inc., 1985 Douglas Drive N, Golden Valley, MN 55422, U.S.A.  
Dinkar.Mylaraswamy@Honeywell.com*

## ABSTRACT

Condition-based maintenance (CBM) enables fleet-level decisions that increase readiness, increase time between overhaul (TBO) and reduce inspections. Since engines account for a significant portion in overall maintenance cost drivers, detection of incipient faults is an important element of the overall CBM equation. The last few years have seen significant progress in design, development and deployment of engine health monitoring. In order for such potential health monitoring solutions to be operationally viable, they must integrate with existing engine designs and maintenance processes. That is, technical factors must be balanced against economic and operational benefits.

A Cost-Benefit Analysis (CBA) is used to provide a comparison of alternative solutions that decision-makers can use to identify the most cost-effective approach to CBM. In this paper we describe our approach to developing the underlying value capture expressions for monetizing cost and benefits. We illustrate the approach to evaluate two options for mechanical components health monitoring techniques for a gas turbine engine. We conclude the paper with how CBA summary results can be presented to a decision maker.

## 1. INTRODUCTION

The last few years have seen significant progress in design, development and deployment of engine health monitoring solutions (Jaw 2005, Mylaraswamy et al. 2009). This progress is exemplified both with respect to novel sensor development (Uluyol 2010), accurate algorithms and increased coverage of engine components being monitored

(Parthasarathy 2011). In order for such potential health monitoring solutions to be operationally viable, they must integrate with existing engine designs and maintenance processes. The technical factors must be balanced against economic and operational benefits. In this paper a Cost-Benefit Analysis (CBA) analysis approach is used to show how competing solutions can be compared to identify the most effective engine health monitoring design for a new development engine.

## 2. THE APPROACH

At the heart of any CBA is a cost model, with an underlying cash flow structure comprised of elements that work towards (benefits) or against (costs) the business objectives. Commonly there are two approaches to developing a cost model: resources based and activity based (Schmidt, 2009). For a new engine development program, considering condition based maintenance (CBM) capabilities, a resources-based approach is appropriate. Similar to a component improvement program where a new engine component must buy its way onto the platform, a CBM system is intended to bring benefits but is also expected to have its own useful life, operating and maintenance costs that must be compared relative to its benefits. Our analysis proceeds by comparing two different scenarios in a cash flow model, an 'as-is' engine design e.g. no CBM and the 'to-be' configuration e.g. a CBM system with a suite of capabilities.

In our analysis the cost elements follow a structure similar to that provided in the Report of the OSD CBM+ Action Group 2010 Summer Study (Secretary of Defense, 2010). At the top level the cost elements include research and development, infrastructure investment, operation and sustainment of the technology and final disposal. We explicitly track and monetize a cost for the additional weight incur for the engine integrated elements of the technology solution. In addition we have made extensive use of a Cost

---

Gordon et al.. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Effectiveness Analysis (CEA) Spreadsheet (US Army 2007) to define and evaluate costs. CEA is a standardize tool developed by the Joint Propulsion Coordinating Committee (JPCC) for evaluating the cost effectiveness of an engineering change to aircraft engines.

The cost is broken down into four categories:

1. Non-recurring Development Cost.
  - a. A one-time cost to mature the technology from its current technology readiness level (TRL) to TRL-7.
  - b. A one-time cost to develop the necessary CBM infrastructure ground support equipment, CBM servers, engine tests, and common software development.
2. Unit cost for deployment.
  - a. A PHM technology may require special sensors, signal processing units and cables. The unit cost for a PHM technology includes the cost to deploy the sensor (if applicable) and any specialized software on the engine.
  - b. In addition there will be a cost for installing the any commonly used processing hardware and common software to support the Engine PHM functions on the aircraft.
3. Sustainment cost. The sustainment cost is estimated:
  - a. One block change per year that includes a major software modification.
  - b. Software configuration changes to update lookup-table, threshold and algorithm settings. We estimate each change will consume 2 man-hours. In the first year of deployment we estimate 48 changes, 24 in the second year and 12 in subsequent years.
4. Added weight was accounted for as a decrease in specific fuel consumption.

More challenging than characterizing the system costs over time is the difficulty of effectively characterizing the CBM system benefits. In this case the true value of any new technology should be measured against a set of metrics that map back to operational and mission effectiveness. As a starting point we used the Army’s top level CBM objectives (US Army 2012):

1. Decrease Maintenance Burden on the Soldier.
2. Increase Platform Availability and Readiness
3. Enhance Safety
4. Reduce Operations & Support (O&S) Costs

These objectives cover traditionally financial and non-financial goals. Despite the fact that these non-financial benefits are indeed very important – the Army states that “the most compelling and supportable benefits described in Cost Benefit Analysis (CBA) are those associated with aviator safety and aviation combat power” (Secretary of Defense, 2010) – there is often a reluctance to assign monetary values to non-financial benefits. But clearly the

cash flow statement is blind to nonfinancial business outcomes so defining an acceptable value is necessary.

In this paper we focus on monetizing one such objective – namely operational availability caused by downtime and inefficient operations. In order to define the underlying expressions, key system boundaries and assumptions, which can be interpreted as our *cost model parameters*, need to be defined. These are listed in Table 1.

Model Parameters	
Number of engines and aircrafts	Unit-level maintenance man-hour rate
Year of introduction of production engines	Depot maintenance man-hour rate
Engine introduction rate	Average cost of fuel
Average yearly flying hours (FH)	Density of fuel (kerosene)
Average mission length (flight hours)	Analysis period in years

Table 1. Parameters in the cost-benefit analysis

In addition, our analysis considers the rate at which users replace existing maintenance paradigms with new technologies. This adoption rate is expressed through a constant,  $\alpha$ , as shown in Table 2.

Year	Adopter Category	Cumulative Acceptance Fraction, $\alpha$
1	Innovators (2.5% of population)	0.025
3	Early Adopters (13.5% of population)	0.16
4	Early Majority (34% of population)	0.5
5	Late Majority (34% of population)	0.84
10	Laggards (16% of population)	1.0

Table 2. Adoption rate of maintenance policies due to introduction of CBM technologies

### 3. QUANTIFYING THE CBM BENEFITS

The objectives of decreasing the Maintenance Burden on the Soldier and reducing Operations & Support (O&S) Costs are financial benefits that are relatively easy to account for in a cash flow model. For example we model the benefit of decreased maintenance burden by considering two sources of activities: (a) scheduled preventative maintenance activities, and (b) troubleshooting activities arising from inconclusive diagnostic indications. Given a frequency of occurrence and cost per occurrence these sources can be evaluated. However increases in aviator safety and aviation combat power (increase platform availability and readiness)

are responsible for the majority of a CBM system value. These are not direct financial benefits and are more challenging to model. But given their significance and complexity, this is where we will focus our attention in the paper.

For readiness and availability we monetize the benefits by first establishing how the CBM capabilities can influence Ao. From the definition of operational availability we have (U.S. Army, 1996),

$$Ao = OR = \frac{(OT+ST)}{(OT+ST) + (TCM+TPM+TALDT)} \quad (1)$$

here Ao is operational availability, OR is operational readiness, OT is operating time, ST is standby time, TCM is total corrective maintenance downtime, TPM is total preventative maintenance downtime and TALDT is total administrative and logistics delay time. By definition, if a CBM capability can reduce TCM, TPM or TALDT, then operational availability can be improved.

To assign a value to the fractional increases in Ao we note that increases in Ao are increases in the fractional utility of the asset, where the total utility is the total satisfaction that the user receives over the life of the asset. Given that the total satisfaction must be greater or equal to the total costs for acquiring, operating, maintaining and disposing the asset, a value can be assigned to Ao based on the flight hour costs,  $FH_{cost}$ , for the helicopter. The cost per flight hour is calculated as follows:

$$\begin{aligned} FH\_value &\geq FH\_cost \\ &= AAC\_aircraft + DMC\_Engine + DMC\_Aircraft \\ &+ Fuel\ cost\ per\ FH \end{aligned} \quad (2)$$

Here  $AAC_{aircraft}$  is the average acquisition cost per flight hour.  $DMC_*$  is the average direct maintenance cost per flight hour. The average acquisition cost AAC is calculated as:

$$Avg\ Acq\ Cost = \frac{Aircraft\ average\ acquisition\ cost}{Expected\ aircraft\ lifetime\ in\ FH} \quad (3)$$

From reference sources, e.g. (Katzomis, 1998), for the Apache and Blackhawk helicopter

$$\begin{aligned} Acquisition\ Cost_{Apache} &= \$43M \\ Acquisition\ Cost_{Blackhawk} &= \$21M \end{aligned} \quad (4)$$

And assuming an average expected lifetime for helicopters of 62,500 flight hours and a fleet mix of 30% Apaches, 70% Blackhawks, the acquisition cost for the fleet becomes:

$$AAC_{Aircraft} = \frac{0.3 * \$43M + 0.7 * \$21M}{62,500\ expected\ lifetime\ FH\ per\ aircraft} = 441\ \$/FH \quad (5)$$

Next we look at two major factors that cause loss of flight hours.

### 3.1. Unscheduled Event Model

In our model an unscheduled/unplanned maintenance occurs at the unit level in the following two conditions:

- a) A malfunction is not detected until the pre-flight inspection rendering the engine and hence the aircraft unavailable for the mission.
- b) A malfunction or an alert is generated during the mission resulting in mission abort.

The model associated with these unscheduled events is expressed as:

*Unscheduled Mx event*

$$\begin{aligned} \text{leads to} &\left\{ \begin{array}{l} Loss\ of\ flight\ hour\ due\ to\ aircraft\ downtime \\ Economic\ impact\ from\ possible\ mission\ abort \\ Safety\ value\ from\ possible\ loss\ of\ life/aircraft \\ Decrease\ in\ operational\ availability \end{array} \right\} \end{aligned} \quad (6)$$

We monetize the net impact of these unscheduled maintenance events using the following method and assumptions:

1. Administrative and logistics delay for an unscheduled event is 8 days. Using the average aircraft utilization rate (based on parameters listed in Table 1), we convert this to equivalent loss of flight hours,  $X_{TALDT}$ . The monetary value is given by:  $FH_{value} \times X_{TALDT}$
2. A mission-abort decision is made mid-way into mission. Using the average aircraft utilization rate (based on parameters listed in Table 1), we convert this to equivalent loss of flight hours,  $X_{MA}$ . The monetary value is given by:  $FH_{value} \times X_{MA}$ . We further multiply this number by the probability of occurrence,  $P(engine\ inflight\ shutdown/malfunction)$ .
3. We assume a constant value for a class A accident. Let  $M_{ACC}$  denote this constant value. This number is multiplied by  $P(class\ A\ accident/inflight\ engine\ shutdown)$ , which is usually a very small number.

If a CBM technology can provide an early indication with an adoption rate  $\alpha$  and prevent  $n_{OI}$  such events, the savings would be:

$$\begin{aligned} n_{OI} \times \alpha \times \{ &FH_{value} \times X_{TALDT} + \\ &P(engine\ inflight\ shutdown/malfunction) \times \\ &FH_{value} \times X_{MA} + \\ &P(class\ A\ accident/inflight\ engine\ shutdown) \times \end{aligned}$$

$$M\_ACC\} \tag{7}$$

Next we describe the value capture expression associated with the other factor that causes loss of flight hours.

### 3.2. Scheduled Event Model

Scheduled Preventative Maintenance, SPM, is defined as activities performed on a periodic basis to keep the aircraft ready for operations. A SPM occurs at a fixed interval  $T_{SPM}$ , may involve more than one maintainer and requires  $L_{SPM}$  man-hours of activity.  $T_{SPM}$  is expected to vary depending on OPTEMPO and mission criticality.

The model associated with a scheduled preventative maintenance event is:

$$\begin{aligned} & \text{Scheduled Mx event} \\ & \xrightarrow{\text{leads to}} \left\{ \begin{array}{l} \text{Additional burden on maintainer} \\ \text{Decrease in operational availability} \end{array} \right\} \end{aligned} \tag{8}$$

A CBM technology can potentially impact a SPM as follows: (1) It can increase the time interval between two successive scheduled preventative maintenance activities. That is,  $T_{SPM,CBM} \geq T_{SPM}$ . (2) It can decrease the time it takes to perform a scheduled preventative maintenance action. That is,  $L_{SPM,CBM} \leq L_{SPM}$ . The expression for reducing the cost associated with SPM is given by:

$$M_{SPM,CBM} = FH \times \alpha \times \left\{ \frac{L_{SPM}}{T_{SPM}} - \frac{L_{SPM,CBM}}{T_{SPM,CBM}} \right\} \times \$AVUM \tag{9}$$

Here \$AVUM is the unit level labor rate and FH is the total flight hours over the evaluation period.

### 3.3 Economic Impact of an Undetected Fault

The net impact of undesired effects resulting from an inability to detect the initial fault are described in Figure 1.

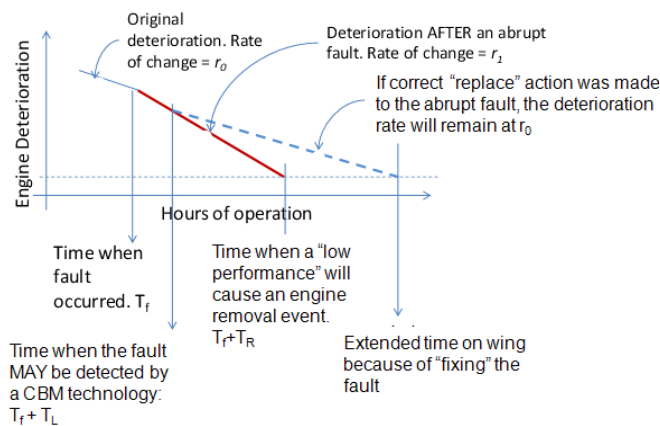


Figure 1. Impact of an undetected abrupt fault

As shown in the Figure 1, an abrupt fault happens at time  $T_f$ . Here we assume that the source of the abrupt fault is contained within a line replaceable unit LRU. Since this fault is “abrupt”, by definition it causes a sudden increase in the engine deterioration rate. The deterioration rate increases from  $r_0 \rightarrow r_1$ . Since the engine is operating at meeting its power requirements, it is operating inefficiently after the onset of the abrupt fault. This causes a jump in the specific fuel consumption SFC.

In the “to-be” scenario, we assume that we have a CBM technology that can detect the onset of the abrupt fault. This detection is done after a period of latency,  $T_L$ . The resulting benefits are expressed as:

- (a) After the replacement action, the engine continues to operate at its original deterioration rate  $r_0$ . This will increase its time on wing. This modeled as a net reduction in  $DMC_{Engine}$ .
- (b) After the replacement action, the engine continues to operate at its deterioration rate, which implies lower fuel consumption rate. This is modeled as a net reduction in net fuel savings
- (c) Secondary damage to other parts. If we can detect the abrupt fault earlier and replace it, we can save an equivalent amount of secondary damage.
- (d) Transition from an unplanned to a planned maintenance.

### 4. AN EXAMPLE – MECHANICAL HEALTH MONITORING

This example discusses how the costs and benefits of a technology referred to as Mechanical Health Monitoring (MHM) can be estimated. MHM consists of the sensors and algorithms needed to collect and fuse data from Oil Debris Monitoring (ODM) and vibration monitoring. The ODM system consists of a sensor that counts individual metallic particles and the associated signal processing electronics and software. The vibration monitoring system consists of multiple accelerometers and the hardware and software required to generate Condition Indicators (CIs) for advanced mechanical diagnostics. The MHM algorithms include a progressive on-board indicator and on-ground indications that enable maintenance to be planned when an incipient fault is detected while the engine continues to be operated. In evaluating the costs and benefits of MHM, two different options are considered. In option 1 an ODM system is compared to the chip detector (Figure 2). Option-2 includes an advanced capability of fusing ODM and vibration indicators is compared to a baseline design of a chip detector. Figure 3 illustrates the system with the addition of vibration monitoring. This comparison enables an evaluation of the additional cost of including vibration monitoring and the increased benefits that come from ODM and vibration fusion.

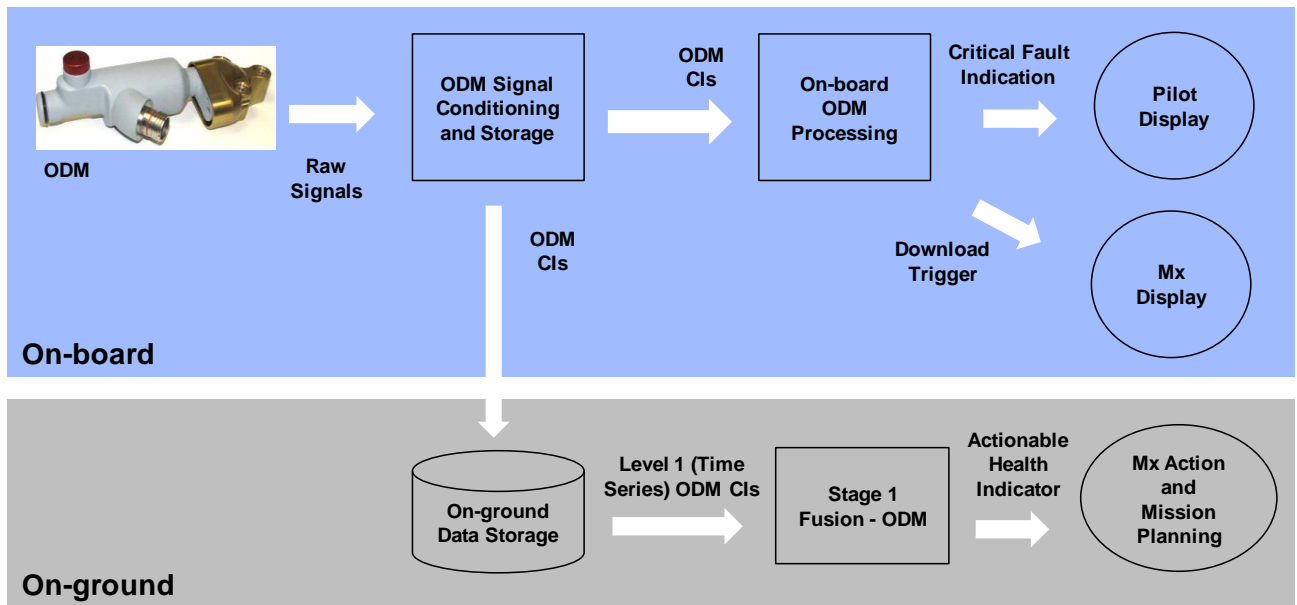


Figure 2. Option 1: ODM system

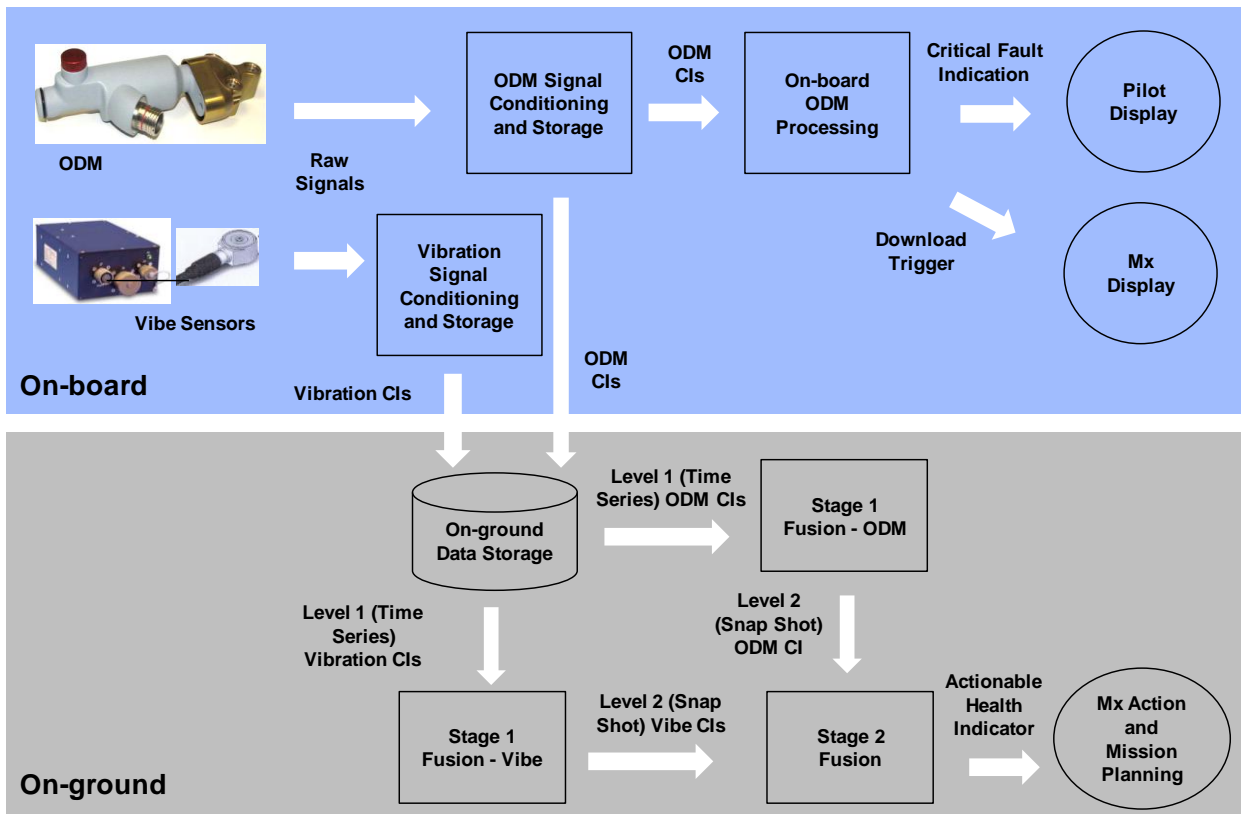


Figure 3. Option 2: ODM and vibration fusion

The benefits for each option are derived from the following value areas:

- 1) Reduction in hours spent investigating and dispositioning false chip detections
- 2) Reduction in mission aborts caused by chip indications in flight
- 3) Converting unscheduled engine replacements to planned maintenance
- 4) Increased aircraft availability due to conversion to planned maintenance
- 5) Minimized secondary damage
- 6) Converting unscheduled LRU-related maintenance actions to planned maintenance (option 2 only)
- 7) Reduction in depot repair costs through fault isolation and smart workscoping (option 2 only)

One of the significant improvements in using these systems is the operational benefits of the transition from unscheduled to planned maintenance, i.e. being able to prepare for maintenance before the aircraft has to be removed from service. This improves availability and avoids additional expenses for expediting shipping, overtime, etc. There can be large variation in how unscheduled maintenance affects operations. In some scenarios, an aircraft being removed from service can have minimal operational impact. In other scenarios, it can be very disruptive. To quantify the value of avoiding these disruptions, we estimate the average additional flight hours that an aircraft is down for unscheduled maintenance and multiply this by our cost per flight hour rate.

The key metric that enables maintenance to transition from unscheduled to planned is the amount of advanced notice (in operating hours) between high confidence detection of the incipient fault, and annunciation that the engine should no longer be operated. Option 2 is typically able to reach a given level of detection confidence earlier than option 1. This is because an independent vibration indication along with a relatively small amount of debris enables higher confidence early in the fault progression. Alternatively, with ODM only, the fault must progress farther before there is enough debris to reliably distinguish between normal and faulted conditions. To model this in the value equation, an engineering estimate is made that Option 1 will provide enough advanced notice to enable planned maintenance 50% of the time. It is assumed that Option 2 will enable planned maintenance 95% of the time. These assumptions are consistent with seeded-fault test data and analysis results. In the instances where there is not enough advanced notice to enable fully planned maintenance, the ground

system is designed to indicate the need for immediate maintenance. This case is considered to be equivalent to a present state of the art chip detector operational baseline. This difference in advanced notice is one of the primary differences between the two options, and leads to significant value that justifies the additional cost of the vibration and fusion functionality.

Figure 4 shows how the consolidated cash flow was developed for the illustrative CBM functions, MHM-I and MHM –II. The cost elements follow a cost element structure echoing the outline provided in Appendix D of the Report of the OSD CBM+ Action Group 2010 Summer Study (Secretary of Defense, 2010). At the top level, the cost elements include research and development, infrastructure investment, operation and sustainment of the technology and final disposal. We also explicitly track and monetize a cost for the additional weight incurred for the engine integrated elements of the technology solution.

## 5. CONCLUSIONS AND SIGNIFICANCE

Integrating an advanced health monitoring system on a new engine platform must exhibit both operational and economic benefits. The objectives of decreasing the Maintenance Burden on the Soldier and reducing Operations & Support (O&S) Costs are financial benefits that are relatively easy to account for in a cash flow model. In this paper we presented an approach for calculating some of qualitative benefits. Specifically, we focused on monetizing the benefits associated with increased availability  $A_o$  arising from reducing downtime and fuel savings arising from reducing inefficient engine operations. Key to these value expressions is the concept of a simple life-cycle model that estimates the minimum cost per aircraft flight hour. The key contribution lies in this relatively simple equation that makes minimal assumptions and readily calculated using available design data such as those used to evaluate a new engine procurement option. Since the value must be greater than equal to the cost, we use this to calculate the marginal benefits that can be derived by an increase in  $A_o$ .

The impact of abrupt undetected faults on engine fuel consumption forms the basis for monetizing inefficient operations. Again we assume a typical engine degradation model used to estimate performance-based maintenance intervals.

We illustrated the cost model and its application to evaluate two CBM options available for mechanical health monitoring. While our intent was not to describe the underlying technologies, but illustrate the cost and benefit elements that were used to evaluate the two options with respect to each other.

Cost Elements	Costs (\$M)	Technology		
		MHM - I	MHM-II	Other
	Development cost	med	med	med
	Unit Cost	low	low	low
	Common Unit Cost	shared across integrated suite		
	Sustainment cost	shared across integrated suite		
	Lifetime Investment	med	med	med
	Comm. Lifetime invest	shared across integrated suite		
	Weight Penalty (lbs) & cost penalty (\$)	small low	med low	med low
Army CBM Objectives	Benefits (\$M)			
Decrease Maintenance Burden	Maintainer Burden reduction	small	small	small
Increased Platform Availability & Readiness	Increased Availability (Ao)	small	med	small
	Utility Value	small	med	small
	Mission abort savings	med	med	
Enhanced Safety	Accident avoidance	med	med	
Reduced Operations and Support (O & S) Costs	% DMC change			small
	Cost saving			
	Fuel saving			
	Unscheduled → planned Mx	small	med	med
	Depot part/labor savings	small	small	small
Net Cash Savings (\$M)		med	large	med

Figure 4. Summary table for presenting CBA results

**REFERENCES**

Jaw L.C, *Recent advancements in aircraft engine health management (EHM) technologies and recommendations for the next step*, in Proceedings of the ASME Turbo Expo, vol. 1, pp. 683-695, 2005, GT2005-68625.

Katzomis, R. (1998, Feb). Trends in O&S Cost for the AH-64A. Retrieved from <http://www.fas.org/man/dod-101/sys/ac/docs/rkatzomis/sld004.htm>.

Mylaraswamy, D., Nwadiogbu, E., and Olson, L., *Engine Performance Trending: Practical Insights*, Trans Inst of Measurement and Control, 31(3-4), 341-354 (2009).

Parthasarathy, G., Mylaraswamy, D., Uluyol, O., Kim, K., Vohonout, S., Thompson, B., *Readiness Approach for Propulsion Engine LRUs*, MFPT Annual Conference, 2011.

Schmidt, M. (3<sup>rd</sup> Edition). (2009). *Business Case Essentials: A Guide to Structure and Content*. Solution Matrix Ltd.

Secretary of Defense for Acquisition, Technology, and Logistics. (2010). *Report of the OSD CBM+ Action Group 2010 Summer Study* (37 pages), October 2010, Washington, D.C.

Uluyol O., Kim K., and Hickenbottom C, *A Systematic Approach to Bearing Health Monitoring*. Presented at the American Helicopter Society 66th Annual Forum, Phoenix, Arizona, 2010.

U.S. Army (1996). ADS-51-HDBK, *Aeronautical Design Standard Handbook, Rotorcraft and Aircraft Qualification Handbook*.

U.S. Army. (2012). ADS-79C-HDBK, *Aeronautical Design Standard Handbook for Condition Based Maintenance Systems for US Army Aircraft*.

U.S. Army. (Version 4.4.0). (2007). *Cost Effectiveness Analysis (CEA) Model*. RDECOM Redstone Arsenal, AL.

## BIOGRAPHIES



**Grant A. Gordon** is a Staff Engineer in the Engine Systems group within Honeywell's Advanced Technology division. He has received three degrees in engineering culminating with a Ph.D. in Engineering Science and Mechanics from The Pennsylvania State University in 1995. At Honeywell, he leads programs to develop prognostic and diagnostic tools for condition based maintenance of airframes and engines. He has published more than 50 technical articles and holds ten patents. He is a certified Program Management Professional (PMP), a member of the American Helicopter Society, serves as an elected chairman on the SAE aerospace G-11 standards committee and on various conference committees.



**Chris Hickenbottom** is a senior systems engineer in Honeywell Aerospace's Condition-Based Maintenance (CBM) Center of Excellence. His background is primarily with propulsion systems for Honeywell's Business and General Aviation (BGA) and Defense and Space (D&S) products. Chris has been involved in many development programs, including the AS907/HTF7000 propulsion engine, the Power and Thermal Management System (PTMS) for the Joint Strike Fighter, and the Zing™ system for the TFE731 Remote Monitoring Service. Chris received his BSME from the University of Arizona.



**Dr. Dinkar Mylaraswamy** joined Honeywell in 1997 after completing his PhD from Purdue University. As the CBM Technology fellow he is responsible for identifying and maturing strategic health management technologies that cut across multiple products and services. He has served as the technical lead on various health management programs—across Petrochemical, Building and Aerospace business within Honeywell as well the US Army, NASA, UK-MOD, and Navair programs. He routinely works with academic institutes and serves on industry panels. He has authored over 30 papers and holds 18 patents in the area of fault diagnosis and its applications.



# An Approach to Prognostic Decision Making in the Aerospace Domain

Edward Balaban<sup>1</sup>, Juan J. Alonso<sup>2</sup>

<sup>1</sup> NASA Ames Research Center, Moffett Field, CA, 94035, USA  
*edward.balaban@nasa.gov*

<sup>2</sup> Stanford University, Stanford, CA, 94305, USA  
*jjalonso@stanford.edu*

## ABSTRACT

The field of Prognostic Health Management (PHM) has been undergoing rapid growth in recent years, with development of increasingly sophisticated techniques for diagnosing faults in system components and estimating fault progression trajectories. Research efforts on how to utilize prognostic health information (e.g. for extending the remaining useful life of the system, increasing safety, or maximizing operational effectiveness) are mostly in their early stages, however. The process of using prognostic information to determine a system's actions or its configuration is beginning to be referred to as Prognostic Decision Making (PDM). In this paper we propose a formulation of the PDM problem with the attributes of the aerospace domain in mind, outline some of the key requirements for PDM methods, and explore techniques that can be used as a foundation of PDM development. The problem of satisfying the performance goals set for specific objective functions is discussed next, followed by ideas for possible solutions. The ideas, termed Dynamic Constraint Redesign (DCR), have roots in the fields of Multidisciplinary Design Optimization and Game Theory. Prototype PDM and DCR algorithms are also described and results of their testing are presented.

## 1. INTRODUCTION

As aerospace vehicles become more complex and their missions more demanding, it is becoming increasingly challenging for even the most experienced pilots, controllers, and maintenance personnel to analyze changes in vehicle behavior that can indicate a fault and accurately predict the short- and long-term effects that the fault can produce. For this reason, some of the latest vehicle designs begin to incorporate automated fault diagnostic and prognostic methods

that can assist with these tasks (Janasak & Beshears, 2007; Benedettini, Baines, Lightfoot, & Greenough, 2009; Reveley, Leone, Briggs, & Withrow, 2010; Delgado, Dempsey, & Simon, 2012). The research into how to utilize prognostics-enabled health information in making autonomous or semi-autonomous decisions on system reconfiguration or mission replanning is still in its early stages, however.

There are other fields (e.g., operations research, medicine, financial analysis, and climatology) where computer-assisted Prognostic Decision Making (PDM) can play or already plays a role - even if the terminology used for it is different (see, for instance, (Räisänen & Palmer, 2001), (Wang & Zhu, 2008), or (Kasmiran, Zomaya, Mazari, & Garsia, 2010)). While the fundamentals of PDM methods for these fields are likely to be similar, we believe that there are important reasons to examine how such methods should be developed and used specifically in the context of aerospace.

First, we believe that PDM development needs to be informed by the unique set of aerospace domain characteristics, where the operating environment is often harsh and dynamic, systems are highly complex, and an incorrect decision can lead to loss of life. Conversely, it would be beneficial to inform vehicle design by the needs and capabilities of PDM algorithms. This includes computing requirements, sensor suite selection, component redundancy considerations, operating procedures, and communication architectures. A capable (and appropriately verified and validated) PDM system can expand both design and operating options for an aerospace vehicle in much the same way as a new composite material can do for its structure or a new type of fuel can do for its propulsion system.

We foresee a number of use cases for PDM in aerospace applications, with some possibilities listed below:

- Maintenance and supply chain management
- Safety assurance for manned aircraft and spacecraft

Edward Balaban et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

- Mission effectiveness maximization for unmanned vehicles

In this paper we propose a set of general properties that problems of interest to PDM researchers may have and consider how methods from the fields of mathematical optimization, multidisciplinary design optimization, and game theory can be utilized in the development of PDM systems for aerospace. The discussion will primarily center on certain elements of mission-, vehicle-, and subsystem-level reasoning, however we believe that the longer-term goal of PDM development should be in creating distributed, yet comprehensively interconnected systems that support information flow from the highest, e.g. fleet, levels down to the individual vehicle components - and back. To achieve that goal, four main areas will need to be addressed: **(1) approaches for effective system (problem) decomposition** into subproblems; **(2) decision-making problem formulations** for different types of subproblems; **(3) decision-making methods** appropriate for the subproblem types; **(4) methods for adjusting problem formulations** (such as constraints) in real-time, if necessitated by prognostic predictions in off-nominal situations.

The paper is organized around the following objectives:

- Provide some motivating examples for considering PDM in the context of aerospace engineering (Section 2)
- Identify some of the more challenging problems in aerospace decision-making and outline the requirements such problems can impose on PDM methods (Section 3)
- Provide the definitions used in this work and formulate the problem class of interest from a constrained optimization point of view (Section 5)
- Outline some of the potential approaches to solving the formulated class of problems (Section 6)
- Discuss the type of situations where a problem formulation may need to be adjusted in real-time and suggest some approaches to doing that (Section 8)
- Describe prototype algorithms for generating PDM solutions and adjusting system constraints (Section 7 and Section 9, respectively)
- Demonstrate the algorithms on example scenarios involving a planetary rover (Section 11)

Additionally, Section 4 contains a review of related prior efforts, and Section 10 describes the software/hardware testbed used in the experiments. The paper concludes with a summary of findings and an outline of potential directions for future work.

## 2. MOTIVATING EXAMPLES

Before we describe the problem class of interest for our current work, it may be helpful to consider a few motivating examples. They are chosen to illustrate the use cases listed in

the Introduction. While only three examples are mentioned here, the field of aerospace has certainly no shortage of them.

### 2.1. A surveying UAV

Our first example is an electrically-powered surveying UAV, such as the SWIFT (Denney & Pai, 2012). The SWIFT is currently in development at NASA Ames Research Center. This example is meant to illustrate the first and the third use cases, that is where PDM could be an integral part of maintenance and logistics operations, as well used for contingency management if degradation of one of the components crosses into the fault region during the mission.

#### Description

- The UAV performs surveying missions over a defined area (e.g., earthquake fault zone mapping, pipeline monitoring, or air sampling)
- Maintenance for degrading or damaged components needs to be scheduled and replacement parts need to be ordered. Each of the objectives listed below has an importance value associated with it that can change from mission to mission or even within the same mission (if, for instance, an in-flight fault or failure occur).

#### Objectives

- Maximize the number of measurements or area coverage per mission
- Maximize vehicle availability for missions
- Maximize safety
- Minimize operational costs

#### Constraints

- Airspace restrictions
- Battery capacity
- Component operating limits
- Return to point of launch (desirable)

### 2.2. United Airlines Flight 232

The second use case (safety assurance for manned vehicles) is illustrated with the example of United Airlines Flight 232 from Denver to Chicago in 1989 (NTSB, 1989):

#### Description

- A fan disk in one of the three engines of the DC-10 aircraft failed and disintegrated
- Fan disk shrapnel disabled the presumably redundant hydraulic controls
- The crew resorted to using differential thrust on the remaining two engines to steer the aircraft to an emergency landing

**Objectives**

Minimize injuries and fatalities

**Constraints**

- Component capabilities and safety margins
- Location and configuration of potential emergency landing sites
- Availability of emergency services at the sites

**2.3. Hayabusa (MUSES-C) spacecraft**

The example of JAXA’s Hayabusa spacecraft (Kawaguchi, Uesugi, & Fujiwara, 2003) illustrates the third use case and is interesting for a number of reasons. While it became the first mission to return samples from an asteroid (Itokawa), it was, however, primarily a technology development mission, with engineering goals assigned point values pre-launch (table 1<sup>1</sup>). Due to long communication delays during certain phases of the mission, autonomous operation was utilized extensively. Several problems jeopardized mission objectives, however, and required numerous changes to the mission plan and the configuration of the spacecraft.

Table 1. Pre-launch mission goals for Hayabusa

Pre-launch mission goals	Points
Operation of ion engines	50
Operation of ion engines for more than 1000 hours	100
Earth gravity assist with ion engines	150
Rendezvous with Itokawa using autonomous navigation	200
Scientific observations of Itokawa	250
Touch-down and sample collection	275
Capsule recovered	400
Samples obtained for analysis	500

**Description**

- A large solar flare damaged solar cells en route to the asteroid
- Reduction in electrical power negatively affected the efficiency of the ion engines
- Two reaction wheels (X and Y) failed
- Release of MINERVA mini-probe failed

**Objectives**

Maximize engineering and scientific payoff

**Constraints**

- Component capabilities and safety margins
- Orbital mechanics
- On-board propellant amount

<sup>1</sup>reproduced from <http://www.isas.jaxa.jp/enter/missions/hayabusa/today.shtml>

**3. PROBLEM CLASS OF INTEREST AND REQUIREMENTS**

As discussed in the Introduction, we believe that PDM systems will eventually need to support decomposition of the overall problem into smaller problems on different levels of system abstraction. Some of these smaller problems could potentially be solved with the more traditional decision-making techniques, such model-predictive control or partial-order planning. While investigating the use of such techniques in the context of prognostic decision-making would certainly be worthwhile, in order to narrow down the scope of this work we focus on the class of problems for which decision-making methods may not yet be sufficiently developed. The examples in the previous section (and others like them) allow us to outline the general attributes of the class:

**Attributes of the problem class of interest**

- The system under consideration is complex, consisting of multiple distinct components
- The operating environment is complex and dynamic
- The system may experience degradation processes, due to either external or internal factors, that lead to faults that can be considered significant. Fault magnitudes and secondary effects may evolve over time.
- In case of a fault (or faults), decision on mitigation actions required in a limited amount of time

**Requirements**

The following high-level requirements could then be proposed on the PDM methods for solving such problems:

1. **Should be general and adaptable**  
It may not be possible to define even partial solutions *a priori* for specific combinations of system state, environmental conditions, constraints, and objectives.
2. **Should utilize prognostic information, if available**  
While offering the benefits of an insight into a future system state, incorporation of prognostic capability may also result in a substantial increase in computational complexity. In practice, obtaining prognostic information could require execution of a computationally-expensive simulation for each potential solution.
3. **Shall accommodate uncertainty and inconsistency in input data**  
Input data available in aerospace applications often suffers from noise, drop-outs, uncertainty of accuracy, and other issues.
4. **Should support system decomposition**  
The ability to account for condition, objectives, and constraints of individual subsystems and components can result in increased solution quality. Carrying out decision-making in a distributed fashion can also be beneficial from the performance point of view.

5. **Should not depend on knowing objective function properties**

Objective functions (defined in Section 5) may not be guaranteed to be convex or differentiable, for example, thus 'blackbox' reasoning techniques may need to be utilized.

6. **Shall be time-boundable**

In most cases a valid solution will be required within a prescribed period of time. In some circumstances the system will also be required to be **interruptable**, i.e. capable of supplying a valid solution even if the decision making process is interrupted before the originally specified time interval has elapsed.

7. **Shall support multi-action solution generation**

In addition to being able to generate single-action solutions, such as setting controller gain values, the system needs to be able to generate multi-action solution sequences.

8. **Should support multiple objectives**

This requirement is motivated by scenarios where, for instance, failure risk is to be minimized while mission payoff is to be maximized. Also applicable to cases where the condition of multiple subsystems or components needs to be taken into account.

A subset of these requirements (*High-dimensional*, *Expensive* (computationally), *Blackbox*) is sometimes referred to in the literature as HEB (Shan & Wang, 2009).

#### 4. PRIOR WORK

Before moving on to describing the initial approach we chose to take in developing decision-making methods, we will review some of the prior related efforts. The research efforts described in this section were chosen from several different fields where prognostic-style information is used for system action determination and we believe them to be representative of the current state of the art.

##### 4.1. Prognostics-enhanced control

Pereira *et al* propose a Model Predictive Control (MPC) approach for actuators that distributes control effort among several redundant units (Pereira, Galvao, & Yoneyama, 2010). Redistribution is performed based on prognostic information on their deterioration. A degradation model of the plant is used that represents damage accumulation to be proportional to the exerted control effort  $u$  and its variation  $\Delta u$ . Bogdanov *et al* (Bogdanov, Chiu, Gokdere, & Vian, 2006) investigate coupling of a prognostic lifetime model for servo motors with a family of LQR controllers. External load disturbances on the servo are assumed to be stochastic.

In (D. W. Brown, Georgoulas, & Bole, 2009) Brown *et al* report on prognostics-enhanced fault-tolerant controller that

trades off performance for RUL. The controller is based on MPC principles, with control boundaries for  $t_{RUL}$  corresponding to a particular input  $u_{RUL}$  used as soft cost constraints. The work is extended with error analysis and estimation of uncertainty bounds for long-term RUL predictions in (D. W. Brown & Vachtsevanos, 2011). In (Bole, Tang, Goebel, & Vachtsevanos, 2011) Bole *et al* also study optimal load allocation given prognostic data about fault magnitude growth (including uncertainty bounds on the prediction). The concept of *Value at Risk (VaR)*, coming from the field of finance, is used as the key performance metric. The case study used in the experiments is an unmanned ground vehicle (UGV) that experiences winding insulation degradation in the drive motors due to thermal stress.

##### 4.2. Post-prognostic decision support and condition-based maintenance

Iyer *et al* use the term *post-prognostic decision support* to describe their framework for Pareto set generation and interactive expression of user preferences throughout the process (Iyer, Goebel, & Bonissone, 2006). The approach is illustrated with a logistics planning example, where mission assets need to be allocated based on the estimated state of health of an asset and the projected availability of replacement parts. An exhaustive search technique was used as the optimization method in the experiments, with the intention to replace it with a genetic algorithm in the future.

In (Haddad, Sandborn, & Pecht, 2011b) and (Haddad, Sandborn, & Pecht, 2011a) Haddad *et al* present a prognostics-enabled optimization model for maximizing availability of an offshore wind farm. The model is based on Real Options Analysis (ROA) and stochastic dynamic programming. The concept of ROA also comes from the field of finance and refers to analysis over either real, tangible assets or opportunities for cost avoidance. The method is illustrated with an example where an optimum subset of turbines to be maintained needs to be found, given the information on their degradation, availability requirements, and cost constraints.

##### 4.3. Automated contingency management

The work done by Tang, Edwards, Orchard, and others on Automated Contingency Management (ACM) includes elements of prognostics-enhanced control, but also extends to prognostic mission replanning (Tang *et al.*, 2007; Edwards, Orchard, Tang, Goebel, & Vachtsevanos, 2010; Tang, Hettler, Zhang, & Decastro, 2011). Diagnostic and prognostic algorithms for various component types were developed and integrated into a prototype decision-making framework for an unmanned ground vehicle (UGV). RUL estimates were used either as a constraint or as an additional element in the cost function of the path-planning algorithm. A *Field D\**-style search routine was used for receding horizon planning. Meth-

ods for estimating and managing process uncertainty were also developed.

## 5. DEFINITIONS AND PROBLEM FORMULATION

In this section we provide the definitions of the concepts used in the rest of this work and represent the problem class described in Section 3 in terms of Partially Observable Markov Decision Processes. The definitions generally follow the conventions found in the contemporary prognostic health management, optimization, game-theoretic, and decision-making literature, with some exceptions as noted. In combining notation conventions used in several different fields, some of the terms had to be assigned symbols that may not be typical for them.

### 5.1. System

The term *system* in this set of definitions is used in a similar sense to the term *plant* from control theory. It can refer to a single component or the entire vehicle, depending on the context. The system is modeled as a constrained, factored, discrete-time Partially Observable Markov Decision Process (POMDP). POMDP (or, in some cases, the more traditional Markov Decision Process), is often used to represent decision-making under uncertainty and with incomplete information about the system (Peek, 1998; Malikopoulos, 2007; Bryce & Cushing, 2007; Boularias, 2010; Bole, 2012). We define POMDP as a tuple  $\{S, A, Z, b_0, T, O, R\}$ , with the components explained below:

$S$	A finite set of partially observable states, $S = \{s_1, s_2, \dots, s_{ S }\}$
$A$	A finite set of possible actions, $A = \{a_1, a_2, \dots, a_{ A }\}$
$Z$	A finite set of observations, $Z = \{z_1, z_2, \dots, z_{ Z }\}$
$b_0$	An initial set of beliefs
$T : S \times A \rightarrow P(S)$	A state transition function, for each state and action giving a probability distribution over next states, $T(s, a, s') = p(s' s, a)$
$O : A \times S \rightarrow P(Z)$	An observation probability function (sensor model), $O(z, a, s') = p(z' s, a)$
$R : S \times A \rightarrow \mathfrak{R}$	A reward function

### 5.2. State Variables

Additionally, a vector of **state variables**

$$X = \{x_1, x_2, \dots, x_{|X|}\}$$

is defined, along with a set of constraints on them:

$$C(X) = \{c_1(X) \geq 0, c_2(X) \geq 0, \dots, c_{|C|}(X) \geq 0\}.$$

### 5.3. Decision Variables

A set of **decision variables**  $U$ , the values of which can be controlled, is defined as well:

$$U = \{u_1, u_2, \dots, u_{|U|}\}$$

Each  $u_i \in U, i = 1, 2, \dots, |U|$ , is coupled with a domain  $D_i$ , over which it is defined. The following inequality and equality constraint sets are specified for the decision variables:

$$G(U) = \{g_1(U) \geq 0, g_2(U) \geq 0, \dots, g_{|G|}(U) \geq 0\},$$

$$H(U) = \{h_1(U) = 0, h_2(U) = 0, \dots, h_{|H|}(U) = 0\}.$$

### 5.4. Decision Making

A **policy**  $\pi$  is defined as a function mapping POMDP states to actions,  $\pi : S \rightarrow A$ , with  $\Pi$  defined as the set of all possible policies.

**Decision-making** in the context of this work is defined as the process of determining a policy  $\pi$  and/or the values of decision variables in  $U$ . For policies, a **decision**  $\delta(\pi) = \{a_1, a_2, \dots, a_n\}$  is defined as the solution to the POMDP (corresponding to a policy  $\pi$ ) and is described as an ordered set of actions.

A **feasible** or **satisfactory** policy  $\pi_s$  is defined as a policy for which  $\delta(\pi_s)$  is such that no  $C(X)$  are violated in any of the states achieved.  $\Pi_f$  is the set of all feasible policies.

If, additionally, **objective functions** and an **objective vector** are defined:

$$\vec{f}(\pi) = \{f_1(\pi), f_2(\pi), \dots, f_{|\vec{f}|}(\pi)\},$$

then the **optimal policy** can be defined as:

$$\pi_0 \triangleq \pi_f : \min \vec{f}(\pi_f),$$

where every objective function is reaching its minimum (best) value. Note that a general assumption of **multiple objectives** and, therefore, multiple objective functions is made.

Finding this strictly optimal (often called **ideal** or **utopian**) policy in practice is usually not possible. Therefore the concept of a compromise policy that achieves good results for the entire objective vector, while possibly not minimizing any particular objective function, is utilized. This concept, known as Pareto optimality, is used widely in economics, operations research, and engineering.

A **Pareto optimal policy** is defined as a policy that is not dominated by any other policy in  $\Pi$ . A vector  $\vec{\alpha} = \{\alpha_1, \alpha_2, \dots, \alpha_k\}$  is defined to dominate vector  $\vec{\beta} = \{\beta_1, \beta_2, \dots, \beta_k\}$  if and only if it is partially less than  $\vec{\beta}$ :

$$(\forall i \in [1, 2, \dots, k], \alpha_i \leq \beta_i) \wedge (\exists j \in [1, 2, \dots, k] : \alpha_j < \beta_j).$$

Dominance of  $\vec{\alpha}$  over  $\vec{\beta}$  is conventionally denoted as  $\vec{\alpha} \prec \vec{\beta}$ .

Policy  $\pi^* \in \Pi$  is then Pareto optimal if and only if:

$$(\forall i = 1, 2, \dots, K, \neg \exists \pi' \in \Pi : \pi' \neq \pi^*, f_i(\pi') \leq f_i(\pi^*)) \\ \wedge (\exists j = 1, 2, \dots, K : f_j(\pi) < f_j(\pi^*)).$$

$\pi^*$  is rarely unique, and, therefore, a **Pareto set** (also known as **Pareto front** (or **Pareto frontier**)) is defined as:

$$\Pi^* \triangleq \{\pi \in \Pi | \neg \exists \pi' \in \Pi, \pi' \prec \pi\}.$$

A representation of a Pareto front for two objective functions is provided on Figure 1. Note, in particular, that a Pareto front should not be assumed to be continuous or convex.

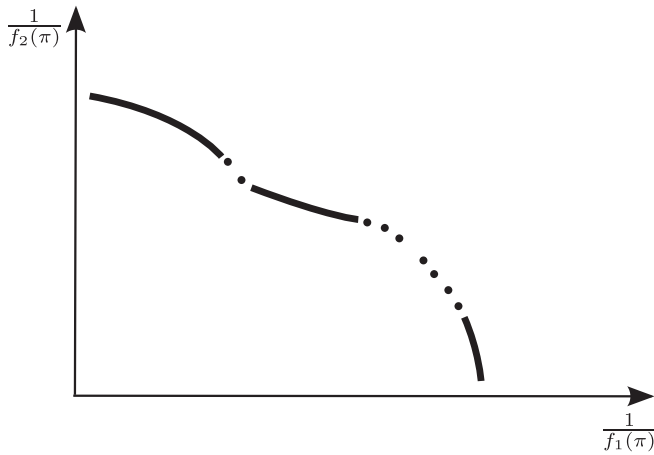


Figure 1. Pareto front

### 5.5. System Degradation and State of Health

*Degradation* is defined as the process of reduction in system performance through time with respect to some criterion (Figure 2). Degradation can be reversible (e.g. through maintenance or self-healing) or irreversible. State of Health (SOH) is a generalized and normalized way of representing degradation, usually defined in the  $[0, 1]$  domain ( $SOH = 1$  corresponds to full health and  $SOH = 0$  represents an inoperable system).  $\eta$  is used to denote the SOH ( $h$  is used in some of the references listed, but is reserved for the decision variables equality constraints in this work).  $\eta$  is uniformly discretized and included as a component of the state vector.

#### Fault

$C_{fault}(X) \in C(X)$  is a subset of the state constraints selected to indicate a significant deviation from nominal behavior, i.e. a fault. A fault occurs when any of the constraints in  $C_{fault}(X)$  is violated. We expect fault constraints to be defined on SOH in most cases, however this definition allows for constraints on other state variables to be used to indicate a fault (such as energy depletion).

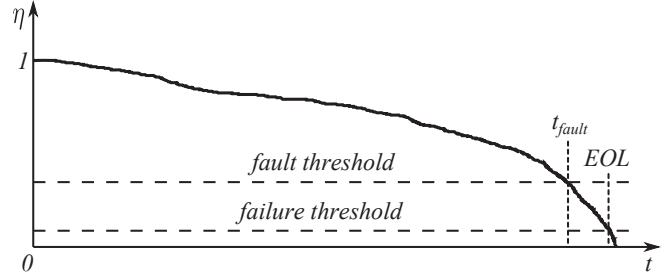


Figure 2. Degradation progression

#### Failure

Similarly, a  $C_{failure}(X) \in C(X)$  subset is defined to indicate deviations from the nominal behavior that render the system functionally unusable.

### 5.6. Prognostics

In this work prognostics is defined as information on projected change in plant behavior through time, e.g. due to wear or degradation (Figure 2). In contrast, a commonly used definition states prediction of the Remaining Useful Life (RUL) and End of Life (EOL) as the goal of prognostics (Daigle & Goebel, 2010; Saxena et al., 2008). We believe that the latter definition may prove to be less convenient for the purposes of PDM, as obtaining intermediate degradation predictions could be important. Decisions on how to minimize degradation could then be made based upon such predictions. For the modeling approach chosen, incorporating prognostic information into the decision process amounts to populating the POMDP with state and transition information.

The following assumptions are made for the above definition:

- A prognostic estimate is defined for a specific instance in time, given the information up to that moment
- Prognosis depends on information regarding the future operating conditions
- Uncertainty in system modeling, outputs, observations, and current/future operating conditions is admissible.

### 6. SELECTING A POLICY GENERATION APPROACH

Having defined the requirements on PDM methods for the problem class of interest and described our modeling approach, we now turn to considering the suitable policy generation techniques. Such techniques are generally classified into *satisficing* or *optimizing* (Simon, 1956), although alternative taxonomies exist as well. The goal of the optimizing techniques is to find solutions on the Pareto frontier or as close to it as possible. The latter only attempt to find feasible solutions. Satisficing techniques are used extensively in many types of applications and often have the advantage of being computationally inexpensive. They also generally lend themselves well to validation and verification.

In this work, however, we chose to formulate the decision-making problem from the optimization point of view - primarily because we believe that this will allow us to take greater advantage of prognostic information. In the rest of the section we comment only briefly on the major types of optimization methods with respect to the requirements in Section 3. As we do not aim to provide a comprehensive survey of modern optimization techniques, interested readers can refer to (Das & Chakrabarti, 2005), (Shan & Wang, 2009), or (Rao, 2009), to list a few.

**Exhaustive search** (or **brute-force methods**) are generally straightforward to implement and are capable of generating exact Pareto sets. Scalability is the main issue with this type of methods, as they quickly become computationally intractable. They can, however, be useful for verifying performance of other optimization methods on simple problems.

**Gradient Descent, Hill Climbing** and similar local search methods are not guaranteed to find global optima. Gradient Descent methods also generally require objective functions to be defined and differentiable over the entire search space. **Linear Programming, Constraint Programming, Newton, and Quasi-Newton** methods require knowledge of objective function properties as well.

**Dynamic Programming (DP)** methods are widely used for policy generation. The main downsides of traditional DP formulations are that for multi-objective problems a single composite objective function needs to be constructed, i.e. a Pareto set is not produced, and that system decomposition can be difficult to accomplish. Some DP-based methods have been developed, however, that attempt to circumvent both of these issues (see (Hussein & Abo-Sinna, 1993; Driessen & Kwok, 1998; Liao, 2002)). Additionally, with factored state spaces being exponential in size with the number of state variables, exact DP methods become unsuitable for large-size problems. In certain applications, approximate DP methods have been used (Kveton, Hauskrecht, & Guestrin, 2006).

**Stochastic methods** (such as Simulated Annealing, Quantum Annealing, Metropolis-Hastings, Cross-Entropy, or Probability Collectives) generally satisfy the requirements we proposed in Section 3. None of them guarantee optimality; they do, on the other hand, possess the *anytime* property (can be interrupted at any time and still return a valid result), can be used with *blackbox* objective functions, and can accommodate system decomposition.

**Genetic algorithms** (often classified together with stochastic methods) also satisfy the proposed requirements. In such algorithms a prototype (candidate) solution is described as an individual member of a population. Biologically-inspired operators (selection, reproduction, mutation, and others) are used, guided by fitness functions. Genetic algorithms produce

a Pareto front approximation in each iteration and, therefore, are also *anytime*.

For this phase of the work we, ultimately, chose to develop a policy generation method based on Probability Collectives. In the future, we also plan investigate policy generation via genetic algorithms. A method based on Simulated Annealing (SA) was used in the prototype constraint redesign framework (Section 9).

## 7. POLICY GENERATION ALGORITHM DEVELOPMENT

The current policy optimization algorithm is referred to as Probabilistic Policy Generator (PPG). With its roots in the work on *Probability Collectives (PC)* (Wolpert, Strauss, & Rajnarayan, 2006), it belongs to the class of *blackbox* optimization methods. Such methods have the goal of finding a value  $x \in X$  that minimizes an associated value  $F(x)$ .  $X$  is an optimization space (not to be mistaken for the  $X$  used to denote POMDP state vectors in other parts of this work) and  $F(x)$  could be an objective or a utility function. The following process is repeated iteratively: (1) an  $x$  is chosen from  $X$ ; (2) statistical information about  $F(x)$  is updated; (3) the next value of  $x$  is chosen using the  $(x, F(x))$  pairs found up to that point.

The main difference between the conventional *blackbox* approaches and PC is that while the former operate directly on the values of  $x$  (by constructing a map  $M$  from a subset of  $\{(x, F(x))\}$  to the next sample  $x$ ), the latter works with probability distributions over  $x$ . That is done by specifying a map  $m$  from a subset of  $\{(x, F(x))\}$  to the next distribution over  $X$ ,  $P(X)$ . That distribution is then sampled to select the next value of  $x$ . The goal of conventional *blackbox* approaches is to design  $M$  in such a way as to increase the likelihood of finding values of  $x$  corresponding to the small values of  $F(x)$ . In the PC case, the goal for designing  $m$  is to generate  $P(X)$  peaked around the small values of  $F(x)$ . This can be more formally described, for example, in terms of the expected value:

$$\begin{aligned} \text{find } \min_P \int F(x)p(x)dx, \text{ s.t.} \\ x \in X, \int p(x)dx = 1, p(x) \geq 0 \forall x, \end{aligned}$$

with the integrals are replaced by sums for discrete distributions.

There are a number of advantages to working with distributions over  $X$  rather than working with  $X$  directly. One is that the same algorithm could, in most cases, be used for different types of space  $X$  without significant modifications. Another is that  $P$  generated by a PC-based algorithm will be peaked in some dimensions, while being broad in others, thus supplying sensitivity information on the importance of getting better estimates for the values of those dimensions. A PC-based algo-

rithm can also be used to combine and, ideally, improve upon solutions produced by other optimization algorithms. To do that,  $P$  is initialized to a set of broad peaks, each centered on a solution generated by the other algorithm(s). As  $P$  is updated, the shapes of the peaks are defined further and some of them become merged, producing combined solutions. Finally, the approach can be extended to multi-component vectors  $\vec{x}$  in a relatively straightforward fashion.

The earlier versions of the PPG algorithm were described in (Balaban et al., 2011; Narasimhan et al., 2012). It uses 'look-ahead' sampling to aggregate information about policy options, gradually increasing the probability of choosing the more optimal solutions. Its input parameters are the following:

$A$	valid actions set
$\vec{f}(\pi)$	objective function vector
$\vec{v}$	objective preference vector
$G_t$	inequality constraint set
$H_t$	equality constraint set
$l$	maximum policy length
$N_1$	number of utility function calls allocated to the first phase of the algorithm
$N_2$	number of utility function calls allocated to the second phase of the algorithm
$M$	number of stages

Execution time is controlled by specifying  $l$ ,  $N_1$ ,  $N_2$ , and  $M$  (further explained below). The algorithm (see Algorithm 1) operates in the following manner:

#### Initialization (lines 2-5)

A set of partial policies,  $\Pi'$ , is initialized with a single member,  $\pi'_0$ . For simplicity, a partial policy  $\pi'$  is defined as the set actions mapped to the first several states achieved for a decision  $\delta$ . For instance,  $\{a_1, a_2\}$  is a partial policy corresponding to the decision  $\{a_1, a_2, a_3, a_4, a_5\}$ . The probability of  $\pi'_0$  achieving maximum utility ( $p(\pi'_0)$ ) is set to 1. Finally, first phase utility function call quotas are allocated per stage (for a total of  $N_1$ ), with increasing stage numbers corresponding to progressively longer policy roots. The allocation is currently done using a cubic function, with the earlier stages receiving a greater proportion of the total number.

#### Partial policy extension (lines 7-15)

The first phase of the algorithm is executed for  $M$  number of stages. In each iteration the partial policies in  $\Pi'$ , generated during the preceding stages, are extended and the probability of them resulting in an optimal solution is estimated. In order to extend the partial policies, sets of feasible follow-on actions are determined first. In the example problem described in Section 11, the rover should visit each of its target locations once at the most. Thus, if a maximum of five locations

maximum is to be visited, partial policy  $\pi' = \{a_1, a_2\}$  (move to node 1, then to node 2) has  $A_{\pi'} = \{a_3, a_4, a_5\}$  as the set of possible follow-on actions. The valid one-action extensions are then  $\{a_1, a_2, a_3\}$ ,  $\{a_1, a_2, a_4\}$ , and  $\{a_1, a_2, a_5\}$ . These *offspring* partial policies replace the *parent* partial policy ( $\pi'$ ) in  $\Pi'$  and split its probability value evenly.

#### Partial policy probability estimation (lines 17-22)

The probability of each partial policy in updated  $\Pi'$  achieving maximum utility is estimated next. To achieve that,  $\Pi'$  is sampled randomly according to the prior distribution. Each sample  $\pi'$  is used to obtain a decision of the maximum length  $l$ , with valid completion actions selected from  $A_{\pi'}$ . The policy  $\pi_i$  corresponding to the sample decision is then evaluated with respect to the objective function vector  $\vec{f}$  and the constraint set  $C(X)$ . Note that in order to satisfy the constraints, the extended decision may be truncated short of the maximum length. For instance, if  $\delta = \{a_1, a_2, a_3, a_4, a_5\}$  does not satisfy one or more of system constraints, while  $\delta = \{a_1, a_2, a_3, a_4\}$  does, then the latter is picked. The utility value  $u(\pi)$  is computed (currently by using the preference vector  $\vec{v}$ ) and the posterior probability of  $\pi'$  is adjusted after the sampling process is complete. A Normalized Root Mean Squared Error (NRMSE) metric is used to aggregate information on how well  $\pi'$  is performing relative to the maximum utility value seen so far:

$$\epsilon_{\pi'} = \sqrt{\frac{\sum_{i=1}^n (u_{max} - u(\pi))^2}{n(u_{max} - u_{min})^2}},$$

where  $n$  is the number of sample decisions constructed for  $\pi'$ , and  $u_{min}$  and  $u_{max}$  are the minimum and the maximum values of the utility function observed so far, respectively. The metric is the same as a normalized  $L_p$  metric (Coello, Lamont, & Veldhuizen, 2007), with  $p = 2$ .

#### Monte Carlo simulation on $\Pi'$ (lines 27-32)

Once the probability distribution  $P(\Pi')$  is shaped, a Monte Carlo simulation is run for  $N_2$  sample policies. Policy roots are picked according to the distribution, extended to the maximum length satisfying  $C(X)$  and evaluated with respect to  $\vec{f}$ .

#### Solution set filtering (lines 36-38)

Finally, the solution set  $\Pi^*$  is reduced using a variant of the bounded objective method and according to the priority vector  $\vec{v}$ . The objective functions in  $\vec{f}(\pi)$  are sorted in descending order, based on the values in  $\vec{v}$ ,  $|v| = K$ .  $\Pi^*$  is then reduced to  $\Pi_{f_1}^*$ , where the highest-ranked objective is maximized.  $\Pi_{f_1}^*$  is subsequently reduced to  $\Pi_{f_2}^*$  and so on, until either  $|\Pi_{f_k}^*| = 1$  ( $k = 1, 2, \dots, K$ ) or  $k = K$ .



**Algorithm 1** PPG

```

1: procedure PPG( $A, \vec{f}(\pi), \vec{v}, l, N_1, N_2, M$ )
2:    $\pi'_0 \leftarrow \{a_0\}$   $\triangleright$  null action to assume the initial state
3:    $\Pi' \leftarrow \{\pi_0\}$   $\triangleright$  set of all policy roots
4:    $p(\pi'_0) = 1$   $\triangleright$  assign initial probability
5:    $N_s \leftarrow \text{allocateUtilityFunctionCalls}(N_1)$ 
6:   for stage  $\leftarrow 1, M$  do
7:     for all  $\pi'_i$  in  $\Pi'$  do
8:        $A_{\pi'_i} \leftarrow \text{getValidActions}(\pi'_i)$ 
9:        $\triangleright$  generate all possible one-action extensions
10:       $\Pi'_{\pi'_i} \leftarrow \text{extendPolicyRoot}(\pi'_i, A_{\pi'_i})$ 
11:       $\Pi'_{new} \leftarrow \{\Pi'_{new}, \Pi'_{\pi'_i}\}$ 
12:      for all  $\pi'_j$  in  $\Pi'_{new}$  do
13:         $p(\pi'_j) \leftarrow p(\pi'_i) / |\Pi'_{new}|$ 
14:      end for
15:    end for
16:     $\triangleright$  update  $P(\Pi')$ 
17:    for  $i \leftarrow 1, N_s(\text{stage})$  do
18:       $\pi'_s \leftarrow \text{getRandomSample}(\Pi', P(\Pi'))$ 
19:       $\pi_s \leftarrow \text{extendPolicy}(\pi'_s, l)$ 
20:       $\vec{f}(\pi_s) \leftarrow \text{evaluatePolicy}(\pi_s, \vec{f}(\pi), H, G)$ 
21:       $u_s \leftarrow \text{calculateUtility}(\vec{f}(\pi_s), \vec{v})$ 
22:       $p(\pi'_s) \leftarrow \text{updateRootProbability}(\pi'_s, u_s)$ 
23:    end for
24:  end for
25:   $\Pi' \leftarrow \{\Pi'_{new}\}$ 
26:   $\Pi_{mc} \leftarrow \emptyset$ 
27:   $\triangleright$  Monte Carlo simulation on  $\Pi'$ 
28:  for  $i \leftarrow 1, N_2$  do
29:     $\pi'_{mc} \leftarrow \text{getRandomSample}(\Pi', P(\Pi'))$ 
30:     $\pi_{mc} \leftarrow \text{extendPolicy}(\pi'_{mc}, l)$ 
31:     $\vec{f}(\pi_s) \leftarrow \text{evaluatePolicy}(\pi_s, \vec{f}(\pi), H, G)$ 
32:     $\Pi_{mc} \leftarrow \{\Pi_{mc}, \pi_s\}$ 
33:  end for
34:   $\Pi^* \leftarrow \Pi_{mc}$ 
35:   $\triangleright$  Filter policy set
36:   $\vec{f}(\pi)_{sorted} \leftarrow \text{sortDescending}(\vec{f}(\pi), \vec{v})$ 
37:   $n=1$ 
38:  while ( $|\Pi^*| \geq 1$ ) & ( $k < K$ ) do
39:    for all  $\pi$  in  $\Pi^*$  do
40:       $\Pi^* \leftarrow \{\text{all } \pi \text{ in } \Pi^* | f_k(\pi) \text{ is max}\}$ 
41:    end for
42:  end while
43: end procedure

```

**8. DYNAMIC CONSTRAINT REDESIGN**

The preceding sections of the paper concentrated on the incorporation of prognostic information into the decision making process and the selection of appropriate policy optimization methods. The outcome of a multi-objective optimization is a Pareto set of policies  $\Pi^*$ . There are three "goldilocks" possibilities with respect to the size of  $\Pi^*$ :

1. The size is acceptable, i.e.  $1 \leq |\Pi^*| \leq N$ , where  $N$  is the maximum number of candidate policies that can be practically down-selected by inspection, using heuristic methods, or by some other means.

2. The size is too large, i.e.  $|\Pi^*| \geq N$ . In this case the set can be reduced either through interaction with a human expert (as described earlier in (Iyer et al., 2006)) or through an autonomous process that adds/tightens constraints in  $C(X)$  and re-runs the optimization until a  $\Pi^*$  of a desired size is achieved.
3. No feasible solutions exist, i.e.  $|\Pi^*| = 0$ . In this case the original constraints in  $C(X)$  may need to be relaxed or eliminated.

The second case is an interesting research area that we hope to explore further in the future. In the current work, however, we focus on the third case. In addition to the absence of feasible solutions, however, there could be another reason why  $\Pi^*$  may not be suitable - which is the subject of the next section.

**8.1. Performance goals satisfaction**

Consider the case where, in addition to constraints in  $C(X)$ , constraints (or, rather, performance goals) were also defined for some or all of the elements of  $\vec{f}$ , as is done in Goal Programming (Tamiz, Jones, & Romero, 1998), for instance:

$$\Gamma(\vec{f}) = \{\gamma_1(\vec{f}) \geq 0, \gamma_2(\vec{f}) \geq 0, \dots, \gamma_{|\vec{f}|}(\vec{f}) \geq 0\}.$$

An example of a Pareto set not satisfying some of the performance goals in  $\Gamma(\vec{f})$  is illustrated on Figure 3.

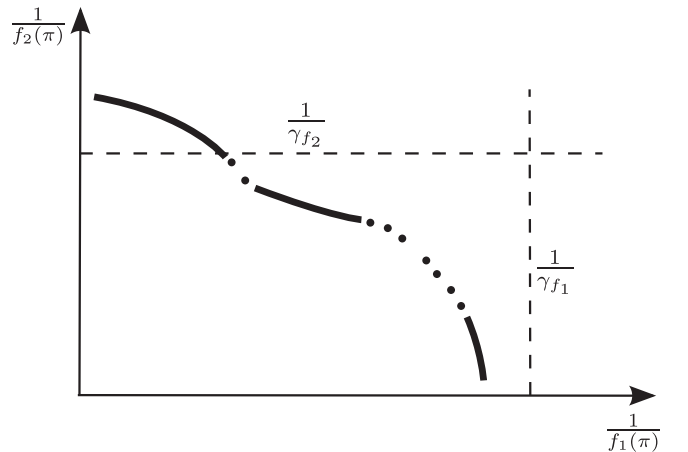


Figure 3. An example of a Pareto set not satisfying a performance goal ( $\gamma_{f_1}$ ).

If no acceptable solutions are found during the optimization process and if the performance goals are considered to be of high enough importance, then constraints in  $C(X)$  may need to be changed or eliminated.

For convenience, in this paper we refer to the process of modifying system constraints as **Dynamic Constraint Redesign (DCR)**. In the context of an aerospace vehicle, DCR could mean knowingly damaging a component or a subsystem beyond repair if that means saving the overall vehicle. Only system constraints will be considered for the purpose of this dis-

cussion, however one can also envision eliminating or relaxing external constraints, such as airspace restrictions or flight separation distances.

DCR can also be thought of as redesigning the vehicle "on the fly", by changing its performance characteristics to the outside of the known envelope - while simultaneously searching for a Pareto optimal policy to best utilize the modifications in the current mission. Some of the same issues arise as during the original design, e.g. subsystem compatibility assurance, choice of design variables, and design variable sensitivity analysis. In the last few decades the field of Multidisciplinary Design Optimization (MDO) has been developed to address these and other issues during the initial design of complex systems. We believe that some of the techniques from MDO community could be beneficial in development of DCR as well.

## 8.2. Multidisciplinary Design Optimization (MDO)

In this section we briefly review some of the most popular MDO approaches and comment on their applicability to DCR (far more extensive descriptions of contemporary MDO approaches and methods can be found, for instance, in (Agte et al., 2009; Shan & Wang, 2009; Honda, Ciucci, Lewis, & Yang, 2010)). First, however, it would be helpful to note some key differences between MDO and DCR problems:

- Robust validation and verification of a candidate point design using independent methods may not be possible for PDM/DCR, unlike in MDO;
- Related to the preceding point, the risk associated with each potential DCR solution needs to be quantified;
- Achieving real-time performance will, generally, be of far greater importance to PDM/DCR than to MDO.

One of the ways to classify modern MDO algorithms is into these two broad categories: *All-At-Once (AAO)* and *decomposition*. **All-at-Once** algorithms, also referred to as All-In-One (AIO) or single-level, aim to achieve design decisions through a single global optimization process (Cramer, Dennis, Frank, Shubin, & Lewis, 1993; N. Brown, 2004). While such formulations have some attractive qualities (for instance, each iteration produces a discipline-feasible solution and sensitivity analysis on design variables is usually easy to perform), they also have significant downsides. A designer using AAO methods is likely to run into scalability issues when applying them to large, complex systems. Also, by aggregating knowledge from the subsystems into a single optimizer, some of the discipline-specific knowledge may be lost. Finally, AAO approaches tend to limit the use of well-proven analysis and optimization techniques at the discipline level.

**Decomposition** methods break down a design optimization problem into multiple subproblems, usually along the boundaries of disciplines, subsystems, or individual components

(Cramer et al., 1993). Some of the better known methods are bi-level, such as **Collaborative Optimization (CO)**, **Concurrent Subspace Optimization (CSSO)**, or **Bi-Level Integrated System Synthesis (BLISS)**, and multi-level, such as **Analytical Target Cascading (ATC)**.

**CO** (Braun, Gage, Kroo, & Sobiesky, 1996; Roth & Kroo, 2008; Roth, 2008) uses target values of the design and state variables, specified at the system level, to guide individual discipline optimizations. Communication between disciplines in most CO implementations is limited, which simplifies implementation, but can also result in slow convergence.

The **CSSO** method (J. E. Renaud & Gabriele, 1993; Sobieszczanski-Sobieski, Agte, & Sandusky, 1998; Sellar, Batill, & Renaud, 1996; G. Renaud & Shi, 2002) performs discipline-specific optimization using local objective functions, variables, and constraints, while approximating effects on system performance using Global Sensitivity Equations, Response Surfaces, or other types of system models. Similarly, system-level models of disciplines are used in order to approximate their behavior. As performance information is accumulated throughout the process, the models can be updated correspondingly.

In **BLISS** (Sobieszczanski-Sobieski et al., 1998; Sobieszczanski-Sobieski, Emiley, Agte, & Sandusky, 2000) each iteration of the procedure improves the design both on the local (discipline) and system levels. First, a concurrent local optimization is performed using the discipline design variables and keeping the system-level variables constant. Then, a system-level optimization on shared variables is done. Total derivatives are communicated among the disciplines to help predict the effects of local design choices on the other disciplines.

**Analytical Target Cascading (ATC)** (Kim, 2001; Kim, Michelena, Papalambros, & Jiang, 2003; Allison, Kokkolaras, Zawislak, & Papalambros, 2005), is primarily intended for problem decomposition by subsystems and components, rather than disciplines. ATC approach is flexible and multi-level, allowing complex system architectures to be represented. Other formal MDO methods can potentially be integrated within an ATC framework (Agte et al., 2009).

Methods founded on the principle of **Lagrangian Duality (LD)** may also be of interest for certain elements of PDM/DCR. Classical LD methods are generally applied to convex problems and accommodate decomposition into smaller sub-problems. In order to handle non-convex problems, Augmented Lagrangian Duality (ALD) theory has been developed (Hestenes, 1969). ALD algorithms, however, lose the decomposition capability. In recent years, several research efforts combined LD and ALD approaches to attain both the 'convexification' properties of ALD and the decomposition properties of traditional LD (Blouin, Lassiter,

Wiecek, & Fadel, 2005; Tosserams, Etman, Papalambros, & Rooda, 2005).

Finally, MDO methods that have evolved from the field of **Game Theory** offer some promising alternatives for design decomposition architectures. The idea of using game formulations in design problems goes back to the work of Vincent (Vincent, 1983) and Rao and Freiheit (Rao & Freiheit, 1991). Some of the further developments are described in (Lewis & Mistree, 1997), (Marston, 2000), and (Clarich & Pediroda, 2004). Games of different forms have been studied for use in MDO applications, at least to some extent: cooperative (Pareto), approximately cooperative, non-cooperative (Nash), coalition, and leader/follower. While intuitively a cooperative (Pareto) form game would appear to be the natural choice when setting up an MDO or a PDM/DCR problem, the other forms have their place as well. For instance, the leader/follower (also known as Stackelberg or extensive) form can be used to set up a sequential analysis problem. The non-cooperative (Nash) form could be used in situations when the established communication protocols between subsystems prove to be insufficient for a particular situation or are affected by a system fault. The coalition form can be used in organizing system analysis by discipline.

For the first DCR prototype we chose to implement a cooperative game-theoretic protocol (described in the next section), with alternative formulations to be implemented and compared in future work. Similarly to BLISS, the implemented algorithm passes the derivatives of local objective functions with respect to shared variables. This is done in order to inform subsystems of the effects their choices may have on the other subsystems.

## 9. DCR ALGORITHM DEVELOPMENT

In the prototyped game-theoretic DCR algorithm the players (subsystems) cooperate in exploring the, potentially, very large option space by taking turns in conducting the search and, when necessary, relaxing some of their constraints. The current formulation of the algorithm tests the concept for two subsystems, with extension to larger numbers of subsystems planned for subsequent work. One constraint per subsystem is currently chosen as the target for redesign ( $c_1$  and  $c_2$ ).

The process (illustrated on Figure 4) starts with one player randomly picked to go first (let us assume that it is Subsystem 1). *Subsystem 1* conducts an iteration of the search, finding its best guess at the optimal policy  $\pi^*$ . The policy needs to satisfy constraints in both  $C$  and  $\Gamma$ . Also, a maximum of  $N$  utility function calls is allowed per iteration. If no acceptable policy is found, the target constraint  $c_1$  is adjusted (becoming, for instance,  $62 - T_{max} > 0$ ). Another search iteration is performed and suitability of solutions is evaluated. The process repeats until a maximum number of search attempts,  $N_{max}$ , is reached or a non-empty set  $\Pi_1^*$  is found.  $\Pi_1^*$ , empty or oth-

erwise, is then sent over to *Subsystem 2*, along with the necessary gradient information on objective function performance (in a non-cooperative formulation only  $\Pi_1^*$ , also known as the Best Reply Correspondence or BRC, would be transmitted). Note that gradient estimates are shared not only for policies in  $\Pi_1^*$ , but also for other policies considered during the search. If there is at least one policy  $\pi^* \in \Pi_1^*$  that is also suitable from the point of view of *Subsystem 2*, then the process is stopped. Otherwise *Subsystem 2* conducts its own search iteration, adjusting  $c_2$  as needed, and hands over control of the search to *Subsystem 1* after either  $N_{max}$  search iterations are completed or a non-empty  $\Pi_2^*$  is found.  $\Pi_2^*$  and the objective function gradients are then transmitted back to *Subsystem 1*. The process continues until a  $\pi^*$  satisfying both subsystems is found.

It is important to take a look at how objective functions for each of the players are designed. In non-cooperative game formulations (and some of the traditional MDO approaches) discipline/subsystem objective functions primarily focus on the needs of that particular discipline or subsystem. In order to help expedite convergence, in this cooperative formulation composite objective functions that take into account the effect a candidate solution may have on global objectives and on the other players are used. The functions take on the following form:

$$\begin{aligned} f_1(\pi) &= w_{1,1}f_g(\pi) + w_{1,2}|\nabla f_{2,l}|\pi + w_{1,3}f_{1,l}(\pi), \\ f_2(\pi) &= w_{2,1}f_g(\pi) + w_{2,2}|\nabla f_{1,l}|\pi + w_{2,3}f_{2,l}(\pi), \end{aligned}$$

where  $f_g$  is the global objective function (currently a single one),  $f_{i,l}$  is the objective function local to the subsystem,  $i$  is the subsystem number, and  $w_{i,j}$  are the weights used to specify the degree of influence of each of the components of  $f_i$ .

Another important feature of the algorithm is that with each iteration the size of the constraint-adjusting step is increased, thus encouraging the players to come up with a solution suitable from the other subsystems' (and global) points of view as quickly as possible.

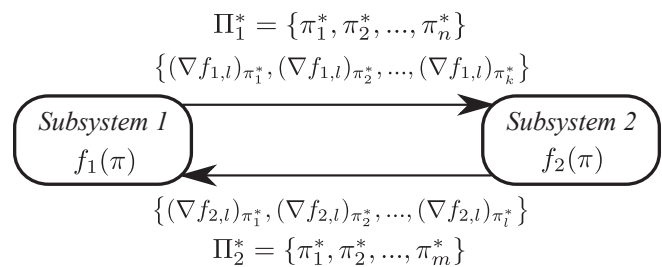


Figure 4. Two-subsystem cooperative game formulation

A variant of Simulated Annealing (Bertsimas & Tsitsiklis, 1993), or SA, is currently utilized for searches of the option space by the subsystems. In this particular case SA was chosen to take advantage of the gradient information exchanged

by the subsystems, while also avoiding getting 'stuck' at the local minima. The algorithm accomplishes the latter by performing randomized jumps to other promising locations of the search space. The probability of continuing with the local search vs. performing a jump is influenced by an *annealing schedule*  $T(t)$ , and is

$$p[x(t+1) = x_j | x(t) = x_i] = w_{ij} \exp \left[ -\frac{1}{T(t)} \max\{0, f(x_j) - f(x_i)\} \right],$$

where

- $X$  A finite search space (again, not to be mistaken for the POMDP state vector).
- $f$  A real-valued objective function  $f$  defined on  $X$ .  $X^* \subset X$  is the set of the global minima of  $f$ .
- $X_i$  The neighbor set of  $x_i$ ,  $X_i \subset \{X - x_i\}$ ,  $x_i \in X$ .
- $w_{ij}$  A probabilistic weight for transition from  $x_i$  to  $x_j$ ,  $x_j \in X_i$ , s.t.  $\sum_{x_j \in X_i} w_{ij} = 1$ , with  $x_j \in X_i \iff x_i \in X_j$  implied.
- $T$  The *annealing schedule*.  $T : N \rightarrow (0, \infty)$  is a non-increasing function and  $N$  is a set of positive integers,  $T(t)$  is the *temperature* at time  $t$ .

The above assumes that  $x_i \neq x_j$ ,  $x_j \in X_i$ . If  $x_i = x_j$  or  $x_j \notin X_i$ , then  $p[x(t+1) = x_j | x(t) = x_i] = 0$ .

## 10. TEST PLATFORM

The testbed being used in the current validation experiments is the K11 planetary rover prototype and its associated software simulator (Balaban et al., 2011). Another testbed targeted for future experiments is the Edge 540 UAV located at NASA Langley (Hogge, Quach, Vazquez, & Hill, 2011). While the algorithmic infrastructure is developed to accommodate the UAV, that part of the work is, otherwise, in its early stages.

### 10.1. K11 overview

The K11 is a large four-wheeled rover platform (approximately 1.4 m long by 1.1 m wide by 0.63 m tall, weighing roughly 150 kg). Each wheel is driven by an independent 250 W graphite-brush motor, connected through a bearing and gearhead system, with each motor controlled by a single-axis digital motion controller. Four 14.8 V 3.3 Ah lithium-ion batteries, connected in series, power the vehicle. The on-board computer runs control and reasoning algorithms, as well as coordinates data acquisition. Measurements available on-board are shown in Table 2 and on Figure 5.

In the table  $F, B, L, R$  refer to front, back, left, and right, respectively. Altitude  $h$  is determined using  $\lambda, \phi$  and a terrain map  $\mathcal{M}$ .

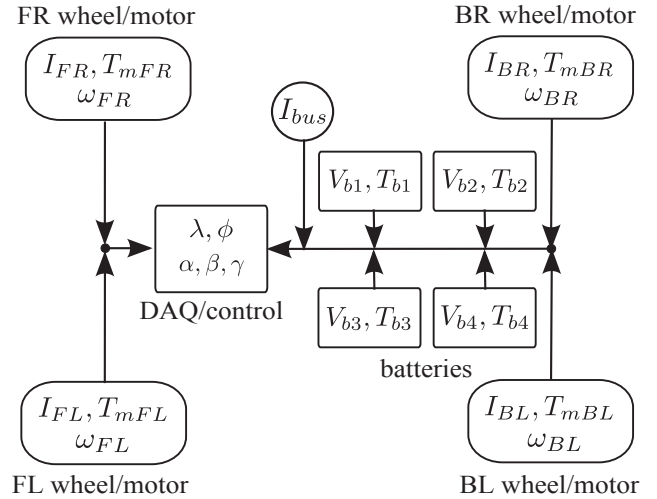


Figure 5. K11 data flow

Table 2. K11 data

measurement	symbol
absolute position (longitude, latitude)	$\lambda, \phi$
wheel angular velocity	$\omega_{FL}, \omega_{FR}, \omega_{BL}, \omega_{BR}$
attitude (yaw, pitch, roll)	$\alpha, \beta, \gamma$
battery temperature	$T_{b1}, T_{b2}, T_{b3}, T_{b4}$
battery voltage	$V_{b1}, V_{b2}, V_{b3}, V_{b4}$
motor temperature	$T_{mFL}, T_{mFR}, T_{mBL}, T_{mBR}$
motor current	$I_{FL}, I_{FR}, I_{BL}, I_{BR}$
power bus current	$I_{bus}$

The software simulator reproduces both nominal and off-nominal behavior of the hardware testbed. The simulator has a dual purpose: (a) to aid in the development of PDM algorithms as a virtual testbed and (b) to provide  $\vec{f}$  estimates during the decision-making process.

### 10.2. Fault Modes

Table 3 describes the K11 fault modes, implemented either in hardware, simulation, or both. Some of the fault modes, such as sensor faults, are injected primarily for testing diagnostic functionality (i.e. such faults have brief fault-to-failure times), while the others exhibit a more continuous fault progression behavior and are used for validation of prognostic algorithms.

### 10.3. Diagnostic Functionality

Two diagnostic algorithms are currently in use with the K11 testbed. The first one, QED (Qualitative Event-based Diagnosis), is described in (Daigle & Roychoudhury, 2010). It utilizes a qualitative diagnosis methodology that isolates faults based on the transients they cause in the system behavior, manifesting as deviations in residual values (Daigle &

Table 3. K11 fault modes.

fault model	subsystem
battery capacity degradation	Power
parasitic electric load	Power
motor failure	Propulsion
increased motor friction	Propulsion
sensor bias/drift/failure	Sensors

Roychoudhury, 2010). The second, Hybrid Diagnosis Engine (HyDE) is a diagnosis algorithm that uses candidate generation and consistency checking to diagnose discrete faults in stochastic hybrid systems (Narasimhan & Brownston, 2007). 'Hybrid' in this case refers to combined discrete and continuous models used by the algorithm to analyze input data and deduce the transitions in system state over time, including changes indicative of faults.

#### 10.4. Prognostic Functionality

Once a fault is detected and diagnosed, a prognostic algorithm appropriate to the type of the fault is invoked. For battery capacity deterioration, as well as for charge estimation, an algorithm based on the Particle Filter framework is planned to be used (Saha & Goebel, 2009) and (Saha et al., 2011). Prognostic estimation of temperature build-up inside the electric motors - which can lead to winding insulation deterioration and eventual failure - will be done using a Gaussian Process Regression algorithm (Balaban et al., 2011). Finally, work is in progress to implement prognostics for electronic components of the motor drive units (such as capacitors and power transistors) using Kalman Filter and Extended Kalman Filter approaches (Celaya, Saxena, & Saha, 2011).

### 11. VALIDATION EXPERIMENTS

The following section describes the scenarios used for validating the policy optimization algorithm, PPG, and the constraint redesign algorithm, DCR. Subsections 11.1 (Policy optimization) and 11.2 (Dynamic Constraint Redesign) are structured in a similar manner: formal scenario formulations are provided first, followed by descriptions of how the experiments were conducted, with the experimental results summarized last. Both of the algorithms have only been tested in simulation at this time.

#### 11.1. Policy optimization

Policy optimization experiments were developed around a scenario (**Scenario R1**, with 'R' denoting rover scenarios) where, for science operations, an unmanned planetary rover is tasked with visiting a certain number of locations. Each location has a scientific payoff (reward) value associated with it. The terrain is of variable elevation and the surface friction coefficient is considered to be constant. The rover has

a finite amount of energy available to complete the mission. At some point during the mission a system fault is detected (e.g., a deteriorating electrical connector) that limits the overall remaining useful life of the vehicle. We also assume that the degradation rate depends on the operating conditions (e.g. the amount of heat generated in the instrumentation compartment during the drive). Either depletion of energy or complete component failure signify EOL. The goal of the PDM system is to reassess the original mission plan and find a suitable (ideally, optimal) compromise between extending the life of the vehicle and achieving the maximum science payoff as possible.

#### 11.1.1. Scenario formulation

Given:

$$c_e, c_\eta$$

Inequality constraints on available energy and health

$$\vec{f}(\pi) = \{f_r(\pi), f_\eta(\pi), f_e(\pi)\}$$

Objective functions for cumulative reward, health degradation, and energy consumption

$$\vec{v} = \{v_r, v_h, v_e\}, (v_r, v_h, v_e \in [0, 1])$$

Optimization preferences vector

$$N = \{n_1, n_2, \dots, n_{|N|}\}$$

Nodes (locations) to be visited

$$a \triangleq \{n_i, n_j\}, i \in [1, 2, \dots, |N| - 1], j \in [2, \dots, |N| - 1]$$

An action constitutes a move between a pair of nodes (start and finish)

$$a_1 = \{n_1, n_i\}, i \neq 1$$

The first action of a decision is a special case (go from the current location, labeled  $n_1$ , to another node)

$$a_m = \{n_j, n_k\} | (a_{m-1} = \{n_i, n_j\}), (i, j, k \in [1, 2, \dots, |N|]), (m \in [2, 3, \dots, |N|])$$

Any action after the first one needs to start on the node where the previous one finished

Find:

$$\Pi^*$$

Pareto set of policies

#### 11.1.2. Design of experiments

A synthetic terrain map  $\mathcal{M}$  was generated (Figure 6) and ten waypoints (nodes) still to be visited by the rover were selected on it. Each node is associated with a reward value (shown in parenthesis). The bar on the right side of the map and the isolines depict the elevation changes.

Test scenarios with increasing numbers of remaining nodes (6-10) were then created. The nodes were selected in such a way so as to make it impossible for the vehicle to visit

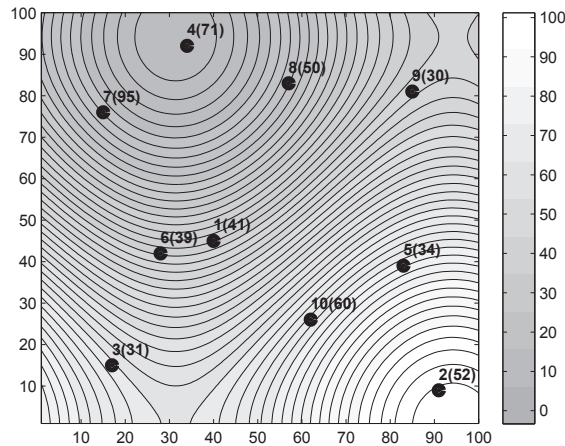


Figure 6. Terrain map with scientific target locations (elevations and distances are in meters)

all of them before either energy depletion or vehicle health deterioration resulted in EOL. PPG was allocated a limited number of utility function calls (UFC) to test performance in resource-constrained conditions. An exhaustive search algorithm (ES), used for verifying PPG results and benchmarking, was not limited in how many times it could invoke the utility function. The metric used for evaluating performance was the cumulative reward for the best path (policy) found by each algorithm. Each scenario was executed 30 times and the mean and standard deviations were computed. All of the code was written in MATLAB (R2010b) and executed on an Intel Core i7 Duo 2.8GHz computer.

### 11.1.3. Experimental results

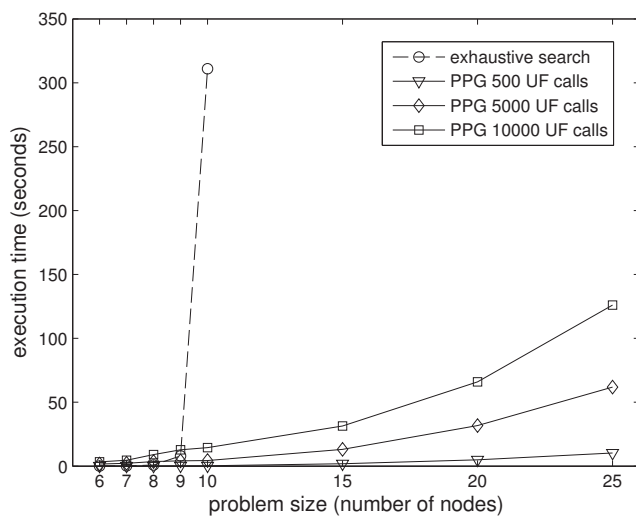


Figure 7. Mean execution times comparison

Table 4 summarizes the cumulative reward results obtained by ES and PPG. The number of utility function calls used by ES is provided for comparison. While not quite achieving scores as high as ES for the larger size problems, PPG still does relatively well, particularly given that in those scenarios it uses a small fraction of UFC used by ES (PPG performance improves, as expected, if more UFC are permitted).

Execution time for each of the algorithms was also recorded for all of the scenarios, with the data summarized in Table 5. It can be observed that execution times for ES start growing exponentially with problem size and using this approach becomes impractical for problems containing more than 10 nodes. While not having the ability to validate the cumulative reward performance on problems larger than that (in reasonable time), we still tested PPG with scenarios containing 15, 20, and 25 nodes. The average execution times are presented on Figure 7 and lead us to believe that the approach adopted for PPG remains practical for real-time applications even for policies with large numbers of actions (at least up to 25). The question of how to evaluate the quality of generated policies in large-size problems is something we hope to investigate in subsequent work.

### 11.2. Dynamic Constraint Redesign

To test the DCR algorithm, a scenario was used (**Scenario R2**) where one of the rover motors (FL) has experienced an *Increased Motor Friction* fault. This results in increased current consumption by the motor and, consequently, a higher rate of heat build-up both in it and the batteries supplying the current. For the purposes of this scenario the batteries are viewed as a single unit, with its temperature denoted by  $T_b$ . Temperature of the affected motor is denoted as  $T_m$ . Even given the fault, the rover is still required to travel a certain distance in a given amount of time in order to reach a point favorable for battery recharging and communication with controllers. To accomplish that, the rover needs to alternate periods of driving with periods of stationarity, in order to not exceed the maximum temperature limits for both the battery and the motor. The two components belong to Power (Po) and Propulsion (Pr) subsystems, respectively. As the components heat up and cool at different rates, a suitable schedule for driving and cooling down periods (policy) needs to be negotiated between the subsystems. As no acceptable policies may exist that satisfy both the minimum distance and the maximum time constraints, the two subsystems may need to negotiate increases in their operating temperature limits. It is in the interest of each subsystem to keep its limit as low as possible, in order to reduce the risk of failure. The rover, as a whole, is also interested in keeping the risk of component failure as low as possible, while still achieving the destination in the time allotted.

Table 4. Maximum cumulative reward values obtained by ES and PPG algorithms (in points)

nodes	ES UFC	ES result	500 UFC PPG mean ( $\sigma$ )	5000 UFC PPG mean ( $\sigma$ )	10000 UFC PPG mean ( $\sigma$ )
6	720	237	235.60 (05.33)	237.00 (00.00)	237.00 (00.00)
7	5040	311	295.80 (11.29)	305.77 (09.11)	305.77 (09.11)
8	40320	343	329.93 (08.51)	342.57 (02.37)	340.83 (04.93)
9	362880	373	326.27 (11.31)	345.47 (16.50)	348.87 (16.76)
10	3628800	403	347.47 (22.93)	382.73 (18.04)	388.97 (16.47)

Table 5. ES and PPG execution time (in seconds)

nodes	ES UFC	ES mean ( $\sigma$ )	500 UFC PPG mean ( $\sigma$ )	5000 UFC PPG mean ( $\sigma$ )	10000 UFC PPG mean ( $\sigma$ )
6	720	0.0192 (0.0003)	0.1756 (0.0049)	1.5961 (0.0183)	3.3543 (0.2253)
7	5040	0.1154 (0.0020)	0.2079 (0.0083)	2.2419 (0.1778)	4.5870 (0.4473)
8	40320	0.8385 (0.0105)	0.2212 (0.0114)	3.5883 (0.2132)	9.0177 (0.8424)
9	362880	8.0367 (0.0448)	0.2350 (0.0072)	4.1321 (0.1315)	12.7910 (0.3515)
10	3628800	310.9904 (3.9258)	0.2412 (0.0060)	4.4041 (0.2654)	14.4285 (0.7322)

### 11.2.1. Scenario formulation

Given:

$v_c = 0.3m/s$  the minimum velocity the rover can maintain without stalling, given the fault. Also assumed to be best (cruise) velocity in terms of energy efficiency

$T_{b,init} = 40^\circ C$  the initial operating temperature of the battery

$T_{m,init} = 35^\circ C$  the initial operating temperature of the motor

$T_{b,max_0} = 60^\circ C$  the initial operating temperature limit for the battery

$T_{m,max_0} = 60^\circ C$  the initial operating temperature limit for the motor

$T_a = 30^\circ C$  the ambient temperature (constant)

$I_s = 5A$  peak current drawn by the affected motor in order to reach  $v_c$  from full stop (start current)

$I_c = 2A$  current drawn by the affected motor at  $v_c$  (cruise current)

$d_{min} = 500m$  the minimum traverse distance

$t_{max} = 3600s$  the maximum time to reach the destination

$t_s = 2s$  the time needed to achieve cruise velocity from a complete stop

A notional current profile for the damaged motor is shown on Figure 8. For simplicity, current draw by the three healthy motors was assumed to be constant throughout the motion at 1A each. It is also assumed that prognostic information on battery and motor EOL is provided.

Find:

$t_d$  drive period duration

$t_c$  cooldown period duration

$T_{b,max_f}$  final operating temperature limit for the battery,  $T_{b,max_f} \in [T_{b,max_0}, \infty)$

$T_{m,max_f}$  final operating temperature limit for the motor,  $T_{m,max_f} \in [T_{m,max_0}, \infty)$

In this formulation  $t_d, t_c, T_{b,max_f}, T_{m,max_f}$  are the decision variables.

### 11.2.2. Design of experiments

Each subsystem was given a maximum of  $M = 3$  search iterations before it had to relinquish control of the process.  $t_d$  and  $t_c$  could be picked from intervals between 10 and 100s, in 10s increments. A simplified version of the simulator (tracking only the distance traveled and the temperature state of the affected motor and the battery) was used as the utility function, in order to speed up execution. The following general thermal state equation was used in the simulator:

$$dT = \frac{1}{C_t}(RI^2 + h(T_a - T))dt,$$

where  $T$  is the component temperature,  $C_t$  is the thermal capacity coefficient,  $R$  is the electrical resistance,  $I$  is the current,  $h$  is the heat transfer coefficient, and  $T_a$  is the ambient temperature. Model parameters used in the experiments are provided in Table 7.

Table 7. Model parameters

parameter	motor	battery	units
$C_t$	11	25	$\frac{J}{K}$
$R$	0.5	1.0	$\Omega$
$h$	0.03	0.08	$\frac{W}{K}$

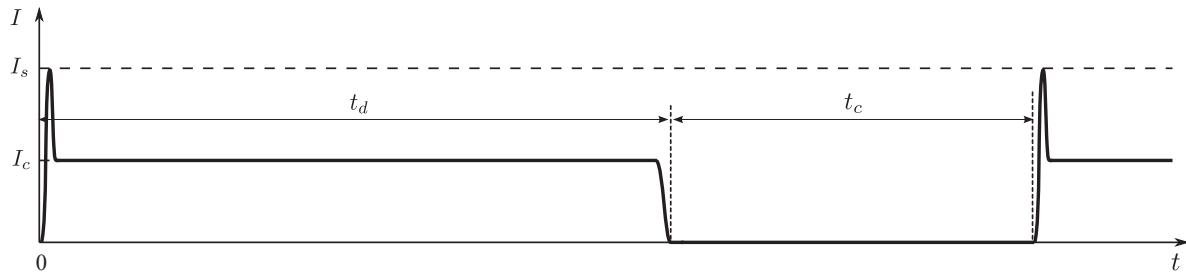


Figure 8. Current profile for the damaged motor

Table 6. DCR iterations in the example run

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
<i>active subsystem</i>	Po	Po	Po	Pr	Pr	Pr	Po	Po	Po	Pr	Pr	Pr	Po	Pr	Pr
$t_d(s)$	50	50	60	30	30	40	70	50	80	40	50	50	80	70	90
$t_c(s)$	80	80	90	90	90	100	100	60	80	90	100	80	80	90	100

Prognostic information was supplied in a differential form as the probability of reaching EOL:

$$dp_{EOL} = \frac{a}{10^{10}} T^3 dt,$$

where  $a = 1.5 \frac{1}{K^3}$  for the battery and  $a = 1.3 \frac{1}{K^3}$  for the motor.

The system probability of EOL was calculated as a weighted sum of the two component EOL probabilities:  $p_{system} = 0.8p_b + 0.2p_m$ . In this case the battery failure was considered to be a greater risk than a motor failure, as in the latter case the possibility of achieving the objective remained by using the remaining three motors. Minimization of risk of premature failure was included in both the local and the global components of the subsystem objective functions.

### 11.2.3. Experimental results

The output from one of the runs of the algorithm is presented on Figure 9 and in Table 6. The top subplot of Figure 9 shows the evolution of temperature constraints for the two subsystems throughout the negotiation process. The middle subplot shows the maximum distances achievable from each of the subsystems' point of view. The process ends when both of the subsystems are predicted to be capable of achieving  $d_{min}$ , albeit with a higher risk of failure while doing so. The bottom subplot shows the estimated risk of system failure for each iteration of the algorithm. Table 6 shows which subsystem had the control of the process during each iteration and the  $\{t_d, t_c\}$  pair it proposed as the best solution. In the example run given here, the final temperature limit for the battery was found to be at approximately  $65.3^\circ C$  and the one for the motor at approximately  $75.5^\circ C$ .

## 12. SUMMARY AND FUTURE WORK

In this paper we outlined our approach to development of prognostic decision making methods for aerospace applica-

tions. First, definitions for prognostic decision making and related concepts were suggested, then a few motivating examples (highlighting potential use cases for PDM) were described. The examples also helped to illustrate the general attributes of the problem type we hope to address: (1) complex, multi-component systems; (2) dynamic operating environments; (3) degradation/fault modes that evolve in their characteristics over time and have the potential of substantially affecting system performance; and (4) decisions on mitigation measures required in a finite amount of time. From there we derived our set of high-level requirements for aerospace PDM systems. With these requirements in mind, we reviewed related prior efforts from the areas of prognostics-enabled control, post-prognostic decision support, condition-based maintenance, and automated contingency management. We then explained our process for selecting suitable policy generation techniques and presented a prototype algorithm that uses probabilistic methods and prognostic information in generation of action policies. The algorithm, PPG, was tested against an exhaustive search algorithm on scenarios involving a planetary rover prototype. We also considered the problem where no feasible policies are found or where feasible policies in the generated Pareto set are not sufficient for attaining performance objectives, given the current system constraints. We proposed that this problem has certain common characteristics with problems from the field of Multidisciplinary Design Optimization and reviewed some of the modern MDO approaches for applicability. One of the approaches is based on game-theoretic principles and served as a foundation for the second algorithm presented, DCR. This algorithm sets up a negotiating framework for subsystems to adjust their operating constraints, if that becomes necessary for achievement of high-importance system objectives. DCR was demonstrated on a problem involving two subsystems, power and propulsion.



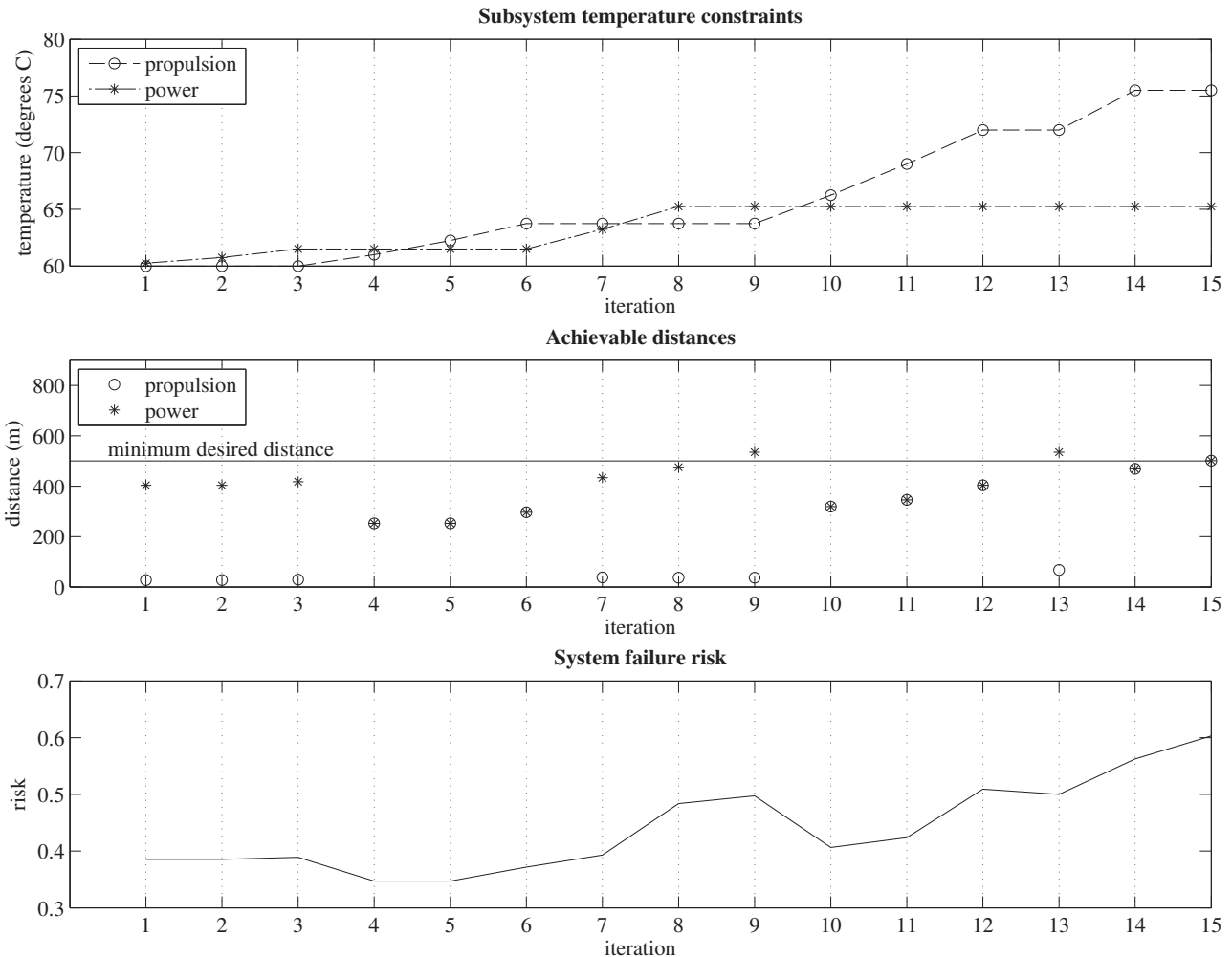


Figure 9. DCR output example

While it is not possible to cover all of the topics discussed in sufficient detail in one paper, we hope that it provides a good foundation for future efforts. The work done so far also gave us a better appreciation for the challenges ahead. One of them is developing more efficient multi-objective optimization algorithms - given the high computational cost of a utility function (simulation) call in a typical application. We plan to continue our development of probabilistic optimization methods and further investigate applicability of evolutionary algorithms. Use of multi-fidelity models and response surfaces for utility simulation will be researched as well.

For the problem of DCR, we plan to concentrate on the following three goals: (1) extend the current, cooperative game DCR algorithm to greater possible numbers of players/subsystems; (2) investigate other formulations, possibly based on ideas in CO, CSSO, and BLISS; (3) develop methods for selection of those constraints that offer the most system benefit if revised (approaches based on Lagrangian Du-

ality appear promising for this purpose). We also hope that decomposition formulations researched for DCR will also prove helpful for the prognostic policy generation work. Finally, identifying and, if necessary, developing suitable performance metrics will become more important as complexity of test scenarios and algorithms increases.

### 13. ACKNOWLEDGMENT

The authors would like to thank their colleagues at NASA Ames Research Center, Stanford University, and Los Alamos National Laboratory for stimulating discussions and assistance with this work. Funding from NASA System-Wide Safety Assurance Technologies (SSAT) project (Aviation Safety Program) is also gratefully acknowledged.

### REFERENCES

Agte, J. S., Weck, O., Sobieszczanski-Sobieski, J., Arend-

- sen, P., Morris, A., & Spieck, M. (2009, April). MDO: Assessment and Direction for Advancement - an Opinion of One International Group. *Structural and Multidisciplinary Optimization*, 40(1-6), 17–33. doi: 10.1007/s00158-009-0381-5
- Allison, J., Kokkolaras, M., Zawislak, M., & Papalambros, P. (2005). On the use of analytical target cascading and collaborative optimization for complex system design. In *6th world congress on structural and multidisciplinary optimization* (pp. 1–10). Rio de Janeiro, Brazil.
- Balaban, E., Narasimhan, S., Daigle, M., Celaya, J., Roychoudhury, I., & Saha, B. (2011). A Mobile Robot Testbed for Prognostics-Enabled Autonomous Decision Making. In *Annual conference of the prognostics and health management society* (pp. 1–16). Montreal, Canada.
- Benedettini, O., Baines, T. S., Lightfoot, H. W., & Greenough, R. M. (2009, March). State-of-the-art in integrated vehicle health management. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 223(2), 157–170. doi: 10.1243/09544100JAERO446
- Bertsimas, D., & Tsitsiklis, J. (1993). Simulated Annealing. *Statistical Science*, 8(1), 10–15.
- Blouin, V. Y., Lassiter, J. B., Wiecek, M. M., & Fadel, G. M. (2005). Augmented Lagrangian Coordination for Decomposed Design Problems. In *6th world congress on structural and multidisciplinary optimization* (pp. 1–10). Rio de Janeiro, Brazil.
- Bogdanov, A., Chiu, S., Gokdere, L. U., & Vian, J. (2006). Stochastic Optimal Control of a Servo Motor with a Lifetime Constraint. In *Proceedings of the 45th IEEE conference on decision and control* (pp. 4182–4187). doi: 10.1109/CDC.2006.377205
- Bole, B. (2012). Using Markov Models of Fault Growth Physics and Environmental Stresses to Optimize Control Actions. In *Aiaa infotech @ aerospace* (pp. 1–7). Garden Grove, CA.
- Bole, B., Tang, L., Goebel, K., & Vachtsevanos, G. (2011). Adaptive Load-Allocation for Prognosis-Based Risk Management. In *Annual conference of the prognostics and health management society* (pp. 1–10).
- Boularias, A. (2010). *Predictive Representations For Sequential Decision Making Under Uncertainty*. Unpublished doctoral dissertation, Laval University.
- Braun, R., Gage, P., Kroo, I., & Sobiesky, I. (1996). *Implementation and Performance Issues in Collaborative Optimization* (Tech. Rep.). NASA Langley Research Center.
- Brown, D. W., Georgoulas, G., & Bole, B. (2009). Prognostics Enhanced Reconfigurable Control of Electro-Mechanical Actuators. In *Annual conference of the prognostics and health management society* (pp. 1–17). Denver, CO.
- Brown, D. W., & Vachtsevanos, G. J. (2011). A Prognostic Health Management Based Framework for Fault-Tolerant Control. In *Annual conference of the prognostics and health management society*. Montreal, Canada.
- Brown, N. (2004). *Evaluation of Multidisciplinary Optimization (MDO) Techniques Applied to a Reusable Launch Vehicle* (Tech. Rep.). Atlanta, GA: Georgia Institute of Technology.
- Bryce, D., & Cushing, W. (2007). *Probabilistic planning is multi-objective* (Tech. Rep. No. Figure 1). Artificial Intelligence Center, SRI International, Inc.
- Celaya, J., Saxena, A., & Saha, S. (2011). Prognostics of Power MOSFETs under Thermal Stress Accelerated Aging using Data-Driven and Model-Based Methodologies. In *Annual conference of the prognostics and health management society* (pp. 1–10). Montreal, Canada.
- Clarich, A., & Pediroda, V. (2004). A Competitive Game Approach for Multi-Objective Robust Design Optimization. In *1st intelligent systems technical conference*. Chicago, IL.
- Coello, C., Lamont, G., & Veldhuizen, D. V. (2007). *Evolutionary algorithms for solving multi-objective problems* (2nd ed.). Springer.
- Cramer, E. J., Dennis, J. E. J., Frank, P. D., Shubin, G. R., & Lewis, R. M. (1993). Problem Formulation for Multidisciplinary Optimization. In *Aiaa symposium on multidisciplinary design optimization* (Vol. 4). SIAM. doi: 10.1137/0804044
- Daigle, M., & Goebel, K. (2010, March). Model-based prognostics under limited sensing. *2010 IEEE Aerospace Conference*, 1–12. doi: 10.1109/AERO.2010.5446822
- Daigle, M., & Roychoudhury, I. (2010). Qualitative Event-Based Diagnosis: Case Study on the Second International Diagnostic Competition. In (pp. 1–8).
- Das, A., & Chakrabarti, B. (2005). *Quantum Annealing and Related Optimization Methods (Lecture Notes in Physics)*. Springer.
- Delgado, I., Dempsey, P., & Simon, D. (2012). *A Survey of Current Rotorcraft Propulsion Health Monitoring Technologies (NASA/TM2012-217420)* (Tech. Rep. No. January). Cleveland, OH: NASA Glenn Research Center.
- Denney, E., & Pai, G. (2012). Perspectives on Software Safety Case Development for Unmanned Aircraft. In *Ieee/ifip international conference on dependable systems and networks (dsn)*. Boston, MA.
- Driessen, B., & Kwok, K. (1998). A multiobjective dynamic programming approach to constrained discrete-time optimal control. In *Proceedings of the 1998 American control conference. acc (IEEE cat. no.98ch36207)*

- (Vol. 5, pp. 3077–3083). American Autom. Control Council. doi: 10.1109/ACC.1998.688424
- Edwards, D., Orchard, M. E., Tang, L., Goebel, K., & Vachtsevanos, G. (2010). Impact of Input Uncertainty on Failure Prognostic Algorithms : Extending the Remaining Useful Life of Nonlinear Systems. In *Annual conference of the prognostics and health management society* (pp. 1–7). Portland, OR.
- Haddad, G., Sandborn, P., & Pecht, M. (2011a). A Real Options Optimization Model To Meet Availability Requirements For Offshore Wind Turbines. In *Mfpt* (pp. 1–12). Virginia Beach, VA.
- Haddad, G., Sandborn, P., & Pecht, M. (2011b, June). Using real options to manage condition-based maintenance enabled by PHM. In *2011 IEEE conference on prognostics and health management* (pp. 1–7). Denver, CO: Ieee. doi: 10.1109/ICPHM.2011.6024318
- Hestenes, M. R. (1969, November). Multiplier and gradient methods. *Journal of Optimization Theory and Applications*, 4(5), 303–320. doi: 10.1007/BF00927673
- Hogge, E., Quach, C., Vazquez, S., & Hill, B. (2011). *A Data System for a Rapid Evaluation Class of Subscale Aerial Vehicle, NASA/TM2011-217145* (Tech. Rep. No. May). Hampton, VA: NASA Langley Research Center.
- Honda, T., Ciucci, F., Lewis, K., & Yang, M. (2010). A Comparison of Information Passing Strategies in System Level Modeling. *International Design*, 1–10.
- Hussein, M. L., & Abo-Sinna, M. A. (1993, November). Decomposition of multiobjective programming problems by hybrid fuzzy-dynamic programming. *Fuzzy Sets and Systems*, 60(1), 25–32. doi: 10.1016/0165-0114(93)90286-Q
- Iyer, N., Goebel, K. F., & Bonissone, P. (2006). Framework for Post-Prognostic Decision Support. *2006 IEEE Aerospace Conference*, 1–10. doi: 10.1109/AERO.2006.1656108
- Janasak, K., & Beshears, R. (2007). Diagnostics to prognostics—a product availability technology evolution. *Reliability and Maintainability Symposium, 2007*, 113–118.
- Kasmiran, K. A., Zomaya, A. Y., Mazari, A. A., & Garcia, R. J. (2010, October). SVM-enabled prognostic method for clinical decision making: The use of CD4 T-cell level and HIV-1 viral load for guiding treatment initiation and alteration. In *2010 IEEE 23rd international symposium on computer-based medical systems (cbms)* (pp. 19–25). IEEE.
- Kawaguchi, J., Uesugi, K., & Fujiwara, A. (2003, January). The MUSES-C mission for the sample and returns technology development status and readiness. *Acta Astronautica*, 52(2-6), 117–123. doi: 10.1016/S0094-5765(02)00146-7
- Kim, H. M. (2001). *Target Cascading in Optimal System Design*. Unpublished doctoral dissertation, The University of Michigan.
- Kim, H. M., Michelena, N. F., Papalambros, P. Y., & Jiang, T. (2003). Target Cascading in Optimal System Design. *Journal of Mechanical Design*, 125(3), 474. doi: 10.1115/1.1582501
- Kveton, B., Hauskrecht, M., & Guestrin, C. (2006). Solving factored MDPs with hybrid state and action variables. *Journal of Artificial Intelligence Research*, 27, 153–201.
- Lewis, K., & Mistree, F. (1997). Modeling Interactions in Multidisciplinary Design: A Game Theoretic Approach. *AIAA Journal*, 35(8), 1387–1392.
- Liao, L. (2002). Adaptive differential dynamic programming for multiobjective optimal control. *Automatica*, 1–27.
- Malikopoulos, A. (2007). A State-Space Representation Model and Learning Algorithm for Real-Time Decision-Making Under Uncertainty. In *Proceedings of the asme international mechanical engineering congress and exposition* (pp. 1–10). Seattle, WA.
- Marston, M. C. (2000). *Game Based Design: a Game Theory Based Approach to Engineering Design* (Vol. 0). Unpublished doctoral dissertation, Georgia Institute of Technology.
- Narasimhan, S., Balaban, E., Daigle, M., Roychoudhury, I., Sweet, A., Celaya, J., & Goebel, K. (2012). Autonomous Decision Making for Planetary Rovers Using Diagnostic and Prognostic Information. In *The 8th international federation of automatic control symposium (safeprocess)*. Mexico City, Mexico.
- Narasimhan, S., & Brownston, L. (2007). HyDE A General Framework for Stochastic and Hybrid Model-based Diagnosis. In *Proceedings of the international workshop on the principles of diagnosis*. Nashville, TN.
- NTSB. (1989). *Aircraft Accident Report PBSO-910406 NTSB/AAR-SO/06* (Tech. Rep.). Washington, DC.
- Peek, N. (1998). Predictive Probabilistic Models for Treatment Planning in Paediatric Cardiology. In *Computational engineering in systems applications*.
- Pereira, E. B., Galvao, R. K. H., & Yoneyama, T. (2010, July). Model Predictive Control using Prognosis and Health Monitoring of actuators. In *2010 IEEE international symposium on industrial electronics* (pp. 237–243). IEEE. doi: 10.1109/ISIE.2010.5637571
- Räisänen, J., & Palmer, T. (2001). A probability and decision-model analysis of a multimodel ensemble of climate change simulations. *Journal of Climate*, 14(15), 3212–3226.
- Rao, S. S. (2009). *Engineering Optimization*. Hoboken, NJ, USA: John Wiley and Sons, Inc. doi: 10.1002/9780470549124
- Rao, S. S., & Freiheit, T. I. (1991). A Modified Game Theory Approach to Multiobjective Optimization. *Journal of Mechanical Design*, 113(September), 286–291.
- Renaud, G., & Shi, G. (2002). Evaluation and implemen-

- tation of multidisciplinary design optimization strategies. In *Congress of the international council of the aeronautical sciences (icas)* (pp. 1–10).
- Renaud, J. E., & Gabriele, G. A. (1993). Improved Coordination in Nonhierarchical System Optimization. *AIAA Journal*, 31(12), 2367–2373. doi: 10.2514/3.11938
- Reveley, M. S., Leone, K. M., Briggs, J. L., & Withrow, C. A. (2010). *Assessment of the State of the Art of Integrated Vehicle Health Management Technologies as Applicable to Damage Conditions (NASA/TM2010-216911)* (Tech. Rep. No. December). Cleveland, OH: NASA Glenn Research Center.
- Roth, B. D. (2008). Enhanced Collaborative Optimization: Application to an Analytic Test Problem and Aircraft Design. In *Analysis and optimization conference* (pp. 1–14).
- Roth, B. D., & Kroo, I. M. (2008). Enhanced Collaborative Optimization: A Decomposition-Based Method for Multidisciplinary Design. In *34th design automation conference*, (Vol. 2008, pp. 927–936). Brooklyn, NY: ASME. doi: 10.1115/DETC2008-50038
- Saha, B., & Goebel, K. F. (2009). Modeling Li-ion Battery Capacity Depletion in a Particle Filtering Framework. In *Annual conference of the prognostics and health management society* (pp. 1–10). San Diego, CA: PHM Society.
- Saha, B., Koshimoto, E., Quach, C. C., Hogge, E., Strom, T. H., Hill, B. L., ... Goebel, K. F. (2011). Battery health management system for electric UAVs. *2011 IEEE Aerospace Conference*, 1–9. doi: 10.1109/AERO.2011.5747587
- Saxena, A., Celaya, J., Balaban, E., Goebel, K. F., Saha, B., Saha, S., & Schwabacher, M. (2008, October). Metrics for evaluating performance of prognostic techniques. *2008 International Conference on Prognostics and Health Management*, 1–17. doi: 10.1109/PHM.2008.4711436
- Sellar, R., Batill, S. M., & Renaud, J. E. (1996). Response Surface Based, Concurrent Subspace Optimization For Multidisciplinary System Design. *34TH AIAA Aerospace Sciences Meeting and Exhibit*, 96 – 714.
- Shan, S., & Wang, G. G. (2009, August). Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions. *Structural and Multidisciplinary Optimization*, 41(2), 219–241. doi: 10.1007/s00158-009-0420-2
- Simon, H. A. (1956). Rational choice and the structure of the environment. *Psychological Review*, 63(2)(March), 129–138.
- Sobieszczanski-Sobieski, J., Agte, J. S., & Sandusky, R. (1998). Bi-Level Integrated System Synthesis. *AIAA Journal*, 38(August), 164–172.
- Sobieszczanski-Sobieski, J., Emiley, M. S., Agte, J. S., & Sandusky, R. (2000). Advancement of Bi-Level System Synthesis (BLISS). In *Aerospace sciences meeting and exhibit*. Reno, NV.
- Tamiz, M., Jones, D., & Romero, C. (1998, December). Goal programming for decision making: An overview of the current state-of-the-art. *European Journal of Operational Research*, 111(3), 569–581. doi: 10.1016/S0377-2217(97)00317-2
- Tang, L., Hettler, E., Zhang, B., & Decastro, J. (2011). A Testbed for Real-Time Autonomous Vehicle PHM and Contingency Management Applications. In *Annual conference of the prognostics and health management society* (pp. 1–11).
- Tang, L., Kacprzyński, G. J., Goebel, K., Reimann, J., Orchard, M. E., Saxena, A., & Saha, B. (2007). Prognostics in the Control Loop. In *Working notes of 2007 fall aaii symposium* (pp. 128–135).
- Tosserams, S., Etman, L. F. P., Papalambros, P. Y., & Rooda, J. E. (2005). Augmented Lagrangian Relaxation for Analytical Target Cascading. In *6th world congress on structural and multidisciplinary optimization* (pp. 1–11).
- Vincent, T. L. (1983). Game Theory as a Design Tool. *ASME Journal Of Mechanisms, Transmissions, and Automation in Design*, 105pp, 165–170.
- Wang, L., & Zhu, J. (2008, May). Financial market forecasting using a two-step kernel learning method for the support vector regression. *Annals of Operations Research*, 174(1), 103–120.
- Wolpert, D. H., Strauss, C. E. M., & Rajnarayan, D. (2006). Advances in Distributed Optimization Using Probability Collectives. *Advances in Complex Systems*, 9(4), 383–436.

# An Integrated Framework for Model-Based Distributed Diagnosis and Prognosis

Anibal Bregon<sup>1</sup>, Matthew Daigle<sup>2</sup>, and Indranil Roychoudhury<sup>3</sup>

<sup>1</sup> *University of Valladolid, Valladolid, Spain*  
*anibal@infor.uva.es*

<sup>2</sup> *NASA Ames Research Center, Moffett Field, CA 94035, USA*  
*matthew.j.daigle@nasa.gov*

<sup>3</sup> *SGT Inc., NASA Ames Research Center, Moffett Field, CA 94035, USA*  
*indranil.roychoudhury@nasa.gov*

## ABSTRACT

Diagnosis and prognosis are necessary tasks for system re-configuration and fault-adaptive control in complex systems. Diagnosis consists of detection, isolation and identification of faults, while prognosis consists of prediction of the remaining useful life of systems. This paper presents a novel integrated framework for model-based distributed diagnosis and prognosis, where system decomposition is used to enable the diagnosis and prognosis tasks to be performed in a distributed way. We show how different submodels can be automatically constructed to solve the local diagnosis and prognosis problems. We illustrate our approach using a simulated four-wheeled rover for different fault scenarios. Our experiments show that our approach correctly performs distributed fault diagnosis and prognosis in an efficient and robust manner.

## 1. INTRODUCTION

Systems health monitoring is essential to guaranteeing the safe, efficient, and reliable operation of engineering systems. Integrated systems health management methodologies include fault diagnosis and prognosis mechanisms, where diagnosis involves *detecting* when a fault has occurred, *isolating* the true fault, and *identifying* the true damage to the system; and prognosis involves *predicting* how much useful life remains in the different components, subsystems, or systems given the diagnosed fault conditions. The information on the fault size and its expected impact on system life can be used to initiate recovery and reconfiguration actions that mitigate the fault or extend system life.

---

Anibal Bregon et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

A large body of research exists for both model-based diagnosis (Gertler, 1998; Blanke et al., 2006) and prognosis methods (Luo et al., 2008; Saha & Goebel, 2009; Orchard & Vachtsevanos, 2009). However, the integration of diagnosis and prognosis algorithms is seldom studied. In fact, many diagnosis methodologies leave out the fault identification step, which is necessary to perform a prediction from the current system state. Recently, we presented an integrated model-based framework for diagnosis and prognosis of complex systems, in which we made use of a common modeling framework for modeling both the nominal and faulty system behavior (Roychoudhury & Daigle, 2011).

In (Roychoudhury & Daigle, 2011), the nominal system behavior is estimated using an observer built with the nominal model. Faults are detected when a statistically significant deviation between the nominal estimates and the observed measurements is observed (Biswas et al., 2003). Fault isolation compares the observed measurement deviations against predictions of how the measurements would deviate for each possible fault (Mosterman & Biswas, 1999). Fault identification performs joint state-parameter estimation using multiple observers, where, for each fault, the faulty system model is constructed as the nominal model integrated with a hypothesized fault model (Roychoudhury, 2009). The prognosis module uses, for each fault hypothesis, a prediction model based on its faulty system model and the identified fault parameters, to predict the remaining useful life of the system (Daigle, Saha, & Goebel, 2012). However, this integrated solution performs the diagnosis and prognosis task in a centralized fashion, which is prone to single points of failure, and does not scale well as the size of the system increases.

To overcome such problems, in this work, we leverage recent results for distributed diagnosis (Bregon et al., 2011)

and distributed prognosis (Daigle, Bregon, & Roychoudhury, 2012), which make use of structural model decomposition techniques, to provide a systematic approach to distributing the different diagnosis and prognosis steps presented in (Roychoudhury & Daigle, 2011).

Distributed diagnosis is achieved by designing local distributed subsystems based on global diagnosability analysis of the system, thus computing globally correct distributed diagnosis results without the use of a centralized coordinator (Bregon et al., 2011). These local distributed subsystems are then used to construct local event-based distributed diagnosers for distributed fault isolation. Distributed fault identification is achieved by developing independent local state-parameter estimators for each hypothesized fault. Regarding distributed prediction, in (Daigle, Bregon, & Roychoudhury, 2012) we developed an architecture that enables a large prognosis problem to be decomposed into several independent local subproblems from which local results can be merged into a global result.

The main contribution of this paper is an integrated framework for distributed model-based diagnosis and prognosis of single faults based on structural model decomposition. The proposed framework scales well and the resulting subproblems are typically small and easy to solve, resulting in an efficient and scalable distributed solution to the combined diagnosis and prognosis problem. We perform a number of experiments on a simulated four-wheeled rover testbed (Balaban et al., 2011) to demonstrate and evaluate our approach.

The rest of the paper is organized as follows. Section 2 provides the problem formulation for our diagnosis and prognosis framework. Section 3 describes the distributed architecture and Section 4 briefly introduces its different components. Section 5 presents the case study and experimental results. Finally, Section 6 concludes the paper.

## 2. PROBLEM FORMULATION

The *nominal system model* is represented as follows:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{f}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{v}(t)), \\ \mathbf{y}(t) &= \mathbf{h}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{n}(t)),\end{aligned}$$

where  $\mathbf{x}(t) \in \mathbb{R}^{n_x}$  is the state vector,  $\boldsymbol{\theta}(t) \in \mathbb{R}^{n_\theta}$  is the parameter vector,  $\mathbf{u}(t) \in \mathbb{R}^{n_u}$  is the input vector,  $\mathbf{v}(t) \in \mathbb{R}^{n_v}$  is the process noise vector,  $\mathbf{f}$  is the state equation,  $\mathbf{y}(t) \in \mathbb{R}^{n_y}$  is the output vector,  $\mathbf{n}(t) \in \mathbb{R}^{n_n}$  is the measurement noise vector, and  $\mathbf{h}$  is the output equation.<sup>1</sup>

Faults in the system are represented as changes in the above nominal system model. In this work, we only consider single faults occurring as changes in system parameters,  $\boldsymbol{\theta}(t)$ . We denote a fault,  $f \in F$ , as a tuple,  $(\theta, g_f)$ , where,  $\theta \in \boldsymbol{\theta}$

is the *fault parameter*, and  $g_f$  denotes the *fault progression function*, which models the way fault  $f$  is manifested in parameter  $\theta$ , i.e.,

$$\dot{\theta}(t) = g_f(t, \mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t), \mathbf{m}_f(t)),$$

where  $\mathbf{x}_f(t) = [\mathbf{x}(t), \theta(t)]^T$ ,  $\boldsymbol{\theta}_f(t) = [\boldsymbol{\theta}(t) \setminus \{\theta(t)\}, \phi_f(t)]^T$ ,  $\phi_f(t) \in \mathbb{R}^{n_{\phi_f}}$  is a vector of *fault progression parameters*, and  $\mathbf{m}_f(t) \in \mathbb{R}^{n_{m_f}}$  is a process noise vector associated with the fault progression function.

To develop our integrated diagnosis and prognosis framework, the *faulty system model* for fault  $f = (\theta, g_f)$  is constructed from the nominal system model by including the parameter as a state and augmenting the state equation by including the fault progression function, i.e.,

$$\begin{aligned}\dot{\mathbf{x}}_f(t) &= \mathbf{f}_f(t, \mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t), \mathbf{v}(t)), \\ \mathbf{y}(t) &= \mathbf{h}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{n}(t)),\end{aligned}$$

where,

$$\mathbf{f}_f(\cdot) = \begin{bmatrix} \mathbf{f}(t, \mathbf{x}(t), \boldsymbol{\theta}(t), \mathbf{u}(t), \mathbf{v}(t)) \\ g_f(t, \mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t), \mathbf{m}(t)) \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\theta}(t) \end{bmatrix}$$

The goal of diagnosis is to: (i) detect a change in some  $\theta \in \boldsymbol{\theta}$ ; (ii) isolate, under the single fault assumption, the true fault  $f \in F$ , i.e., both the parameter  $\theta$  that has changed, and its fault progression function  $g_f$ ; and (iii) identify (i.e. estimate) the fault by computing  $p(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t) | \mathbf{y}(0:t))$ , where  $\mathbf{y}(0:t)$  denotes all measurements observed up to time  $t$ .

The goal of prognosis is to determine the end of (useful) life (EOL) of a system, and/or its remaining useful life (RUL). For a given fault,  $f$ , using the fault estimate,  $p(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t) | \mathbf{y}(0:t))$ , a probability distribution of EOL,  $p(\text{EOL}_f(t_P) | \mathbf{y}(0:t_P))$ , and/or RUL,  $p(\text{RUL}_f(t_P) | \mathbf{y}(0:t_P))$  is computed at a given time point  $t_P$  (Daigle, Saha, & Goebel, 2012). Since the prognosis problem is stochastic, EOL/RUL are random variables and we represent them by probability distributions. The acceptable behavior of the system is expressed through a set of  $n_c$  constraints,  $C_{\text{EOL}_f} = \{c_i\}_{i=1}^{n_c}$ , where  $c_i : \mathbb{R}^{n_{x_f}} \times \mathbb{R}^{n_{\theta_f}} \times \mathbb{R}^{n_u} \rightarrow \mathbb{B}$  maps a given point in the joint state-parameter space given the current inputs,  $(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t))$ , to the Boolean domain  $\mathbb{B} \triangleq [0, 1]$ , where  $c_i(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t)) = 1$  if the constraint is satisfied (Daigle, Saha, & Goebel, 2012). If  $c_i(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t)) = 0$ , then the constraint is not satisfied, and the behavior of the system is deemed to be unacceptable. These individual constraints are combined into a single *threshold function*  $T_{\text{EOL}_f} : \mathbb{R}^{n_{x_f}} \times \mathbb{R}^{n_{\theta_f}} \times \mathbb{R}^{n_u} \rightarrow \mathbb{B}$ ,

<sup>1</sup>Here, we use bold typeface to denote vectors, and use  $n_a$  to denote the length of a vector  $\mathbf{a}$ .

defined as

$$T_{EOL_f}(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t)) = \begin{cases} 1, & 0 \in \{c_i(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t))\}_{i=1}^{n_c} \\ 0, & \text{otherwise.} \end{cases}$$

So,  $EOL_f$  may be defined as

$$EOL_f(t_P) \triangleq \inf\{t \in \mathbb{R} : t \geq t_P \text{ and } T_{EOL_f}(\mathbf{x}_f(t), \boldsymbol{\theta}_f(t), \mathbf{u}(t)) = 1\},$$

i.e., EOL is the earliest time point at which the threshold is reached. RUL is expressed given EOL as

$$RUL_f(t_P) \triangleq EOL_f(t_P) - t_P.$$

### 3. DISTRIBUTED ARCHITECTURE

For a large system, both the diagnosis and prognosis problems are correspondingly large. A centralized approach does not scale well, can be computationally expensive, and prone to single points of failure. Therefore, we propose to decompose the *global* integrated diagnosis and prognosis problem into independent *local* subproblems. In this work, we build on the ideas from structural model decomposition (Blanke et al., 2006; Pulido & Alonso-González, 2004) to compute local independent subproblems, which may be solved in parallel, thus providing scalability and efficiency. Model decomposition is not a new concept, and several approaches have been developed for purposes of system identification, estimation, learning, and diagnosis (Staroswiecki & Declerck, 1989; Pulido & Alonso-González, 2004; Williams & Millar, 1998). Structural model decomposition allows decomposing a global model into a set of local submodels for which local diagnosis and prognosis problems can be directly defined. The global model of the system, denoted as  $\mathcal{M}$ , is defined as follows.

**Definition 1 (Model).** The model of a system,  $\mathcal{M}$ , is a tuple  $\mathcal{M} = (X, \Theta, U, Y, C)$ , where  $X$  is the set of state variables of  $\mathbf{x}$ ,  $\Theta$  is the set of unknown parameters of  $\boldsymbol{\theta}$ ,  $U$  is the set of input variables of  $\mathbf{u}$ ,  $Y$  is the set of output variables of  $\mathbf{y}$ , and  $C$  is the set of model constraints of  $\mathbf{f}$ ,  $\mathbf{h}$ , and EOL constraints of  $C_{EOL_f}$ .

The basic idea of the model decomposition problem is to decompose the global system model into a set of submodels satisfying given constraints, such that each submodel contains sufficient analytical redundancy to generate fault hypotheses from observed measurement deviations. A submodel in our framework is defined as follows.

**Definition 2 (Submodel).** A submodel  $\mathcal{M}_i$  of a system model  $\mathcal{M} = (X, \Theta, U, Y, C)$  is a tuple  $\mathcal{M}_i = (X_i, \Theta_i, U_i, Y_i, C_i)$ , where  $X_i \subseteq X$ ,  $\Theta_i \subseteq \Theta$ ,  $U_i \subseteq X \cup U \cup Y$ , and  $Y_i \subseteq Y$  are the state, parameter, input, and output variables, respectively,

and  $C_i \subseteq C$  are the submodel constraints.<sup>2</sup>

Next, we discuss the fundamental ideas of our model decomposition approach and show the constraints needed to obtain the different submodels for distributed diagnosis and prognosis. Then, we propose our integrated approach.

#### 3.1. Model Decomposition for Distributed Diagnosis and Prognosis

Model decomposition in our framework can be accomplished by using some variables (either measured variables or variables for which the values are known) as local inputs,  $U_i$ , such that each one of the submodels satisfies a set of constraints and contains the minimum number of equations to compute a given set of outputs,  $Y_i$ . As a result, submodels computed this way contain only a small subset of the equations of the model that are decoupled from the rest of the system model equations. In general, any set of variables in the system can be chosen as the local inputs to the submodel,  $U_i$ . The choice of  $U_i$  and the constraints to fulfill depends on the particular problem to be solved through model decomposition.

The first model decomposition problem computes minimal submodels from the nominal system model. For this model decomposition problem, constraints are that submodels use the global model inputs and some measured values as local inputs, i.e.,  $U_i \subseteq U \cup (Y - Y_i)$ . An algorithm for computing the set of minimal submodels that satisfies these properties is given in (Daigle et al., 2011), which is based on the model decomposition algorithms presented in (Pulido & Alonso-González, 2004; Bregon et al., 2012).

The second model decomposition problem computes submodels for residual generation and fault isolation. For this model decomposition problem, the constraints are that submodels are constructed by merging the minimal submodels, to fulfill global diagnosability conditions as in (Bregon et al., 2011).<sup>3</sup> Once the globally diagnosable subsystems have been designed, the merged submodels are used for distributed residual generation and to compute event-based local diagnosers for fault isolation. These design and diagnoser computation processes are detailed in (Bregon et al., 2011). Because the subsystems are designed to be globally diagnosable, the resulting local diagnosers are independent, and can provide globally correct diagnosis results without a centralized coordinator.

The third model decomposition problem computes submodels for distributed fault identification. For each consistent fault hypothesis  $f$ , the joint state-parameter estimators are computed from the minimal submodels of the faulty system model  $\mathcal{M}(f)$  with the constraints that  $U_i \subseteq U \cup (Y - Y_i)$  with

<sup>2</sup>A faulty submodel for a fault  $f$  is denoted as  $\mathcal{M}_i(f)$ .

<sup>3</sup>In this work, a subsystem is globally diagnosable if all faults in the subsystem are distinguishable from every other fault in the system using only local measurements.

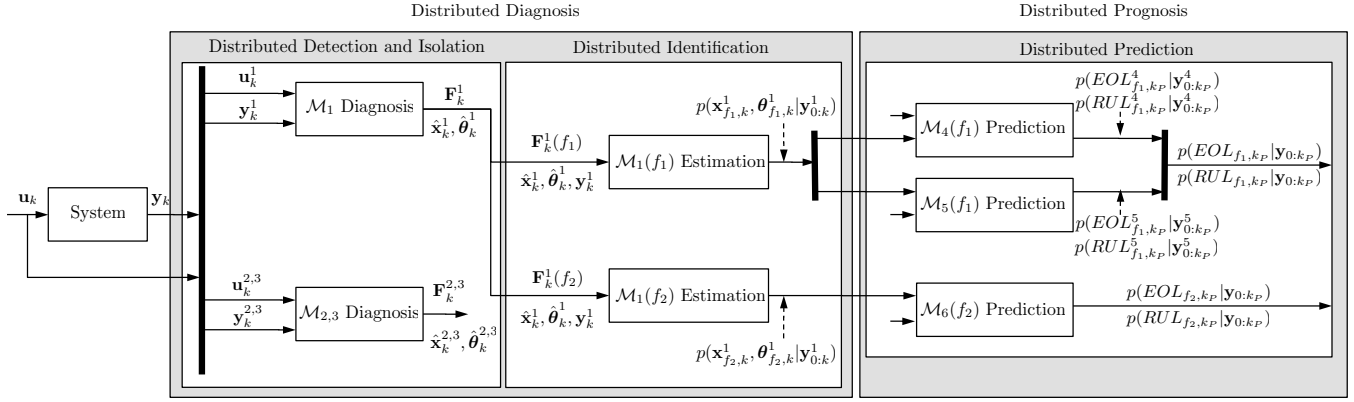


Figure 1. An instantiation of the integrated diagnosis and prognosis architecture.

$Y_i$  as a singleton. It will be shown later that the fault identification module is the central part of our diagnosis-prognosis integration approach and provides the joint state-parameter estimations for the prediction module.

Finally, for distributed prediction, the model decomposition problem starts off from the faulty system model, and, as detailed in (Daigle, Bregon, & Roychoudhury, 2012), it fulfills the following constraints: (i) the submodels use  $U_i \subseteq U_P$ , where  $U_P \subseteq X \cup U$  (here,  $U_P$  is a set of variables whose future values can be predicted *a priori*, which depends on the hypothesized faults); and (ii) each computed submodel has at least one  $c \in C_{EOL_f}$  belonging to  $C_i$ , and over all submodels, all constraints in  $C_{EOL_f}$  are covered. This ensures that  $T_{EOL_f}$  may be computed for the system from the local constraints.

### 3.2. Distributed Architecture

Figure 1 illustrates an example architecture for our distributed diagnosis and prognosis scheme. At each discrete time step,  $k$ , the system takes as input both  $\mathbf{u}_k$  and  $\mathbf{y}_k$  and splits them into local inputs  $\mathbf{u}_k^i$  and local outputs  $\mathbf{y}_k^i$  for the local diagnosers. Within each  $\mathcal{M}_i$  local diagnoser, nominal tracking is performed, computing estimates of nominal measurements,  $\hat{\mathbf{y}}_k^i$ . The fault detector compares the estimated measurements against the observed measurements, to determine statistically significant deviations for the residual,  $\mathbf{r}_k^i = \mathbf{y}_k^i - \hat{\mathbf{y}}_k^i$ . Qualitative values of the deviations in the residuals are used by the event-based diagnoser to isolate faults. The set of isolated fault candidates  $\mathbf{F}_k^i$  together with the estimated nominal states,  $\hat{\mathbf{x}}_k^i$ , parameters,  $\hat{\boldsymbol{\theta}}_k^i$ , and the measurements,  $\mathbf{y}_k^i$ , are used as input for the corresponding identification module. Identification is performed for each hypothesized fault in a distributed way, e.g., for the isolated faults  $f_1$  and  $f_2$  in Figure 1, we run an instantiation of the identification submodel for each one the faults, i.e.,  $\mathcal{M}_1(f_1)$  and  $\mathcal{M}_1(f_2)$ . Fault identification uses the minimal submodels from the faulty system model, and computes local state-parameter es-

timates  $p(\mathbf{x}_{f,k}^i, \boldsymbol{\theta}_{f,k}^i | \mathbf{y}_{0:k}^i)$ . These local estimates are then used as input to the prediction submodels. In some cases, the local estimates have to be split or merged with other estimates according to the prediction submodels. For example, in the figure, estimates from  $\mathcal{M}_1(f_1)$ , are used by both local prediction submodels  $\mathcal{M}_4(f_1)$  and  $\mathcal{M}_5(f_1)$ , and those submodels may also need estimates not included within submodel  $\mathcal{M}_1(f_1)$ . These estimates are typically obtained from the local diagnosers or other fault identification blocks.<sup>4</sup> Distributed prediction modules compute, for each hypothesized fault, local EOL/RUL predictions,  $p(EOL_{f,k_P}^i | \mathbf{y}_{0:k_P}^i)$  and  $p(RUL_{f,k_P}^i | \mathbf{y}_{0:k_P}^i)$ , at given prediction time  $k_P$  based on the local EOL constraints. Finally, local predictions are combined into global predictions  $p(EOL_{f,k_P} | \mathbf{y}_{0:k_P})$  and  $p(RUL_{f,k_P} | \mathbf{y}_{0:k_P})$  for each hypothesized fault. The next section describes the details of the different modules of the distributed integrated diagnosis and prognosis architecture.

## 4. DIAGNOSIS AND PROGNOSIS APPROACH

Figure 1 shows the basic modules of our distributed integrated approach. In this section we give details on how each module is implemented, and establish the integration between the diagnosis and prognosis tasks.

### 4.1. Distributed Diagnosis

For distributed diagnosis, each local diagnoser first takes a subset of the local inputs  $\mathbf{u}_k^i$  and local outputs  $\mathbf{y}_k^i$ , to compute an estimate of its output measurements  $\hat{\mathbf{y}}_k^i$ . Tracking is performed in discrete time using a robust filtering scheme, e.g., the extended or unscented Kalman filter (Julier & Uhlmann, 2004), which provides accurate tracking in the presence of sensor noise, process noise, and discretization error.

<sup>4</sup>Since prediction submodels are constructed by using any variable which value can be hypothesized as input, in some cases, prediction submodels cannot always be formed by just merging the minimal estimation submodels. To indicate this, we named our prediction submodels differently from the estimation submodels, e.g.,  $\mathcal{M}_4(f_1)$  and  $\mathcal{M}_5(f_1)$  instead of  $\mathcal{M}_1(f_1)$ .



For fault detection, a statistical test is used to look for significant deviations in the residual signal  $\mathbf{r}_k^i$ , which is computed as the difference between  $\hat{\mathbf{y}}_k^i$  and the  $\mathbf{y}_k^i$ . In our approach, we use a  $Z$ -test as described in (Biswas et al., 2003).

Fault isolation is performed using local event-based diagnosers, constructed as detailed in the previous section (Bregon et al., 2011; Daigle et al., 2009). Fault isolation is triggered when a fault is detected, and it works as follows. Initially, all event-based local diagnosers start in their initial state, and the set of faulty candidates is empty. Local residual deviations cause the local diagnosers to move from one state to another. These residual deviations are abstracted to a tuple of qualitative symbols  $(\sigma_1, \sigma_2)$  for each residual signal, where  $\sigma_1$  represents magnitude changes and  $\sigma_2$  represents slope changes. A + (resp. -) value indicates a change above (resp. below) normal for a measurement residual or a positive (resp. negative) residual slope. A 0 implies no change in the measurement value or a flat residual slope. The symbols are generated using a sliding window technique as described in detail in (Biswas et al., 2003). If there is a match between an event from the current state and a tuple of qualitative symbols generated by any residual, the local diagnoser moves to the next state and remains active. If not, the local diagnoser blocks. This process continues until a local diagnoser reaches an accepting state, which corresponds to a unique isolation result.

In our distributed diagnosis approach, identification submodels,  $\mathcal{M}_i(f)$ , are obtained, as explained in the previous section, as minimal submodels from the faulty system model. A local state-parameter estimator is constructed for each identification submodel  $\mathcal{M}_i(f)$ , and produces a local estimate  $p(\mathbf{x}_{f,k}^i, \boldsymbol{\theta}_{f,k}^i | \mathbf{y}_{0:k}^i)$  by using an appropriate algorithm. In this paper, we use an unscented Kalman filter (UKF) (Julier & Uhlmann, 2004) with a variance control algorithm (Daigle, Saha, & Goebel, 2012).

## 4.2. Distributed Prognosis

The local state-parameter estimates for each local distributed prediction module are constructed from the local estimates of the distributed fault identification submodels. Each prediction submodel is made up of a set of states  $X_i$  and parameters  $\Theta_i$ , and constructs a local distribution  $p(\mathbf{x}_{f,k}^i, \boldsymbol{\theta}_{f,k}^i | \mathbf{y}_{0:k}^i)$ , by assuming that the local state-parameter estimates are sufficiently represented by a mean  $\boldsymbol{\mu}^i$  and covariance  $\boldsymbol{\Sigma}^i$ . For each prediction submodel  $\mathcal{M}_i(f)$ , we combine the estimates of the local identification submodels that estimate states and parameters in  $X_i \cup \Theta_i$  into  $\boldsymbol{\mu}^i$  and  $\boldsymbol{\Sigma}^i$ . If two submodels estimate the same state variable or parameter, then many different techniques can be applied depending on the desired performance of the prediction submodels, e.g., taking the estimate with the smallest variance, or taking an average.

Several approaches can be used to perform prediction for each

---

### Algorithm 1 EOL Prediction

---

**Inputs:**  $\{(\mathbf{x}_{k_P}^{i(j)}, \boldsymbol{\theta}_{k_P}^{i(j)}), w_{k_P}^{i(j)}\}_{j=1}^N$   
**Outputs:**  $\{EOL_{k_P}^{i(j)}, w_{k_P}^{i(j)}\}_{j=1}^N$   
**for**  $j = 1$  **to**  $N$  **do**  
      $k \leftarrow k_P$   
      $\mathbf{x}_k^{i(j)} \leftarrow \mathbf{x}_{k_P}^{i(j)}$   
      $\boldsymbol{\theta}_k^{i(j)} \leftarrow \boldsymbol{\theta}_{k_P}^{i(j)}$   
     **while**  $T_{EOL}^i(\mathbf{x}_k^{i(j)}, \boldsymbol{\theta}_k^{i(j)}, \hat{\mathbf{u}}_k^i) = 0$  **do**  
         Predict  $\hat{\mathbf{u}}_k^i$   
          $\boldsymbol{\theta}_{k+1}^{i(j)} \sim p(\boldsymbol{\theta}_{k+1}^i | \boldsymbol{\theta}_k^{i(j)})$   
          $\mathbf{x}_{k+1}^{i(j)} \sim p(\mathbf{x}_{k+1}^i | \mathbf{x}_k^{i(j)}, \boldsymbol{\theta}_k^{i(j)}, \hat{\mathbf{u}}_k^i)$   
          $k \leftarrow k + 1$   
          $\mathbf{x}_k^{i(j)} \leftarrow \mathbf{x}_{k+1}^{i(j)}$   
          $\boldsymbol{\theta}_k^{i(j)} \leftarrow \boldsymbol{\theta}_{k+1}^{i(j)}$   
     **end while**  
      $EOL_{k_P}^{i(j)} \leftarrow k$   
**end for**

---

prediction submodel. In this work, given the mean and covariance information, we represent the distribution with a set of sigma points derived using the unscented transform. Then, each sigma point is simulated forward to EOL, and we recover the statistics of the EOL distribution given by the sigma points (Daigle & Goebel, 2010).

Algorithm 1 (Daigle, Saha, & Goebel, 2012), shows the pseudocode for the prediction procedure. The algorithm is executed for each submodel  $i$ , deriving local EOL predictions using its local threshold function based on the local EOL constraints. For a given submodel, each sample  $j$  is propagated forward until  $T_{EOL_f}^i(\mathbf{x}_{f,k}^i, \boldsymbol{\theta}_{f,k}^i)$  evaluates to 1. The algorithm hypothesizes future inputs  $\hat{\mathbf{u}}_k^i$ . Then, the global EOL/RUL is determined by the minimum of the local EOL/RUL distributions for each prediction submodel, i.e.,  $p(EOL_{f,k_P}^i | \mathbf{y}_{0:k_P}^i)$  and  $p(RUL_{f,k_P}^i | \mathbf{y}_{0:k_P}^i)$ . To compute this, we sample from each local EOL distribution and take the minimum of the local samples. This is repeated many times and the statistics of the global EOL distribution are computed (Daigle, Bregon, & Roychoudhury, 2012).

## 5. CASE STUDY

In this section, we apply our distributed diagnosis and prognosis approach to a four-wheeled rover testbed developed at NASA Ames Research Center. We develop a model of the rover, and demonstrate the approach using simulated scenarios.

### 5.1. Nominal System Modeling

The rover model was originally presented in (Balaban et al., 2011). In this section we summarize the main features and include some extensions to the model.

The rover consists of a symmetric rigid frame with four

independently-driven wheels. The wheel speeds are governed by

$$\dot{\omega}_{FL} = \frac{1}{J_{FL}} (\tau_{mFL} - \tau_{fFL} - \tau_{glFL} + \tau_{grFL}), \quad (c1)$$

$$\dot{\omega}_{FR} = \frac{1}{J_{FR}} (\tau_{mFR} - \tau_{fFR} - \tau_{glFR} - \tau_{grFR}), \quad (c2)$$

$$\dot{\omega}_{BL} = \frac{1}{J_{BL}} (\tau_{mBL} - \tau_{fBL} - \tau_{glBL} + \tau_{grBL}), \quad (c3)$$

$$\dot{\omega}_{BR} = \frac{1}{J_{BR}} (\tau_{mBR} - \tau_{fBR} - \tau_{glBR} - \tau_{grBR}). \quad (c4)$$

The  $F$ ,  $B$ ,  $L$ , and  $R$  subscripts stand for *front*, *left*, *back*, and *right*, respectively. Here, for wheel  $w$ ,  $J_w$  denotes the wheel inertia;  $\tau_{mw}$  is the motor torque;  $\tau_{fw} = \mu_{fw}\omega_w$  is the wheel friction torque, where  $\mu_{fw}$  is a friction coefficient;  $\tau_{glw} = r_w\mu_{gl}(v_w - v)$  is the torque due to slippage, where  $r_w$  is the wheel radius,  $\mu_{gl}$  is a friction coefficient,  $v_w$  is the translational wheel velocity, and  $v$  is the translation velocity of the rover body; and  $\tau_{grw} = r_w\mu_{grw}\omega \cos \gamma$  is the torque due to the rotational movement of the rover body, where  $\mu_{grw}$  is a friction coefficient,  $\omega$  is the rotational velocity of the rover body, and  $\gamma = \arctan l/b$  with  $l$  being the rover's length and  $b$  being its width.

The translational velocity  $v$  of the rover is described by

$$\dot{v} = \frac{1}{m} (F_{glFL} + F_{glFR} + F_{glBL} + F_{glBR}), \quad (c5)$$

where  $m$  is the rover mass, and for wheel  $w$ ,  $F_{glw} = \mu_{gl}(v_w - v)$  is the force due to slippage. The rotational velocity  $\omega$  is described by

$$\begin{aligned} \dot{\omega} = \frac{1}{J} (&d \cos \gamma F_{glFR} + d \cos \gamma F_{glBR} - d \cos \gamma F_{glFL} \\ &- d \cos \gamma F_{glBL} - dF_{grFL} - dF_{grFR} - dF_{grBL} \\ &- dF_{grBR}). \end{aligned} \quad (c6)$$

Here,  $J$  is the rotational inertia of the rover and  $d$  is the distance from the center of the rover to each wheel.

The wheels are driven by DC motors with PI control that sets the voltages  $V$  applied to the motors. The motor currents  $i$  are governed by

$$\dot{i}_{FL} = \frac{1}{L} (V_{FL} - i_{FL}R_{FL} - k_{\omega}\omega_{FL}), \quad (c7)$$

$$\dot{i}_{FR} = \frac{1}{L} (V_{FR} - i_{FR}R_{FR} - k_{\omega}\omega_{FR}), \quad (c8)$$

$$\dot{i}_{BL} = \frac{1}{L} (V_{BL} - i_{BL}R_{BL} - k_{\omega}\omega_{BL}), \quad (c9)$$

$$\dot{i}_{BR} = \frac{1}{L} (V_{BR} - i_{BR}R_{BR} - k_{\omega}\omega_{BR}), \quad (c10)$$

where  $L$  is the motor inductance,  $R$  is the motor resistance, and  $k_{\omega}$  is an energy transformation term. The motor torque is  $\tau_{mw} = k_{\tau}i_w$ , where  $k_{\tau}$  is an energy transformation gain. The voltages applied to the motors are determined by the controllers, where for wheel  $w$ ,  $V_w = P * (u_w - \omega_w) + I * e_{iw}$ , where  $P$  is a proportional gain,  $u_w$  is the commanded wheel speed,  $I$  is an integral gain, and  $e_{iw}$  is the integral error term.

The integral error terms are governed by

$$\dot{e}_{iFL} = u_{FL} - \omega_{FL}, \quad (c11)$$

$$\dot{e}_{iFR} = u_{FR} - \omega_{FR}, \quad (c12)$$

$$\dot{e}_{iBL} = u_{BL} - \omega_{BL}, \quad (c13)$$

$$\dot{e}_{iBR} = u_{BR} - \omega_{BR}. \quad (c14)$$

The batteries, which are connected in series, are described by a simple electrical circuit equivalent model that includes a large capacitance  $C_b$  in parallel with a resistance  $R_p$ , together in series with another resistance  $R_s$ .<sup>5</sup> The battery charge variables  $q_i$  are governed by

$$\dot{q}_1 = -V_1/R_{p1} - (i_{FL} + i_{FR} + i_{BR} + i_{BL}), \quad (c15)$$

$$\dot{q}_2 = -V_2/R_{p2} - (i_{FL} + i_{FR} + i_{BR} + i_{BL}), \quad (c16)$$

$$\dot{q}_3 = -V_3/R_{p3} - (i_{FL} + i_{FR} + i_{BR} + i_{BL}), \quad (c17)$$

$$\dot{q}_4 = -V_4/R_{p4} - (i_{FL} + i_{FR} + i_{BR} + i_{BL}). \quad (c18)$$

The available sensors measure the voltages of the batteries,

$$V_1^* = q_1/C_{b1} - R_{s1} * (i_{FL} + i_{FR} + i_{BR} + i_{BL}), \quad (c19)$$

$$V_2^* = q_2/C_{b2} - R_{s2} * (i_{FL} + i_{FR} + i_{BR} + i_{BL}), \quad (c20)$$

$$V_3^* = q_3/C_{b3} - R_{s3} * (i_{FL} + i_{FR} + i_{BR} + i_{BL}), \quad (c21)$$

$$V_4^* = q_4/C_{b4} - R_{s4} * (i_{FL} + i_{FR} + i_{BR} + i_{BL}), \quad (c22)$$

the motor currents,

$$i_{FL}^* = i_{FL}, \quad (c23)$$

$$i_{FR}^* = i_{FR}, \quad (c24)$$

$$i_{BL}^* = i_{BL}, \quad (c25)$$

$$i_{BR}^* = i_{BR}, \quad (c26)$$

and the wheel speeds,

$$\omega_{FL}^* = \omega_{FL}, \quad (c27)$$

$$\omega_{FR}^* = \omega_{FR}, \quad (c28)$$

$$\omega_{BL}^* = \omega_{BL}, \quad (c29)$$

$$\omega_{BR}^* = \omega_{BR}. \quad (c30)$$

Here, the \* superscript indicates a measured value.

## 5.2. Faulty System Modeling

In this work, we consider different faults in the motors and the batteries. First, we consider friction-based damage progression in the motors, resulting in an increase in motor friction over time. For wheel  $w$ , the fault progression function is defined as:

$$\dot{\mu}_{fFL} = \nu_{fFL} \mu_{fFL} \omega_{FL}^2, \quad (c31)$$

$$\dot{\mu}_{fFR} = \nu_{fFR} \mu_{fFR} \omega_{FR}^2, \quad (c32)$$

$$\dot{\mu}_{fBL} = \nu_{fBL} \mu_{fBL} \omega_{BL}^2, \quad (c33)$$

$$\dot{\mu}_{fBR} = \nu_{fBR} \mu_{fBR} \omega_{BR}^2, \quad (c34)$$

<sup>5</sup>We use a simple model here only for demonstration purposes. More detailed battery models for prognosis can be found in the literature, e.g., (Saha & Goebel, 2009).

Submodel	$X_i$	$\Theta_i$	$U_i$	$Y_i$	$C_i$
$\mathcal{M}_1$	$q_1$	$C_{b1}$	$i_{FL}^*, i_{FR}^*, i_{BL}^*, i_{BR}^*$	$V_1^*$	$C_{15}, C_{19}, C_{23}, C_{24}, C_{25}, C_{26}$
$\mathcal{M}_2$	$q_2$	$C_{b2}$	$i_{FL}^*, i_{FR}^*, i_{BL}^*, i_{BR}^*$	$V_2^*$	$C_{16}, C_{20}, C_{23}, C_{24}, C_{25}, C_{26}$
$\mathcal{M}_3$	$q_3$	$C_{b3}$	$i_{FL}^*, i_{FR}^*, i_{BL}^*, i_{BR}^*$	$V_3^*$	$C_{17}, C_{21}, C_{23}, C_{24}, C_{25}, C_{26}$
$\mathcal{M}_4$	$q_4$	$C_{b4}$	$i_{FL}^*, i_{FR}^*, i_{BL}^*, i_{BR}^*$	$V_4^*$	$C_{18}, C_{22}, C_{23}, C_{24}, C_{25}, C_{26}$
$\mathcal{M}_5$	$i_{FL}, e_{iFL}$	$R_{FL}$	$u_{FL}, \omega_{FL}^*$	$i_{FL}^*$	$C_7, C_{11}, C_{23}, C_{27}$
$\mathcal{M}_6$	$i_{FR}, e_{iFR}$	$R_{FR}$	$u_{FR}, \omega_{FR}^*$	$i_{FR}^*$	$C_8, C_{12}, C_{24}, C_{28}$
$\mathcal{M}_7$	$i_{BL}, e_{iBL}$	$R_{BL}$	$u_{BL}, \omega_{BL}^*$	$i_{BL}^*$	$C_9, C_{13}, C_{25}, C_{29}$
$\mathcal{M}_8$	$i_{BR}, e_{iBR}$	$R_{BR}$	$u_{BR}, \omega_{BR}^*$	$i_{BR}^*$	$C_{10}, C_{14}, C_{26}, C_{30}$
$\mathcal{M}_9$	$\omega_{FL}, v, \omega, \mu_{fFL}$	$\nu_{fFL}$	$i_{FL}^*, \omega_{FR}^*, \omega_{BL}^*, \omega_{BR}^*$	$\omega_{FL}^*$	$C_1, C_{31}, C_5, C_6, C_{23}, C_{28}, C_{29}, C_{30}$
$\mathcal{M}_{10}$	$\omega_{FR}, v, \omega, \mu_{fFR}$	$\nu_{fFR}$	$i_{FR}^*, \omega_{FL}^*, \omega_{BL}^*, \omega_{BR}^*$	$\omega_{FR}^*$	$C_2, C_{32}, C_5, C_6, C_{24}, C_{27}, C_{29}, C_{30}$
$\mathcal{M}_{11}$	$\omega_{BL}, v, \omega, \mu_{fBL}$	$\nu_{fBL}$	$i_{BL}^*, \omega_{FL}^*, \omega_{FR}^*, \omega_{BR}^*$	$\omega_{BL}^*$	$C_3, C_{33}, C_5, C_6, C_{25}, C_{27}, C_{28}, C_{30}$
$\mathcal{M}_{12}$	$\omega_{BR}, v, \omega, \mu_{fBR}$	$\nu_{fBR}$	$i_{BR}^*, \omega_{FL}^*, \omega_{FR}^*, \omega_{BL}^*$	$\omega_{BR}^*$	$C_4, C_{34}, C_5, C_6, C_{26}, C_{27}, C_{28}, C_{29}$

Table 1. Set of minimal submodels for the rover testbed computed from the nominal system model.

Submodel	$X_i$	$\Theta_i$	$U_i$	$Y_i$	$C_i$
$\mathcal{M}_{5,9}$	$\omega_{FL}, v, \omega, \mu_{fFL}, i_{FL}, e_{iFL}$	$\nu_{fFL}, R_{FL}$	$u_{FL}, \omega_{FR}^*, \omega_{BL}^*, \omega_{BR}^*$	$\omega_{FL}^*, i_{FL}^*$	$C_5 \cup C_9$
$\mathcal{M}_{6,10}$	$\omega_{FR}, v, \omega, \mu_{fFR}, i_{FR}, e_{iFR}$	$\nu_{fFR}, R_{FR}$	$u_{FR}, \omega_{FL}^*, \omega_{BL}^*, \omega_{BR}^*$	$\omega_{FR}^*, i_{FR}^*$	$C_6 \cup C_{10}$
$\mathcal{M}_{7,11}$	$\omega_{BL}, v, \omega, \mu_{fBL}, i_{BL}, e_{iBL}$	$\nu_{fBL}, R_{BL}$	$u_{BL}, \omega_{FL}^*, \omega_{FR}^*, \omega_{BR}^*$	$\omega_{BL}^*, i_{BL}^*$	$C_7 \cup C_{11}$
$\mathcal{M}_{8,12}$	$\omega_{BR}, v, \omega, \mu_{fBR}, i_{BR}, e_{iBR}$	$\nu_{fBR}, R_{BR}$	$u_{BR}, \omega_{FL}^*, \omega_{FR}^*, \omega_{BL}^*$	$\omega_{BR}^*, i_{BR}^*$	$C_8 \cup C_{12}$
$\mathcal{M}_{1,2,3,4}$	$q_1, q_2, q_3, q_4$	$C_{b1}, C_{b2}, C_{b3}, C_{b4}$	$i_{FL}^*, i_{FR}^*, i_{BL}^*, i_{BR}^*$	$V_1^*, V_2^*, V_3^*, V_4^*$	$C_1 \cup C_2 \cup C_3 \cup C_4$

Table 2. Residual generation and fault isolation submodels.

where  $\mu_{fw}$  is the fault parameter, and  $\nu_{fw}$  is the fault progression parameter.

We also consider abrupt resistance increases in the motors, represented as an abrupt change in parameter  $R_w$  for wheel  $w$ , with  $\Delta R_w$  as the fault progression parameter.

For the batteries, we consider abrupt capacitance decreases, represented as an abrupt change in parameter  $C_{bi}$  for capacity  $i$ .  $\Delta C_{bi}$  is the fault progression parameter.

We are interested in predicting when any of the rover batteries are at their charge threshold, beyond which the batteries will be damaged. These faults can cause the charge thresholds to be reached earlier since they will affect current draw. The constraints are given as

$$q_1 > q^-, \quad (c_{35})$$

$$q_2 > q^-, \quad (c_{36})$$

$$q_3 > q^-, \quad (c_{37})$$

$$q_4 > q^-, \quad (c_{38})$$

where the charge threshold is given by  $q^- = 2 \times 10^4$  C. The rover cannot be operated when any of the constraints  $c_{35}$ – $c_{38}$  are violated.

### 5.3. Results

To demonstrate the validity of the approach, we describe two different faulty scenarios of the rover. In the first, friction damage is progressing on one motor, and in the second, a capacitance decrease occurs in one battery. In all cases, the rover travels between various waypoints, moving at an average speed of 0.5 m/s. Table 1 shows the minimal submodels for the rover derived by using measured values as local inputs. Table 2 shows the submodels for residual generation

Submodel	$X_i$	$\Theta_i$	$U_i$	$Y_i$	$C_i$
$\mathcal{M}_{17}(C_{b1})$	$q_1, C_{b1}$	$\Delta C_{b1}$	$i_{FL}, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_{15}, C_{19}, C_{35}$
$\mathcal{M}_{18}(C_{b2})$	$q_2, C_{b2}$	$\Delta C_{b2}$	$i_{FL}, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_{16}, C_{20}, C_{36}$
$\mathcal{M}_{19}(C_{b3})$	$q_3, C_{b3}$	$\Delta C_{b3}$	$i_{FL}, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_{17}, C_{21}, C_{37}$
$\mathcal{M}_{20}(C_{b4})$	$q_4, C_{b4}$	$\Delta C_{b4}$	$i_{FL}, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$C_{18}, C_{22}, C_{38}$

Table 4. Prediction submodels for capacitance faults.

and fault isolation. These submodels have been designed to obtain globally diagnosable subsystems by using the design algorithm in (Bregon et al., 2011). In this work, we have considered five subsystems, one for each wheel components and another one for the batteries. For example, the subsystem for the front left wheel components is not globally diagnosable if we only consider submodel  $\mathcal{M}_9$  (which includes the front left wheel friction wear parameter,  $\nu_{fFL}$ ). The design algorithm in (Bregon et al., 2011) determines that we need to merge submodels  $\mathcal{M}_5$  and  $\mathcal{M}_9$  to make the front left wheel subsystem globally diagnosable. The process is similar for the rest of the subsystems.

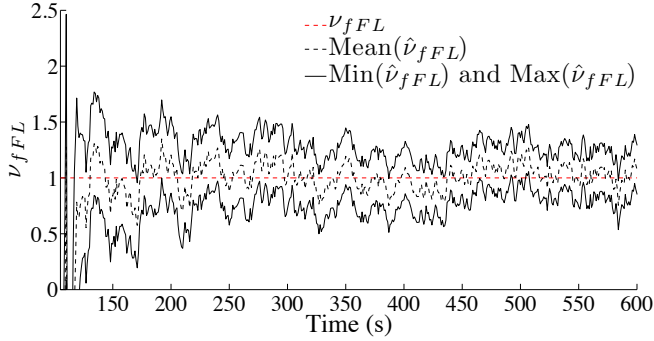
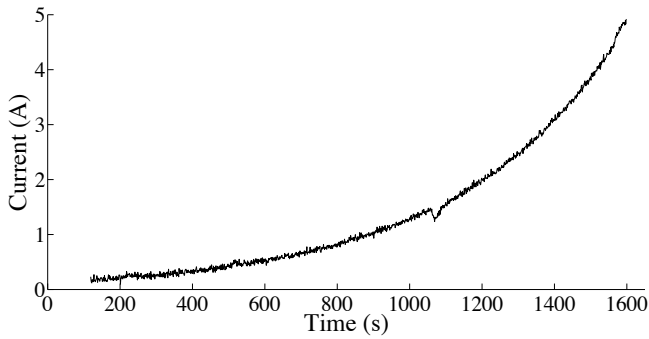
Minimal submodels for identification can be computed from the minimal submodels in Table 1 by defining the fault progression function (if necessary), and by making the fault parameter to become a state and the corresponding fault progression parameter to become the parameter. Regarding prediction, the correct prediction submodels to use depend on the scenario, as will be shown later.

#### 5.3.1. Friction Damage Progression

We first consider a scenario in which, for the front-left motor, the friction begins to increase. The friction damage progression begins at  $t = 50$  s with friction wear parameter  $\nu_{fFL} = 1 \times 10^{-3}$  s. A fault is detected by the local di-

Submodel	$X_i$	$\Theta_i$	$U_i$	$Y_i$	$C_i$
$\mathcal{M}_{13}(\mu_{fFL})$	$q_1, i_{FL}, e_{iFL}, \omega_{FL}, \mu_{fFL}$	$C_{b1}, \nu_{fFL}, R_{FL}$	$u_{FL}, v, \omega, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$c_1, c_{31}, c_7, c_{11}, c_{15}, c_{35}$
$\mathcal{M}_{14}(\mu_{fFL})$	$q_2, i_{FL}, e_{iFL}, \omega_{FL}, \mu_{fFL}$	$C_{b2}, \nu_{fFL}, R_{FL}$	$u_{FL}, v, \omega, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$c_1, c_{31}, c_7, c_{11}, c_{16}, c_{36}$
$\mathcal{M}_{15}(\mu_{fFL})$	$q_3, i_{FL}, e_{iFL}, \omega_{FL}, \mu_{fFL}$	$C_{b3}, \nu_{fFL}, R_{FL}$	$u_{FL}, v, \omega, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$c_1, c_{31}, c_7, c_{11}, c_{17}, c_{37}$
$\mathcal{M}_{16}(\mu_{fFL})$	$q_4, i_{FL}, e_{iFL}, \omega_{FL}, \mu_{fFL}$	$C_{b4}, \nu_{fFL}, R_{FL}$	$u_{FL}, v, \omega, i_{FR}, i_{BL}, i_{BR}$	$\emptyset$	$c_1, c_{31}, c_7, c_{11}, c_{18}, c_{38}$

Table 3. Prediction submodels using commanded wheel speeds and rover velocities as local inputs.


 Figure 2. Estimated  $\nu_{fFL}$  values.

 Figure 3. Current  $i_{FL}^*$  increase through time.

agnoser computed from submodel  $\mathcal{M}_{5,9}$  at 119.25 s, via an increase in the motor current  $i_{FL}$ . The initial candidate list is immediately reduced to one candidate,  $\{\nu_{fFL}\}$ , based on the signatures and orderings (other faults in the front left wheel, like  $R_{FL}$ , produce different fault signatures). Thus the true fault is isolated.

Fault identification is initiated once the candidate is isolated. For the friction damage progression fault, the wear rate  $\nu_{fFL}$  estimate averages to  $\nu_{fFL} = 1 \times 10^{-3}$  s with very small output error. Figure 2 shows the wear parameter estimate for friction damage.

As a result of the continuously increasing friction, the current drawn by the motor increases as well in order for the motor controller to maintain the same desired wheel speed (Figure 3 shows this increase in the current through time). Hence, the total current drawn from the batteries is increased, and EOL occurs around half an hour. Because  $i_{FL}$  is constantly changing, and in a way that is dependent on the motor state, it is

incorrect to use it as a local input for prediction and to decompose the prediction problem into independent local prediction problems for the batteries and motors, i.e., it is not known a priori. Therefore, we compute submodels using as local inputs average values for the remaining motor currents, average commanded wheel speeds, and average rover translational velocity  $v$  and rotational velocity  $\omega$ . The prediction submodels for this case are shown in Table 3. EOL for this fault is computed by merging the local EOL from those submodels in the table. Note that the prediction submodels used in this case do not correspond directly to those used for estimation. So, when constructing the estimate for  $\mathcal{M}_{13}$ , for example, it takes the estimates from  $\mathcal{M}_1$  and  $\mathcal{M}_9$ .

The prediction results are shown in Figure 4. The increased friction causes the batteries to discharge faster, and EOL occurs around 1650 s. Here, we used the relative accuracy (RA) as a measure of prediction accuracy, and the relative standard deviation (RSD) as a measure of spread. Each prediction metric is averaged over multiple prediction points (one every 100 s of usage) (see (Saxena et al., 2010; Daigle, Saha, & Goebel, 2012) for the mathematical definitions of these metrics). For this experiment, RA averages to 91.63% and RSD averages to 16.26%.

For the sake of comparison, we also ran this experiment using the centralized approach. Figure 5 shows the prediction results obtained. Looking at the prediction metrics, we see that the centralized approach behaved very similar to the distributed approach but a little bit worse, with RA averaging 90.90% and RSD averaging 17.72%. However, this is just a particular example, but, in general, both approaches obtain equivalent results.

### 5.3.2. Capacitance Decrease

As a second scenario, we consider a capacitance decrease fault in battery 3 of the rover,  $C_{b3}$ . The fault begins at  $t = 50$  s with an abrupt decrease from 2000 to 1800 in the capacity of the battery. The fault is detected immediately by the local diagnoser computed from submodel  $\mathcal{M}_{1,2,3,4}$  at 50.0 s, via an increase in the voltage  $V_3$ . The fault candidate is immediately isolated,  $\{C_{b3}\}$ , based on the signatures and orderings, thus starting the fault identification. For the capacitance fault, the estimated value of the capacitance averaged  $C_{b3} = 1798.6$  C with very small output error. As a result of the decrease in capacitance, the battery discharges at a faster rate, and so reaches end of discharge more quickly. The prediction sub-

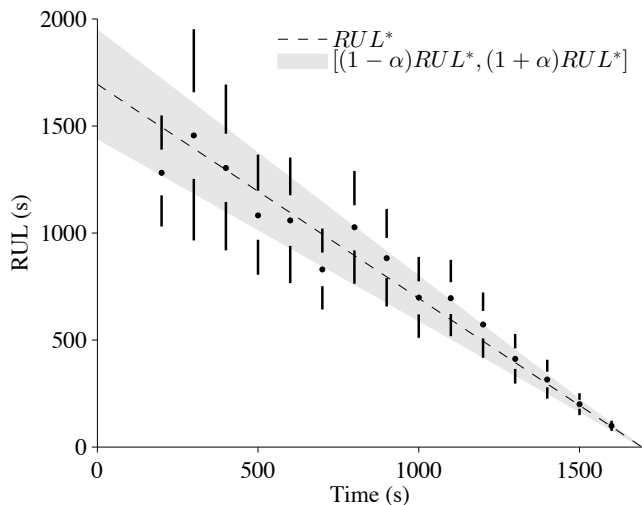


Figure 4. Predicted RUL of the rover for the distributed approach. The mean is indicated with a dot and confidence intervals for 5% and 95% by lines. The gray cone depicts an accuracy requirement of 15%.

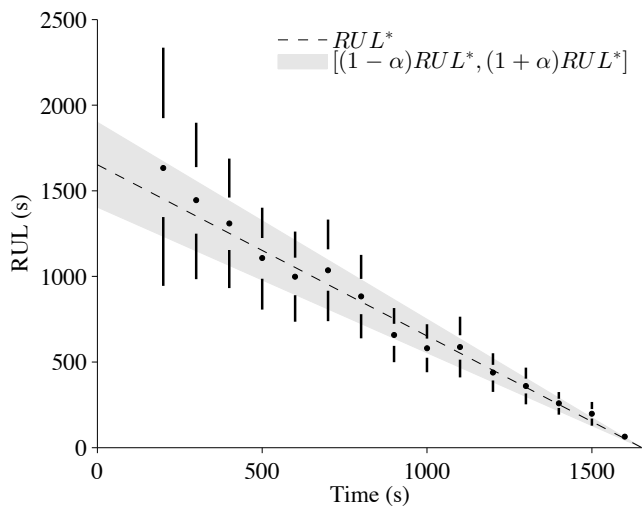


Figure 5. Predicted RUL of the rover for the centralized approach. The mean is indicated with a dot and confidence intervals for 5% and 95% by lines. The gray cone depicts an accuracy requirement of 15%.

models for faults in the capacity of the batteries are shown in Table 4. For this scenario, with a fault in  $C_{b3}$ , we used submodel  $\mathcal{M}_{19}$ , obtaining RA average to 98.25% and RSD average to 10.12%.

## 6. CONCLUSIONS

This paper presented a distributed integrated model-based diagnosis and prognosis framework. Our approach starts off with a common modeling paradigm to model both the nominal behavior and fault progression, and then proposes a framework where the global system model is decomposed into smaller independent submodels. These submodels are then used to distribute the different diagnosis and prognosis tasks. Model decomposition is carried out based on the requirements and constraints of each task. We demonstrated our approach on a four-wheeled rover testbed, where we diagnosed faults and prognosed the EOL/RUL accurately. We compared results obtained by using our distributed approach against those obtained using a centralized approach, showing that both approaches obtain the same results.

Most approaches in the literature focus in either the diagnosis or the prognosis task. Some works have proposed the integration of both tasks within a common framework (Patrick et al., 2007; Orchard & Vachtsevanos, 2009; Roychoudhury & Daigle, 2011), however, unlike our approach, these approaches perform the diagnosis and prognosis tasks in a centralized way, thus suffering from scalability issues due to the large number of states and parameters in real-world systems. To the best of our knowledge, there is no approach in the literature which combines, in a distributed way, the diagnosis and prognosis tasks. Our approach is limited by the number and location of the sensors in the system. Since our decomposition algorithm is guided by the set of available sensors, the distribution capabilities of the approach is determined by them.

In future, we will apply this approach to larger systems, to study the scalability of our diagnosis and prognosis scheme, and will perform a more detailed comparison against the results obtained by using a centralized approach. We will also extend the capability of this approach to hybrid systems, as well as diagnosis and prognosis of multiple faults.

## ACKNOWLEDGMENTS

A. Bregon's funding for this work was provided by the Spanish MCI TIN2009-11326 grant. M. Daigle and I. Roychoudhury's funding for this work was provided by the NASA System-wide Safety and Assurance Technologies (SSAT) Project.

## REFERENCES

- Balaban, E., Narasimhan, S., Daigle, M., Celaya, J., Roychoudhury, I., Saha, B., et al. (2011, September). A Mobile Robot Testbed for Prognostics-Enabled Autonomous Decision Making. In *Annual Conference of the Prognostics and Health Management Society* (p. 15-30). Montreal, Canada.
- Biswas, G., Simon, G., Mahadevan, N., Narasimhan, S., Ramirez, J., & Karsai, G. (2003, June). A robust method for hybrid diagnosis of complex systems. In *Proceedings of the 5th Symposium on Fault Detection, Supervision and Safety for Technical Processes* (pp. 1125-1131).
- Blanke, M., Kinnaert, M., Lunze, J., & Staroswiecki, M. (2006). *Diagnosis and Fault-Tolerant Control*. Springer.
- Bregon, A., Biswas, G., & Pulido, B. (2012). A Decomposition Method for Nonlinear Parameter Estimation in TRANSCEND. *IEEE Trans. Syst. Man. Cy. Part A*, 42(3), 751-763.
- Bregon, A., Daigle, M., Roychoudhury, I., Biswas, G., Koutsoukos, X., & Pulido, B. (2011, Oct). Improving Distributed Diagnosis Through Structural Model Decomposition. In *Proceedings of the 22nd International Workshop on Principles of Diagnosis* (p. 195-202). Murnau, Germany.
- Daigle, M., Bregon, A., & Roychoudhury, I. (2011, September). Distributed Damage Estimation for Prognostics Based on Structural Model Decomposition. In *Proceedings of the Annual Conference of the Prognostics and Health Management Society 2011* (p. 198-208).
- Daigle, M., Bregon, A., & Roychoudhury, I. (2012). *Distributed prognostics based on structural model decomposition*. (Manuscript submitted for publication)
- Daigle, M., & Goebel, K. (2010, October). Improving Computational Efficiency of Prediction in Model-based Prognostics Using the Unscented Transform. In *Annual Conf. of the Prognostics and Health Management Society 2010*.
- Daigle, M., Koutsoukos, X., & Biswas, G. (2009, July). A Qualitative Event-based Approach to Continuous Systems Diagnosis. *IEEE Trans. on Control Systems Technology*, 17(4), 780-793.
- Daigle, M., Saha, B., & Goebel, K. (2012, March). A comparison of filter-based approaches for model-based prognostics. In *Proceedings of the 2012 IEEE Aerospace Conference*.
- Gertler, J. J. (1998). *Fault Detection and Diagnosis in Engineering Systems*. New York, NY: Marcel Dekker, Inc.
- Julier, S. J., & Uhlmann, J. K. (2004, March). Unscented filtering and nonlinear estimation. *Proc. of the IEEE*, 92(3), 401-422.
- Luo, J., Pattipati, K. R., Qiao, L., & Chigusa, S. (2008, September). Model-based prognostic techniques applied to a suspension system. *IEEE Trans. on Systems, Man and Cybernetics, Part A: Systems and Humans*, 38(5), 1156-1168.
- Mosterman, P. J., & Biswas, G. (1999, November). Diagnosis of Continuous Valued Systems in Transient Operating Regions. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Trans. on*, 29(6), 554-565.
- Orchard, M. E., & Vachtsevanos, G. (2009). A Particle-filtering Approach for On-line Fault Diagnosis and Failure Prognosis. *Trans. of the Institute of Measurement and Control*, 31(3/4), 221-246.
- Patrick, R., Orchard, M. E., Zhang, B., Koelemay, M., Kacprzyński, G., Ferri, A., et al. (2007, September). An Integrated Approach to Helicopter Planetary Gear Fault Diagnosis and Failure Prognosis. In *Proc. of the 42nd Annual Systems Readiness Technology Conf.* Baltimore, MD, USA.
- Pulido, B., & Alonso-González, C. (2004). Possible Conflicts: a compilation technique for consistency-based diagnosis. *IEEE Trans. on Systems, Man, and Cybernetics, Part B, Special Issue on Diagnosis of Complex Systems*, 34(5), 2192-2206.
- Roychoudhury, I. (2009). *Distributed Diagnosis of Continuous Systems: Global Diagnosis Through Local Analysis*. Unpublished doctoral dissertation, Vanderbilt University.
- Roychoudhury, I., & Daigle, M. (2011, October). An integrated model-based diagnostic and prognostic framework. In *Proceedings of the 22nd International Workshop on Principles of Diagnosis* (p. 44-51).
- Saha, B., & Goebel, K. (2009, September). Modeling Li-ion battery capacity depletion in a particle filtering framework. In *Proc. of the Annual Conf. of the Prognostics and Health Management Society 2009*.
- Saxena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2010). Metrics for offline evaluation of prognostic performance. *Int. Journal of Prognostics and Health Management*.
- Staroswiecki, M., & Declerck, P. (1989, July). Analytical redundancy in nonlinear interconnected systems by means of structural analysis. In *IFAC Symp. on Advanced Information Processing in Automatic Control*.
- Williams, B., & Millar, B. (1998). Decompositional Model-based Learning and its analogy to diagnosis. In *Proc. of the Fifteenth National Conf. on Artificial Intelligence, AAAI'98* (p. 197-204).

## BIOGRAPHIES

**Anibal Bregon** received his B.S., M.S., and Ph.D. degrees in Computer Science from the University of Valladolid, Spain, in 2005, 2007, and 2010, respectively. From September 2005 to June 2010, he was Graduate Research Assistant

with the Intelligent Systems Group at the University of Valladolid, Spain. He has been visiting researcher at the Institute for Software Integrated Systems, Vanderbilt University, Nashville, TN, USA; the Dept. of Electrical Engineering, Linkoping University, Linkoping, Sweden; and the Diagnostics and Prognostics Group, NASA Ames Research Center, Mountain View, CA, USA. Since September 2010, he has been Assistant Professor and Research Scientist at the Department of Computer Science from the University of Valladolid.

Dr. Bregon is a member of the Prognostics and Health Management Society and the IEEE. His current research interests include model-based reasoning for diagnosis, prognostics, health-management, and distributed diagnosis of complex physical systems.

**Matthew Daigle** received the B.S. degree in Computer Science and Computer and Systems Engineering from Rensselaer Polytechnic Institute, Troy, NY, in 2004, and the M.S. and Ph.D. degrees in Computer Science from Vanderbilt University, Nashville, TN, in 2006 and 2008, respectively.

From September 2004 to May 2008, he was a Graduate Research Assistant with the Institute for Software Integrated Systems and Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN. During the summers of 2006 and 2007, he was an intern with Mission Critical Technologies, Inc., at NASA Ames Research Center. From June 2008 to December 2011, he was an Associate Sci-

entist with the University of California, Santa Cruz, at NASA Ames Research Center. Since January 2012, he has been with NASA Ames Research Center as a Research Computer Scientist. His current research interests include physics-based modeling, model-based diagnosis and prognosis, simulation, and hybrid systems.

Dr. Daigle is a member of the Prognostics and Health Management Society and the IEEE. He is a recipient of a University Graduate Fellowship from Vanderbilt University, a best paper award in the Annual Conference of the Prognostics and Health Management Society 2011, a NASA Ames Group Achievement Award in 2011, and an Ames Contractor Council Excellence Award in 2011. He has published over 40 peer-reviewed papers in the area of Systems Health Management.

**Indranil Roychoudhury** received the B.E. (Hons.) degree in Electrical and Electronics Engineering from Birla Institute of Technology and Science, Pilani, Rajasthan, India in 2004, and the M.S. and Ph.D. degrees in Computer Science from Vanderbilt University, Nashville, Tennessee, USA, in 2006 and 2009, respectively. Since August 2009, he has been with SGT, Inc., at NASA Ames Research Center as a Computer Scientist.

Dr. Roychoudhury is a member of the Prognostics and Health Management Society and the IEEE. His research interests include hybrid systems modeling, model-based diagnostics and prognostics, distributed diagnostics and prognostics, and Bayesian diagnostics of complex physical systems.

# Uncertainty Representation and Interpretation in Model-based Prognostics Algorithms based on Kalman Filter Estimation

José R. Celaya<sup>1</sup>, Abhinav Saxena<sup>2</sup>, and Kai Goebel<sup>3</sup>

<sup>1,2</sup> *SGT Inc. NASA Ames Research Center, Moffett Field, CA, 94035, USA*

*jose.r.celaya@nasa.gov*  
*abhinav.saxena@nasa.gov*

<sup>3</sup> *NASA Ames Research Center, Moffett Field, CA, 94035, USA*

*kai.goebel@nasa.gov*

## ABSTRACT

This article discusses several aspects of uncertainty representation and management for model-based prognostics methodologies based on our experience with Kalman Filters when applied to prognostics for electronics components. In particular, it explores the implications of modeling remaining useful life prediction as a stochastic process and how it relates to uncertainty representation, management, and the role of prognostics in decision-making. A distinction between the interpretations of estimated remaining useful life probability density function and the true remaining useful life probability density function is explained and a cautionary argument is provided against mixing interpretations for the two while considering prognostics in making critical decisions.

## 1. INTRODUCTION

Model-based prognostics methodologies in electronics prognostics have been developed based on Bayesian tracking methods such as Kalman Filter, Extended Kalman Filter, and Particle Filter. The models used in these methodologies are mathematical abstractions of the time evolution of the degradation process and the cornerstone for the estimation of remaining useful life. The Bayesian tracking framework allows for estimation of state of health parameters in prognostics making use of available measurements from the system under consideration. In this framework, health parameters are regarded as random variables for which, in the case of Kalman and Extended Kalman filters, their distribution are regarded as Normal and the estimation process focuses on computing estimates of the expected value and variance as they relate to the mean and variance that fully parametrize the Normal distribution. In addition to the health estimation process, fore-

casting of the health parameters is required up to a future time that results in crossing of the pre-established failure condition threshold. This is ultimately required in order to compute remaining useful life.

Previous work applied to electrolytic capacitor and power MOSFETs (Metal-Oxide Semiconductor Field-Effect Transistor) has focused on implementation of the previously described process and has presented remaining useful life results without any uncertainty measure associated to them (J. R. Celaya et al., 2011; J. Celaya, Saxena, Kulkarni, et al., 2012; J. Celaya et al., 2011; J. Celaya, Kulkarni, et al., 2012). Other work on prognostics based on particle filtering has been presented regarding remaining useful life as a random variable and presenting corresponding uncertainty estimates (Saha et al., 2009; Daigle & Goebel, 2011). This work focuses on reviewing uncertainty representation techniques used in model-based prognostics and on providing an interpretation of uncertainty for the electronics prognostics applications previously presented, and based on Kalman filter approaches for health state estimation.

The Bayesian tracking framework allows for modeling of sources of uncertainty in the measurement process and also on the degradation evolution dynamic model as applied on the application under consideration. This is done in terms of an additive noise in the model, which is regarded as zero mean and normally distributed random variable. This allows for the aggregation of different sources of uncertainty for the health state tracking step. Its implications on the uncertainty estimation for remaining useful life (RUL) including future state forecasting are discussed in this paper.

### 1.1. Model-based prognostics background

As mentioned earlier, a model-based prognostics methodology based on Bayesian tracking consists of two steps, health state estimation and RUL prediction. The following is a high

José R. Celaya et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.



level description of the process that will help to provide the appropriate context for the upcoming discussion.

**State of health estimation:** To initiate the prediction, it is necessary to first establish a starting point, the current state of health. A model-based algorithm employs dynamic models of the physical behavior of the system or component under consideration, along with dynamic degradation models of key parameters that represent the degradation over time. Bayesian tracking algorithms like Kalman filter, extended Kalman filter, and particle filter are among the algorithms typically employed in a model-based prognostics methodology (Daigle & Goebel, 2011; Saha & Goebel, 2009; J. Celaya et al., 2011; J. Celaya, Saxena, Kulkarni, et al., 2012). In such methodologies, dynamic models of the nominal system and degradation models are posed as a discrete state-space system in which the state variable  $\mathbf{x}(t)$  consists of physical variables, and in some cases, it includes degradation model parameters to be estimated online.

The models consist of a state equation representing the time evolution of the state as shown in Eq. (1a); where  $u(t)$  is the system input and  $w(t)$  is a zero-mean and normally distributed additive noise representing random model error. In addition, the measurement equation (Eq. (1b)) relates the state variable to measurements of the systems  $y(t)$ . The term  $v(t)$  is a zero-mean and normally distributed additive noise representing the random measurement error. The measurement and model noise normality assumption could be relaxed when using computational Bayesian methods like particle filtering.

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), u(t)) + w(t) \quad (1a)$$

$$y(t) = h(\mathbf{x}(t), u(t)) + v(t) \quad (1b)$$

The state of the system, as it evolves through time, is periodically estimated by the filter as measurements  $y(t)$  of key variables become available through the life of the system. This is the health state estimation step of the model-based prognostics algorithm. Typically, in a model-based prognostics method, a Bayesian tracking algorithm attempts to estimate the expected value of the joint probability density function of the state  $\mathbf{x}(t_p)$ . Where  $t_p$  is the time at which a remaining useful life prediction is computed using only system observations up to this point in time. Different assumptions about the probability density function are used depending on the filter used.

**Remaining useful life estimation (prediction):** In order to compute remaining useful life, the state-equation (Eq. (1a)) of the model is used to compute the state evolution in a forecasting mode until an end-of-life threshold is reached at time denoted by  $t_{EOL}$ . The last state estimate at time  $t_p$  in the

health state estimation step is typically used as initial state value for forecasting  $\mathbf{x}(t)$  up to  $t_{EOL}$ . Remaining useful life  $R(t_p)$  at time of prediction  $t_p$  is defined as

$$R(t_p) = t_{EOL} - t_p, \quad (2)$$

where  $t_p$  is deterministic and known, and  $t_{EOL}$  is a random variable function of the failure threshold and the state estimate  $\mathbf{x}(t_p)$ . This function includes the state forecasting step and the identification of when the failure threshold is crossed.

## 1.2. Ideas explored in this paper

In this paper we explore how the state vector variable should be interpreted during the tracking phase and how it is related to the process of final RUL prediction. This probability interpretation is often overlooked in the literature by interpreting the state vector as the health indicator and a threshold is used on this variable in order to compute EOL (end-of-life) and RUL.

Here, we discuss how the state estimation process is defined in the Bayesian framework. We will, in particular, focus on the output of the estimation process in the Kalman filter framework. Furthermore, we try to interpret the objective of the Kalman filter, whether to estimate  $\mathbf{x}(t)$  as a random variable or to estimate a parameter of the probability density function of  $\mathbf{x}(t)$ —such as expected value or variance— or both.

In addition, we will challenge how we usually think about RUL and how it has been interpreted using other, similar, methods. The main objective here is to characterize its impact on uncertainty representation and management. For instance, if RUL is considered as a random variable and we assume that a model-based prognostics framework based on the Kalman filter generates RUL with a particular variance, then it is incorrect to arbitrarily expect, assume, or force the variance to be small. The variance of random variables such as RUL is not under our control as explained in the next section.

These concepts are discussed in the context of prognostics of electronics, particularly, the uncertainty propagation in power MOSFET and capacitor prognostics applications as presented in J. R. Celaya et al. (2011); J. Celaya, Saxena, Kulkarni, et al. (2012) and J. Celaya et al. (2011); J. Celaya, Kulkarni, et al. (2012) respectively. In these applications, uncertainty has not been explicitly considered in the prediction results and this paper is an effort towards augmenting the methods used there with an uncertainty management methodology.

## 1.3. Background on Uncertainty Management

There are several different types of sources of uncertainty that must be accounted for in a prognostic system formulation. These sources may be categorized into following four categories and accordingly require separate representation and management methods.

1. **Aleatoric or Statistical Uncertainties:** these uncertainties arise from inherent variability in any process and cannot be eliminated. They can be characterized by multiple experimental runs but cannot be reduced by improved methods or measurements. Sampling fluctuations from the characterized probability density function of a source of aleatoric uncertainty can result in different predictions every time. Examples of such uncertainties include manufacturing variations, material properties, etc.
2. **Epistemic or Systematic Uncertainties:** these uncertainties arise due to unknown details that cannot be identified and hence are not incorporated into a process. With improved methods and deeper investigations these uncertainties may be reduced but are rarely eliminated. Modeling uncertainties fall under this category and include modeling errors due to unmodeled phenomena in both system model and the fault propagation model.
3. **Prejudicial Uncertainties:** these uncertainties arise due to the way a process is set up and is expected to change if the process is redesigned. Conceptually these can be considered a type of epistemic uncertainty, except it is possible to control these to a better extent. Examples for these uncertainties include sensor noise, sensor coverage, information loss due to data processing, various approximations and simplifications, numerical errors, etc.

While it is possible to reduce some of these uncertainties, it is not possible or practically beneficial to eliminate them altogether. However, representing them and accounting for them in prognostic outputs is extremely important. Uncertainties in a prognostic estimate directly affect the associated decision making process, which is typically expressed through the concept of risk due to unwanted outcomes. Several PHM approaches quantify risk based on uncertainty quantification in an algorithm's output and incorporate it into a corresponding cost-benefit equation through monetary concepts (Bedford & Cooke, 2001).

### 1.3.1. Uncertainty management in prognostics

In the context of prognostics and health management uncertainties are talked about from quantification, representation, and management points of view (deNeufville, R., 2004; Hastings & McManus, 2004; Ng & Abramson, 1990; Orchard et al., 2008; Tang et al., 2009). While all three are different processes they are often confused with each other and interchangeably used.

**Uncertainty quantification:** Deals with characterizing a source of uncertainty so it can be incorporated into models and simulations as correctly as possible. A characterization or quantification step may involve carefully designed experimentation with actual systems observed in realistic and relevant environments. An accurate quantification of uncertainties is considered very challenging as also acknowledged

in Engel (2009). Quantification of uncertainty from various sources in a process has been investigated and a sensitivity analysis conducted to identify which input uncertainty contributes most to the output uncertainty in prognostics for fatigue crack damage (Sankararaman et al., 2011). This allows prioritizing and subsequently focusing on more critical uncertainties instead of all.

**Uncertainty representation:** Next step is the representation of uncertainty, which is, often times, guided by the choice of modeling and simulation frameworks. There are several methods for uncertainty representation that vary in the level of granularity and detail. Some common theories include classical set theory, probability theory, fuzzy set theory, fuzzy measure (plausibility and belief) theory, and rough set (upper and lower approximations) theory. However, in the PHM domain the representation of uncertainties is dominated by probabilistic measures (DeCastro, 2009; Orchard et al., 2008; Saha et al., 2009), which offer a mathematically rigorous approach but assume availability of a statistically sufficient database. Other approaches, such as possibility theory (Fuzzy logic) and Dempster-Shafer theory, can be employed when only scarce or incomplete data are available (Wang, 2011). Furthermore, the choice of type of probability density function affects the quality of prognostic outputs. Several approaches in the literature resorted to assuming Normal probability density functions, however this choice should be guided by the results of the uncertainty characterization and quantification step.

**Uncertainty management:** The most loosely used term in the PHM literature in the context of uncertainty is that of uncertainty management. Uncertainty management includes two main functions, to incorporate all relevant and/or significant sources of uncertainty into prognostic models and simulations. Therefore, the problem formulation stage itself lays a foundation for an effective uncertainty management. Once all relevant sources of uncertainty are identified and included, the uncertainty propagation is the next component towards effective management. Various measures of uncertainty must be combined in an appropriate manner in the prognostic model as the input variability filters through a complex (possibly non-linear) system model.

If, in a perfect situation, all sources of uncertainties are identified, modeled, and managed correctly, the output probability density function for random variables like RUL or End-of-life (EOL) would match the true spread and would not change from one experiment to another. This is, however, in practice impossible to achieve because no model is perfect and not all sources of uncertainties can be characterized. Furthermore, an exhaustive sampling-based method such as a Monte Carlo simulation would be computationally, prohibitively expensive. This has inspired the development of intelligent sampling based algorithms (DeCastro, 2009; Orchard et al., 2008;

Saha et al., 2009) and mathematical transformations, such as Support vectors (Saha & Goebel, 2008) and Principle Component analysis (Usynin & Hines, 2007), that result in minor approximations but capture most details of the true variability. It may not be possible to identify and accurately characterize all sources of uncertainty and hence use of a sensitivity analysis is recommended to isolate the most important factors (Gu et al., 2007; Sankararaman et al., 2011; Tang et al., 2009). Through effective uncertainty management practices one can at most strive towards bringing the predicted estimate close to the true spread and not arbitrarily reducing the spread of RUL itself. What can be minimized, is the variability in the estimate of a given parameter of interest, not the variability in the parameter of interest itself.

## 2. REMAINING USEFUL LIFE STOCHASTIC MODELING

Remaining useful life in a prognostics context is defined differently than in a reliability context. In prognostics, it is implied that remaining useful life at time  $t_p$  is a condition-based estimation of the usage time left until failure, using measurements of key variables and past usage information up to time  $t_p$ . This process typically consists of forecasting the future state of health beyond  $t_p$  and identifying when the state of health will cross a failure threshold representative of a functional failure. In addition, RUL in prognostics considers – implicitly or explicitly – future usage conditions. This is not the case in the reliability context. Given the many sources of uncertainty evident from a quick assessment of all that is involved in computing RUL for a system, it is common to consider RUL as a non-deterministic quantity. Furthermore, RUL is also a time evolving process, meaning that RUL at time  $t_p$  will be different than RUL for  $t \neq t_p$ . This can be well illustrated with the use of the alpha-lambda prognostics metric (Saxena et al., 2010) as seen in various publications on prognostics (J. R. Celaya et al., 2011; J. Celaya et al., 2011).

### 2.1. Remaining useful life as a stochastic process

A random process or stochastic process is defined as a collection of random variables. Following the definition presented in Gross and Harris (1998), a stochastic process is a “mathematic abstraction of an empirical process whose development is governed by probabilistic laws”. Furthermore, it is defined as a family of random variables  $\{X(t), t \in T\}$  where  $T$  is the time range and  $X(t)$  is the state of the process at time  $t$ . The time range could be discrete or continuous.

A stochastic process is also used in the signal-processing context to represent non-deterministic (stochastic) signals (Oppenheim & Schaffer, 1989). From Kalman (1960) we get the following explanation as it relates to filtering: “Intuitively, a random process is simply a set of random variables which are indexed in such a way as to bring the notion of time into the picture”.

In several applications, RUL prediction is a process in which periodic computations of RUL are generated through the life of the system under consideration. In our previous work on power MOSFET prognostics (J. R. Celaya et al., 2011), periodic measurements (up to every minute) are available. RUL is computed periodically and can be considered as a random process  $R(t)$ . In contrast, in our previous work on electrolytic capacitor prognostics (J. Celaya et al., 2011), measurements are not available at regular time intervals. RUL computations are made multiple times whenever a measurement is available. In this case,  $R(t)$  can also be considered as a random process but the set  $T$  will contain only the times at which RUL was computed.

### 2.2. Implications on uncertainty management

The definition of RUL as a random variable or random process has many implications on uncertainty management and in the representation of uncertainty in a particular model-based prognostics methodology. If RUL is not modeled within a probability framework, like a fuzzy variable or just a deterministic variable, uncertainty management activities will differ. To illustrate, let us consider a simple point estimate example from basic mathematical statistics (Bain & Engelhardt, 1992).

**A parameter estimation example:** Let us assume that we can perform a set of run to failure experiments with high level of control, ensuring same usage and operating conditions. In addition, remaining useful life at time  $t_p$  is computed by measuring the elapsed time from  $t_p$  until failure for all the  $n$  samples  $(R_1, \dots, R_n)$  on the set of run to failure experiments. Assuming that these random samples come from a probability density function  $f_R(r)$ , with expected value  $E(R) = \mu$  and variance  $Var(R) = \sigma^2$ .

Let  $\theta_1$  be a parameter estimator of the mean  $\mu$  of  $f_R$ , with expected value  $E(\theta_1) = \mu_{\theta_1}$  and variance  $V(\theta_1) = \sigma_{\theta_1}^2$ . This estimator will be a function of all the sample values and will have a probability density function  $f_{\theta_1}$ .  $\theta_1$  is a point estimate of the random variable  $R$  such as the sample mean, the median or some other location statistic. Now, from the uncertainty management perspective in prognostics, it is necessary to judge the ability of the algorithm to properly compute the point estimate of the process, in this case, to properly estimate  $\mu$ . So it is expected that this estimate  $\theta_1$  has the least variability, the least variance possible, therefore making  $\theta_1$  less uncertain. As a result,  $\sigma_{\theta_1}^2$  should be as small as possible. It is, on the other hand, incorrect to expect the estimation process to reduce  $\sigma^2$  itself.

This is often misinterpreted for prognostics methodologies base on computational statistics that do not directly focus on a point estimate but on generating an approximation of the distribution of  $R$ . Since the variability can be assessed by a measure of spread like the sample standard deviation com-

puted directly from the sample distribution of  $R$ , again, this variation should not be arbitrarily decreased by tuning of the algorithm since it is intended to represent the real statistical uncertainty of the process.

The previous discussion applies to RUL predictions without loss of generality as long as they are modeled as random variables, which is typically the case. The concept can be further described considering the sample average  $\bar{R}$  as the estimator ( $\theta_1 = \bar{R}$ ). From basic probability theory (Bain & Engelhardt, 1992), one can observe that  $\mu_{\theta_1} = \mu$  and the  $\sigma_{\theta_1}^2 = \sigma^2/n$ . This estimator is unbiased, and its variance  $\sigma_{\theta_1}^2$  can be reduced by increasing sample size. But  $\sigma^2$  cannot be reduced because it is the inherent variability in the random variable  $R$ .

### 2.3. Implications on how RUL is computed by statistical models

Let us now consider the complete RUL computation algorithm including state estimation and prediction steps, i.e., the prognostics algorithm is a black box estimation of RUL. This statistical model can have different focus in providing estimations of  $R(t)$ . The following situations (although not exhaustive) are considered here:

1.  $R(t)$  could be assumed to be a known random variable with a known probability density or mass function (parametric case). Therefore, the statistical model will focus on providing the best possible estimator of the parameters or key quantities function of the random variable as the expected value and the variance. For instance, if  $R(t)$  is presumed Normal, then the statistical model will provide an estimate of the mean and the standard deviation since they fully parametrize the Normal random variable.
2. A computational statistics model could be used to avoid making assumptions about the distribution of  $R(t)$  therefore focusing on computing an approximation of the probability density/mass function of  $R(t)$ . This will be a choice for the cases in which there is no knowledge about the distribution or the non-parametric case is preferred. It will also be the case for when there is no analytical solution tractable for the statistical model structure therefore the use of a computational model, based on Monte Carlo simulation approaches, is needed.

The uncertainty management focus will differ under the two situations described above. In case one, where distribution parameters are estimated, the uncertainty management should focus on properly estimating the spread parameter  $\theta_s$  of  $R(t)$ . A spread parameter  $\theta_s$  could be variance or some other estimator focused on representing the variability of the distribution. This estimator should properly aggregate all the previously identified sources of uncertainty, like measurement, model, future input and environment uncertainty. From the uncertainty management perspective, one should not expect  $\theta_s$  to be small. Instead, one should expect it to be an accu-

rate representation of the real uncertainty in the real RUL of the system. A similar situation arises in the second case. In this case an approximation of the distribution of  $R(t)$  is computed. Its shape and therefore the spread or variability represented by this approximation, should be the real uncertainty of the RUL in the system and should not be made arbitrarily small either by tuning the statistical method to do so or by any other arbitrary transformation to make this approximation more crisp around the location parameter.

### 2.4. Implications on decision-making

Being able to capture the uncertainty correctly is of paramount importance in prognostics. This might not always be the case for other applications involving parameter estimation. For instance, in a control application, the frequency of the compensation loop is generally high enough to be able to dampen the effects of uncertainty in the parameter estimation process. For prognostics, this will typically not be the case. If the prognostics situation under consideration is used for contingency management, in which safety of operation is at stake; properly estimating the uncertainty of the true RUL is necessary. If the uncertainty estimation is incorrect, then this can lead to risky decision-making, leading to reduced safety and possibly increasing the change of catastrophic failure. A similar argument can be made if prognostics is used in a logistics settings such as condition-based maintenance in manufacturing systems or in military operations.

The previous argument can also be made from the opposite end by considering the implications of the decision-making method on how RUL is computed and how uncertainty management is performed. For the last few years, research in prognostics and health management (PHM) has mainly focussed on the prognostics element, which deals with methods to predict RUL. There have been several methodologies published and many more under development for a variety of man-made systems. As a result of the previous effort, prognostics methodologies have been developed in a sort of unbounded or unguided way with respect to how the actual method is going to be used in the decision-making process. This means that input from the types of decisions that will use the prognostics information and from the overall optimization of system performance have so far not been considered.

The type of decision-making application may dictate the prognostics methodology as well as the types of estimates to be generated (recall cases in Section 2.2.3). Consequently, this will also have an impact on requirements generation. For instance a fleet based optimization of aircraft maintenance operations considers very different decisions as compared to an unmanned aerial vehicle (UAV) mission reconfiguration based on prognostics indication on power train failures. Following the same argument, it is clear that different decision-making methodologies will have different capabili-

ties in terms of handling the prognostics information. For instance, an optimization of a particular decision process might not be able to work with random variables, therefore a point estimate would be provided. This will be different if the optimization itself is able to deal with RUL as a random variable, in this case, the computation distribution function of  $R(t)$  or the estimators of the parameters that fully parametrize it would be provided. If the decision-making process, can further use information about how reliable the prognostics information is, then information about a measure of quality of the estimators, which is different than just bias, would be provided.

### 3. UNCERTAINTY INTERPRETATION AND COMPUTATION IN MODEL-BASED PROGNOSTICS WITH KALMAN FILTER ESTIMATION

Model-based prognostics methodologies for electronics components like electrolytic capacitors (J. Celaya et al., 2011; J. Celaya, Kulkarni, et al., 2012) and power MOSFETs (J. R. Celaya et al., 2011; J. Celaya, Saxena, Kulkarni, et al., 2012) have been previously introduced. The methodologies make use of empirical degradation models and a single precursor to failure parameter to compute RUL. These methodologies rely on accelerated aging experiments to identify degradation behavior and to create time dependent degradation models. The process followed in these methodologies is presented in the block diagram in Figure 1.

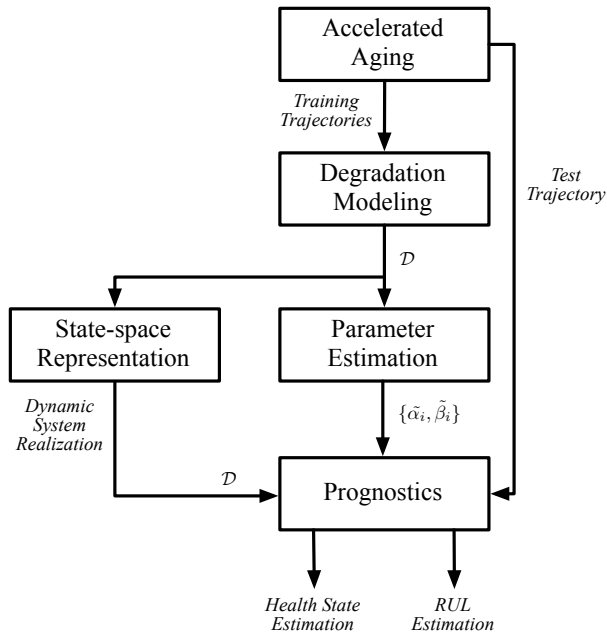


Figure 1. Methodology for electronics component prognostics development.

Accelerated aging tests provided measurements throughout the aging process, including measurements at pristine con-

dition and measurements after failure condition. Empirical degradation models that are based on the observed degradation process during the accelerated aging tests are developed. The objective of the models is to generate a parametrized model of the time-dependent degradation process for these components. The time dependent degradation model is transformed into a discrete-time linear dynamic system in order to be used in a Bayesian tracking setting. The Kalman filter algorithm is used to track the state of health and the degradation model is used to make predictions of remaining useful life once no further measurements are available.

#### 3.1. Prognostics methodology

The methodology consists of the three main steps described below and it is depicted in Figure 2. This is the explanation of what it is considered inside the prognostics block in Figure 1.

This methodology follows from the general concepts of model-based prognostics described in Section 1.1.1. In the electronics component case, the system dynamics consists only of the degradation process dynamics since the prognostics focuses at the component level only.

1. State tracking (Kalman Filter): The state variable in the degradation model  $\mathcal{D}$  is a precursor of failure parameter represented by Eq. (3a). When the degradation model uses static parameters (parameters not estimated online by the filter), then the state variable is a scalar quantity and the state evolution equation is scalar. The degradation model is expressed as a discrete time dynamic model in order to estimate the state as new measurements become available. The simplified Kalman filter model set up is given as

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}, \quad (3a)$$

$$y_k = Hx_k + v_k. \quad (3b)$$

The output of this step is the optimal state estimate  $\hat{x}_p$ .

2. Health state forecasting: It is necessary to forecast the state variable once there are no more measurements available at the time of RUL prediction  $t_p$ . This is done by evaluating the degradation model (Eq. (3a)) through time using the state estimate  $\hat{x}_p$  from the previous step as the initial state value for forecasting.
3. Remaining life computation: RUL is computed as the time between time of prediction  $t_p$  and the time at which the forecasted state crosses the failure threshold value  $F$ .

This process is repeated for different values of  $t_p$  through the life of the component under consideration.

#### 3.2. Kalman Filter Background

The Kalman filter framework is based on Bayesian parameter estimation. A Bayes estimator allows to estimate parameters

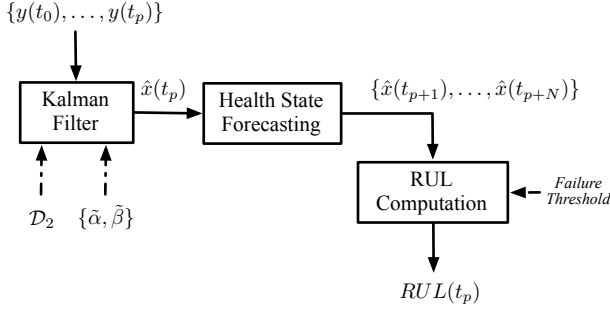


Figure 2. Model-based prognostics methodology

based on prior knowledge about the parameter distribution. In the tracking problem, system measurements serve as a form of prior knowledge, therefore the objective is to estimate the state  $x(t)$  conditional to all the previous measurements of the system. The Bayes estimation framework is based on the concepts of risk and loss functions in which the risk is defined as the expected loss (Bain & Engelhardt, 1992). This background information is relevant since it helps to understand the statistical origins of the Kalman filter framework which is the focal point of the discussions in this paper. Based on the seminal paper for the Kalman filter (Kalman, 1960), the optimal state estimate is given as  $x^*(t) = E[x(t)|y(t_0), \dots, y(t)]$ . This is the solution that minimizes the risk (expected loss), for a loss function based on the estimation error. Furthermore, the random process for the state and for the process noise are Normal. Additional details on the problem formulation and assumptions are presented in Kalman (1960).

**Implications on Kalman filter for prognostics:** Considering a scalar implementation of the Kalman filter over discrete-time model as in Eqs. (3). The output of the filter referred to as the optimal state estimate  $x_k^*$  is basically given by the conditional state estimate  $\hat{x}_k = E[x_k|y_k]$  and the state conditional probability density function is given by,

$$p(x_k|y_k) \sim N(\hat{x}_k, P_k), \quad (4)$$

where  $P_k$  is the filter's estimate of the error variance.

The output of the filter is the estimate of the expected value  $\hat{x}_k$ , and the estimation error covariance  $P_k$ . The state random variable  $x_p$  is normally distributed with mean  $\hat{x}_k$  and variance  $P_k$ .

### 3.3. Uncertainty propagation in prognostics

Based on the previous discussion regarding the interpretation of the Kalman filter output in terms of probabilities, it can be observed that the health state estimation output is a Normal random variable with known parameters considering the sources of uncertainties derived from modeling error and measurement error.

**Uncertainty in the health state estimation step:** We assume

here a scalar case for state estimation, like in the case of the capacitor prognostics method where the health indicator is a scalar state variable (J. Celaya et al., 2011). Time index  $p$  is defined as the time of RUL prediction  $t_p$ , which is also the time of the last available measurement in the state estimation step. The state estimate  $x_p$  is a normally distributed random variable with mean  $\hat{x}_p$  and variance  $P_k$ .

$$x_p \sim N(\hat{x}_p, P_k) \quad (5)$$

This variable includes the propagation of measurement uncertainty and also model error uncertainty as included in the Kalman filter implementation.

**Uncertainty in the health state forecasting step:** Forecasting is needed for the state variable to be able to estimate its value at a future time until it crosses a pre-established failure threshold  $F$ . The forecasting process is carried out using the state equation (Eq. (3a)) recursively, using the last health state estimate  $\hat{x}_p$  as initial value. Let  $\tilde{x}_p(l)$  be the  $l_{th}$  step ahead forecast starting from  $x_p$ . From the uncertainty propagation point of view and focusing on a one step ahead forecasting using Eq. (3a), the forecast value is given by

$$\tilde{x}_p(1) = Ax_p + Bu_p + w_p. \quad (6)$$

Variables  $x_p$  and  $w_p$  are Normal and independent with known mean and variance. Following from basic probability theory, the forecast  $\tilde{x}_p(1)$  is also Normal. In general, the  $l_{th}$  step ahead forecast  $\tilde{x}_p(l)$  will have a Normal distribution as well. It should also be noted that  $\tilde{x}_p(l)$  is a function of the last state estimate ( $\tilde{x}_p(l) = f(x_p)$ ). Considering the forecast variables as random variables and given the analytical properties of the Normal distribution, the probability density function  $f_{\tilde{x}_p(l)}$  can be derived analytically and is given by,

$$\tilde{x}_p(l) \sim N(\mu_l, \sigma_l^2); \quad (7)$$

where the mean is given by

$$\mu_l = A^l \hat{x}_p + Bu_p + \sum_{i=0}^{l-1} A^i, \quad (8)$$

and the variance is given by

$$\sigma_l^2 = (A^2)^l P_k + \sum_{i=1}^{l-1} (A^2)^i \sigma_w^2 + \sigma_w^2. \quad (9)$$

**Uncertainty in RUL:** Computing the uncertainty in the RUL is more complicated from an analytical point of view. Defining  $R(t_p)$  as the remaining useful life,

$$R(t_p) = t_{EOL} - t_p. \quad (10)$$

The time at end-of-life ( $t_{EOL}$ ) is a continuous variable which

is computed from the forecast  $\tilde{x}_p(l)$ .

Let  $\tilde{x}_p(j)$  be the first forecast value to cross the failure threshold  $F$ . An interpolation between  $\tilde{x}_p(j)$  and  $\tilde{x}_p(j-1)$  is used to compute  $t_{EOL}$ . Considering that the forecasts are functions of  $x_p$ , RUL is also a function of  $x_p$ .

$$R(t_p) = g(x_p). \quad (11)$$

From the random variable uncertainty propagation point of view,  $R(t_p)$  is a function  $g$  of a normally distributed random variable, therefore, it is also a random variable. It is nevertheless difficult to derive its probability density function analytically. There is also no information that suggests that  $R(t_p)$  will be Normal. The probability density function of  $R(t_p)$  can be approximated using computational statistics methods. This can be done by taking  $N$  samples from  $x_p$  and computing  $R(t_p)$  for each sample. An histogram can be built from the  $N$  computed  $R(t_p)$  values and a density estimation method could be used to generate the approximation of the probability density function.

### 3.4. Discussion

From the analytical results presented for the first two steps of the prognostics process (Section 3.3.3), it can be observed that the variance will be larger after the forecasting process. In addition, there is no evidence to suggest that  $R(t_p)$  will be Normal and further investigation is needed to explore its dependance on the forecasting process, like number of steps ahead forecasts and step length. It is also clear, that simply defining the variance of  $R(t_p)$  as  $P_k$  or  $\sigma_l^2$  is not an accurate representation of the uncertainty in the process.

The model-based methodology for electronics prognostics based on the Kalman filter is able to capture additive degradation model error uncertainty and additive measurement uncertainty. In order for the approximation of the probability density function of  $R(t_p)$  to be a true representation of the system uncertainty, the variances of the measurement noise and modeling noise should be properly estimated. If considered as tuning parameters, then the generated uncertainty in  $R(t_p)$  will not be representative of the real process.

### 4. CONCLUSION

This article presented a discussion on uncertainty representation and management for model-based prognostics methodologies based on the Bayesian tracking framework and specifically for a Kalman filter application to electronics components. In particular, it explores the implication of modeling remaining useful life prediction as a stochastic process and how it relates to remaining useful life computation by statistical models, to uncertainty representation and management, and to the role of prognostics in decision-making. A discussion on how uncertainty propagates from the health state

estimation process through the health state forecasting process is provided. Remaining useful life computation steps under uncertainty are presented and analytical results on uncertainty quantification are provided under a simplified scenario. A proper propagation of uncertainty through the RUL prediction step as well as its correct interpretation are key to developing decision-making methodologies that make use of the remaining useful life prediction estimates and their corresponding uncertainties in order to make actionable choices that will optimize reliability, operations or safety in view of the prognostics information.

*This work was originally presented in the 2012 AIAA Infotech@Aerospace Conference (J. Celaya, Saxena, & Goebel, 2012). It is reproduced here with minor updates and corrections based on suggestions by reviewers.*

### ACKNOWLEDGMENT

The work reported herein was in part funded by NASA ARMD/AvSafe project SSAT and by NASA/OCT/AES project ACLO. Authors would also like to acknowledge the members of Prognostics Center of Excellence at NASA Ames Research Center for engaging in valuable and insightful discussions.

### NOMENCLATURE

$R$	Remaining useful life random variable
$t_p$	Time of remaining useful life prediction
$R(t_p)$	Remaining useful life prediction at time $t_p$
$\hat{x}_k$	Optimal state estimator from Kalman filter
$\tilde{x}_k(l)$	$l_{th}$ step ahead forecast from $x_k$
$t_{EOL}$	Time at end-of-life
$x(t)$	Scalar continuous state variable for filter model
$\mathbf{x}(t)$	Vector continuous state variable for filter model
$x_k$	Scalar discrete-time state variable for filter model
$F$	Failure threshold
$N(\mu, \sigma^2)$	Normal distribution with mean $\mu$ and variance $\sigma^2$

### REFERENCES

- Bain, L. J., & Engelhardt, M. (1992). *Introduction to probability and mathematical statistics*. Duxbury.
- Bedford, T., & Cooke, R. M. (2001). *Probabilistic risk analysis: Foundations and methods*. Cambridge University Press.
- Celaya, J., Kulkarni, C., Biswas, G., & Goebel, K. (2011, September 25-29). A model-based prognostics methodology for electrolytic capacitors based on electrical overstress accelerated aging. In *Proceedings of annual conference of the phm society*. Montreal, Canada.
- Celaya, J., Kulkarni, C., Saha, S., Biswas, G., & Goebel, K. (2012, January). Accelerated aging in electrolytic ca-

- pacitors for prognostics. In *2012 proceedings - annual reliability and maintainability symposium (rams)* (p. 1 -6). doi: 10.1109/RAMS.2012.6175486
- Celaya, J., Saxena, A., & Goebel, K. (2012, June 19-21). A discussion on uncertainty representation and interpretation in model-based prognostics algorithms based on kalman filter estimation applied to prognostics of electronics components. In *Infotech@aerospace 2012 online conference proceedings*. Garden Grove, California.
- Celaya, J., Saxena, A., Kulkarni, C., Saha, S., & Goebel, K. (2012, January). Prognostics approach for power mosfet under thermal-stress aging. In *2012 proceedings - annual reliability and maintainability symposium (rams)* (p. 1 -6). doi: 10.1109/RAMS.2012.6175487
- Celaya, J. R., Saxena, A., Saha, S., & Goebel, K. (2011, September 25-29). Prognostics of power mosfets under thermal stress accelerated aging using data-driven and model-based methodologies. In *Proceedings of annual conference of the phm society*. Montreal, Canada.
- Daigle, M. J., & Goebel, K. (2011). A model-based prognostics approach applied to pneumatic valves. *International Journal of Prognostics and Health Management*, 2 (2)(008).
- DeCastro, J. A. (2009). Exact nonlinear filtering and prediction in process model-based prognostics. In *Annual conference of the prognostics and health management society*. San Diego, CA..
- deNeufville, R. (2004). Uncertainty management for engineering systems planning and design. In *Engineering systems symposium mit*. Cambridge, MA..
- Engel, S. J. (2009). PHM engineering perspectives, challenges and 'crossing the valley of death'. In *Annual conference of the prognostics and health management society*. San Diego, CA..
- Gross, D., & Harris, C. M. (1998). *Fundamentals of queueing theory* (3rd ed.). John Wiley & Sons Inc.
- Gu, J., Barker, D., & Pecht, M. (2007). Uncertainty assessment of prognostics of electronics subject to random vibration. In *Artificial intelligence for prognostics: Papers from the aaii fall symposium*.
- Hastings, D., & McManus, H. (2004). A framework for understanding uncertainty and its mitigation and exploitation in complex systems. In *Engineering systems symposium mit* (p. 19). Cambridge MA..
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME—Journal of Basic Engineering*, 82 (Series D), 35-45.
- Ng, K.-C., & Abramson, B. (1990). Uncertainty management in expert systems. *IEEE Expert Systems*, 20.
- Oppenheim, A. V., & Schaffer, R. W. (1989). *Discrete-time signal processing*. Prentice Hall.
- Orchard, M., Kacprzynski, G., Goebel, K., Saha, B., & Vachtsevanos, G. (2008, oct.). Advances in uncertainty representation and management for particle filtering applied to prognostics. In *Prognostics and health management, 2008. phm 2008. international conference on* (p. 1 -6). doi: 10.1109/PHM.2008.4711433
- Saha, B., & Goebel, K. (2008, march). Uncertainty management for diagnostics and prognostics of batteries using bayesian techniques. In *Aerospace conference, 2008 ieee* (p. 1 -8). doi: 10.1109/AERO.2008.4526631
- Saha, B., & Goebel, K. (2009). Modeling li-ion battery capacity depletion in a particle filtering framework. In *Proceedings of annual conference of the phm society*.
- Saha, B., Goebel, K., Poll, S., & Christophersen, J. (2009, feb.). Prognostics methods for battery health monitoring using a bayesian framework. *IEEE Transactions on Instrumentation and Measurement*, 58(2), 291 -296. doi: 10.1109/TIM.2008.2005965
- Sankararaman, S., Ling, Y., Shantz, C., & Mahadevan, S. (2011). Uncertainty quantification in fatigue crack growth prognosis. *International Journal of Prognostics and Health Management*, 2-1(1).
- Saxena, A., Celaya, J. R., Saha, B., Saha, S., & Goebel, K. (2010). Metrics for offline evaluation of prognostic performance. *International Journal of Prognostics and Health Management*, 1 (1)(001).
- Tang, L., Kacprzynski, G., Goebel, K., & Vachtsevanos, G. (2009, march). Methodologies for uncertainty management in prognostics. In *Aerospace conference, 2009 ieee* (p. 1 -12). doi: 10.1109/AERO.2009.4839668
- Usynin, A., & Hines, J. W. (2007). Uncertainty management in shock models applied to prognostic problems. In *Artificial intelligence for prognostics: Papers from the aaii fall symposium*.
- Wang, H.-f. (2011, January). Decision of prognostics and health management under uncertainty. *International Journal of Computer Applications*, 13(4), 1–5. (Published by Foundation of Computer Science)

## BIOGRAPHIES

**José R. Celaya** is a research scientist with SGT Inc. at the Prognostics Center of Excellence, NASA Ames Research Center. He received a Ph.D. degree in Decision Sciences and Engineering Systems in 2008, a M. E. degree in Operations Research and Statistics in 2008, a M. S. degree in Electrical Engineering in 2003, all from Rensselaer Polytechnic Institute, Troy New York; and a B. S. in Cybernetics Engineering in 2001 from CETYS University, Meéxico.

**Abhinav Saxena** is a Research Scientist with Stinger Ghafarian Technologies at the Prognostics Center of Excellence NASA Ames Research Center, Moffet Field CA. His research focus lies in developing and evaluating prognostic algorithms for engineering systems using soft computing techniques. He is a PhD in Electrical and Computer Engineering from Georgia Institute of Technology, Atlanta. He earned his B. Tech



in 2001 from Indian Institute of Technology (IIT) Delhi, and Masters Degree in 2003 from Georgia Tech. Abhinav has been a GM manufacturing scholar and is also a member of IEEE, AAAI and ASME.

**Kai Goebel** received the degree of Diplom-Ingenieur from the Technische University Munchen, Germany in 1990. He received the M.S. and Ph.D. from the University of California at Berkeley in 1993 and 1996, respectively. Dr. Goebel is a senior scientist at NASA Ames Research Center where he is the deputy area lead for Discovery and Systems Health in the Intelligent Systems division. In addition, he directs

the Prognostics Center of Excellence and he is the Associate Principal Investigator for Prognostics of NASAs Integrated Vehicle Health Management Program. He worked at General Electrics Corporate Research Center in Niskayuna, NY from 1997 to 2006 as a senior research scientist. He has carried out applied research in the areas of artificial intelligence, soft computing, and information fusion. His research interest lies in advancing these techniques for real time monitoring, diagnostics, and prognostics. He holds fifteen patents and has published more than 200 papers in the area of systems health management.

# Prognostic-Enabling of an Electrohydrostatic Actuator (EHA) System

Sonia Vohnout<sup>1</sup>, David Bodden<sup>2</sup>, Byoung Uk Kim<sup>3</sup>, Robert Wagoner<sup>4</sup>, Neil Kunst<sup>5</sup>, Patrick Edwards<sup>6</sup>, Bill Gleeson<sup>7</sup>, Dennis Cascio<sup>8</sup>, Steve Brzuszkiewicz<sup>9</sup>, Roy Wagemans<sup>10</sup>, Matthew Rounds<sup>11</sup>, and N. Scott Clements<sup>12</sup>

<sup>1,3,4,5,6,7,11</sup>*Ridgetop Group, Inc., Tucson, AZ, 85741, USA*

*svohnout@ridgetopgroup.com*  
*tkim@ridgetopgroup.com*  
*rwagoner@ridgetopgroup.com*  
*nkunst@ridgetopgroup.com*  
*pedwards@ridgetopgroup.com*  
*bgleeson@ridgetopgroup.com*  
*mrounds@ridgetopgroup.com*

<sup>2,11</sup>*Lockheed Martin Aerospace, Fort Worth, TX, 78744, USA*

*david.s.bodden@lmco.com*  
*scott.clements@lmco.com*

<sup>7,8</sup>*Moog, Inc., East Aurora, NY, 14052, USA*

*dcascio@moog.com*  
*sbrzuszkiewicz@moog.com*

<sup>9</sup>*Dell Services (Netherlands), 1014 AK Amsterdam, Netherlands*

*Roy\_Wagemans@Dell.com*

## ABSTRACT

A proof-of-concept prognostic solution for certain failure modes in the power electronics that drive the flight-critical F-35 Joint Strike Fighter (JSF) electrohydrostatic actuators (EHA) is presented. This program was led by Ridgetop Group under U.S. NAVAIR Small Business Innovation Research (SBIR) funding, and included Lockheed Martin Aeronautics Company (LM), Moog, and Dell Services (Netherlands). Degradation of the optocoupler that isolates the control electronics from the power electronics was simulated in the lab by physically changing resistance values to alter the current transfer ratio. It is proposed that this degradation would also be indicative of insulated gate bipolar transistor (IGBT) wearout. The experimental approach, the test facility, the data analysis and the findings are discussed. An Off-Board Prognostics Health Management (OBPHM) Demonstrator, developed by Ridgetop Group and Dell Systems and representative of the production OBPHM application currently deployed for the F-35 is described. Implementation considerations and challenges are also discussed.

Sonia Vohnout et al. Copyright 2012 Lockheed Martin, Ridgetop Group, Moog, Dell Services. Reprinted with Permission. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

## 1. INTRODUCTION

The F-35 electrohydrostatic actuators (EHA) were designed to be maintenance-free, i.e., there were not going to be scheduled maintenance activities for the life of the aircraft. The only repair activities would be associated with identification of faults via built-in test (BIT) or integrated cautions and warnings (ICAW) from system monitors which provide coverage for all critical failure modes. Prognostics have been proposed for some failure modes of the EHA, but not the power electronics unit (EU), which provides the power and control to its associated EHA.

Predicting the future state of health (SOH) of critical components in the EHA system (EHAS) could possibly prevent some loss of mission availability due to unforeseen failures. The objective of this program was to identify the feasibility of assessing the SOH of certain critical components in the EHA power electronics via a novel approach to pre-flight BIT. This approach utilizes frequency-shaped actuator commands during BIT to assess SOH. The program was led by Ridgetop Group under U.S. NAVAIR Small Business Innovation Research (SBIR) funding, and included Lockheed Martin Aeronautics Company (LM), Moog, and Dell Services (Netherlands). Ridgetop Group specializes in electronic prognostic solutions for critical systems. These include sensor array

detectors, harnesses for “prognostic-enabling” critical systems, and analysis software to comprise a complete solution.

As the prime contractor for the F-35 Program, LM’s operational experience with the EHA system and flight control system BIT design was instrumental in defining the data requirements, test plan definition and execution, and assessing the feasibility of integrating diagnostic/prognostic capabilities into the F-35 EHA and the Autonomic Logistics Information System (ALIS).

Moog leads the F-35 EHA subcontractor team which has responsibility for development of the actuators for the primary and secondary flight control surfaces (Figure 1). Moog’s design expertise with the EHA power electronics was utilized to help identify the candidate electronic components that could be artificially degraded during the experiments, and they also provided the test labs and test personnel to conduct the experiments.

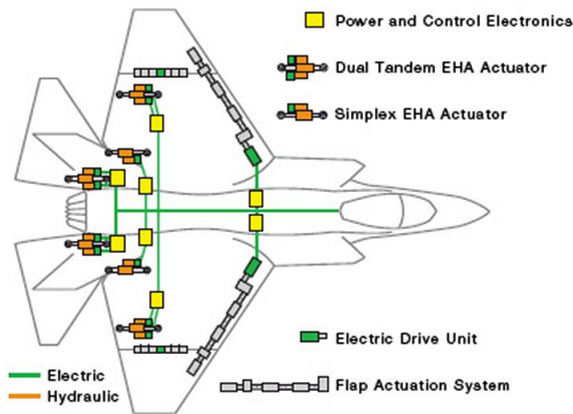


Figure 1. F-35 “Power-by-wire” systems

Dell Services has more than a decade of experience working with Lockheed Martin on the design and development of the Off-Board Prognostics & Health Management (OBPHM) system during the Concept Demonstration and System Design & Development phases of the F-35 program (see Figure 2). This knowledge and experience was used to create a low-cost OBPHM-compatible prognostic demonstrator framework that will be used to present the viability of prognostic capability for the EHA system.

The experimental approach, the test facility, the data analysis and the findings are discussed in the following sections.

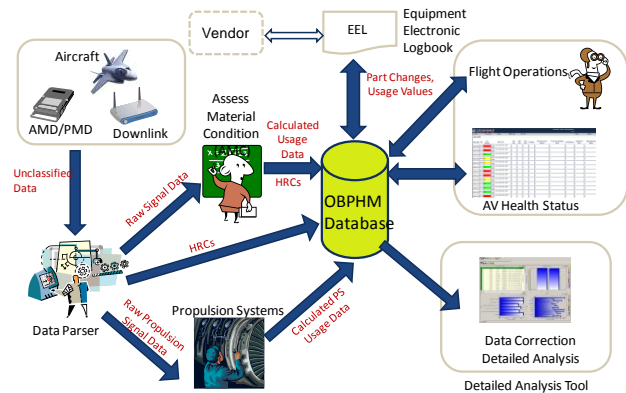


Figure 2. Autonomic Logistics Information System (ALIS) OBPHM system sketch

## 2. EHA SYSTEM PROGNOSTIC EXPERIMENTS

Various experiments were designed and conducted at Moog’s East Aurora Aircraft Controls facility to characterize fault-to-failure progression (FFP) signatures of the EHA. The original objective was to emulate insulated gate bipolar transistor (IGBT) degradation in the motor drive H Bridge and assess if this degradation could be identified during initiated built-in test (IBIT) by examination of the actuator response to an input command. However, since the servo drive IGBTs are typically packaged as a single hybrid module with DC-link and gate drive inputs, along with 3-phase motor drive outputs, it was impractical to vary the high-side collector resistance to directly emulate IGBT degradation. Hence focus was shifted to the more accessible gate driver board and propagation of damage from this isolated low-voltage control circuitry to the high voltage power electronics circuitry.

Our fundamental hypothesis is:

- Degradation or damage to the discrete circuitry surrounding the gate drive logic could result in measurable drift from nominal switch operation.
- Decreased dead-time between high- and low-side device switching could lead to both high- and low-side power transistors in resistive (linear) mode momentarily.
- Excessive power transistor heating accelerates wear and ultimately results in premature end of life.

We then analyzed the gate driver circuitry to identify the components that would affect device switching parameters, be relatively easy to apply synthetic degradation to, and perhaps, be prone to wear. The EHA design utilizes optocouplers to isolate the gate drive circuitry from the PWM controller. Prior research strengthens our basic premise that as damage to the optocoupler accumulates, its ability to deliver switch signals to the IGBT on time may be inhibited. Additionally, the optocoupler circuit includes a series resistor that can easily be modified to synthesize

decreased current transfer ratio (CTR) consistent with damage to the optocoupler's crystalline lattice. Therefore, the optocoupler was selected.

Applying our proven servo drive damage propagation analysis methodology, shown in Figure 3, entails:

- Applying various fault conditions to each critical stage of the servo drive, starting with the gate driver (D1 in Figure 3) and progressing to the power transistors (D2) and motor windings (D3) of each phase.
- Conducting lab experiments to acquire and characterize the pertinent multivariate servo drive data associated with each fault condition and the resulting stress effect on other components in the system.
- Analyzing the FFP signatures of the acquired multivariate data to produce reasoner algorithms that effectively detect precursor events that mark incipient failure of the servo drive subsystem or damage to its individual components.

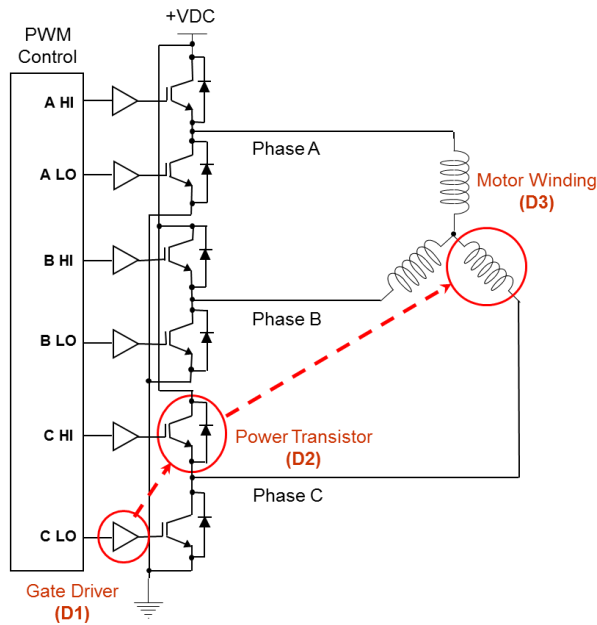


Figure 3. Damage propagation analysis methodology

The optocoupler was synthetically aged by changing the series resistor value to acquire FFP signatures from no degradation to total device failure, under various load conditions. Five different resistance values were utilized to emulate 25%, 50%, 75%, 95% and 100% degradation. The 100% degradation represented a catastrophic collector-to-emitter open circuit fault. The acquired data were recorded in a database and used to develop analysis algorithms to assess the SOH and estimate the remaining useful life (RUL) of the actuator servo drive power electronics.

## 2.1. EHA System Prognostic Testing

The IBIT and power-up built-in test (PBIT) requirement is that failure detection be designed to detect and isolate greater than 99% of all functional failures within the EHA system. IBIT is executed in two parts, one to test the processing circuitry and the other to verify the drive electronics and the EHA, including the bypass solenoids. These tests are capable of being invoked separately, with the processing circuitry test always engaged prior to the drive electronics/EHA test. The purpose of IBIT is to execute a series of test steps as a means to detect latent system failures that would prevent the system from meeting its reliability and availability requirements. IBIT for the EHA system is invoked by a signal from the vehicle management computer (VMC). IBIT is exited upon a Terminate IBIT command from the VMC.

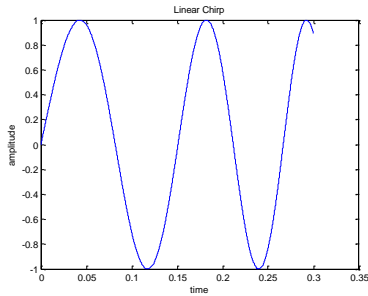
The particular test that would lend itself to evaluation of the frequency response characteristics consistent with the goals of the test program would be the EHA rate test.

The duplex actuators on the flaperon and horizontal tail each have dual pumps and motors that are tested. In addition, there is triplex redundancy in the control electronics (two physical, one model) for each pump/motor. Consequently, six maximum rate command tests are run.

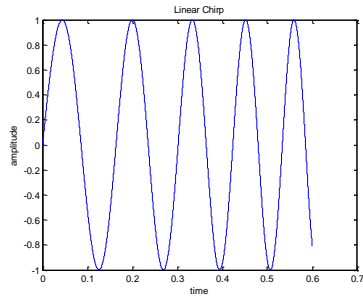
The critical item is the amount of time available for each of the six tests. It can be calculated that there is approximately a 300 msec time allocation for each of the six max rate commands. There is a total maximum time allocation to IBIT for the EHAS so any dynamic movement of the surfaces should be kept to less than 300 msec as a target value for each of the six tests.

In order to maximize the prognostic signature available during IBIT, different types of frequency shaped actuator motion profiles were tested. These included: 1) a chirp type of sine sweep command, and 2) a sinusoidal input at a selected frequency. The bandwidth for these sinusoids was chosen at the upper end of the frequency response capability for the actuators in order to maximize the number of sinusoids in the motion profile. For the chirp signals, this was initially selected to be a 5.5 to 10 Hz frequency sweep. The waveforms for a linear chirp for a 300 and 600 msec time period are illustrated in Figure 4(a) and Figure 4(b), respectively. The 600 msec waveforms were utilized in the test program to identify any improvements to the prognostic signature that could be obtained with more sinusoids in the stimulus.

An exponential chirp waveform for a 5.5 to 10 Hz frequency sweep is illustrated in Figure 5(a) and Figure 5(b) for a time period of 300 and 600 msec, respectively. For this limited frequency sweep and time, there is not a significant difference in the waveforms between the linear and exponential chirp signal.

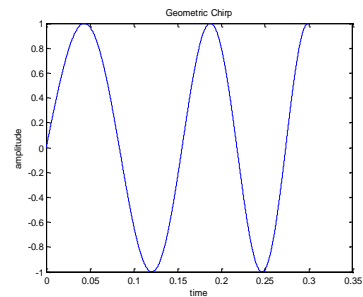


(a) 300 msec, 5.5 to 10 Hz

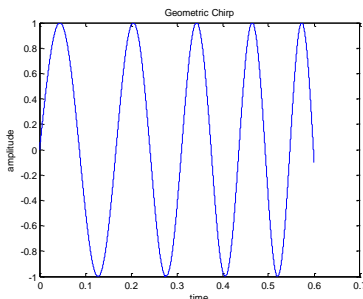


(b) 600 msec, 5.5 to 10 Hz

Figure 4. Linear Chirp Waveforms



(a) 300 msec, 5.5 to 10 Hz



(b) 600 msec, 5.5 to 10 Hz

Figure 5. Exponential signal waveform

## 2.2. Experiments Overview

The Moog test facility in East Aurora, New York was utilized to conduct the experiments. The lab setup is

illustrated in Figure 6 through Figure 8. This includes the flaperon EHA, shown in Figure 6, the power drive electronics (PDE) unit shown in Figure 7, and the integrated test computer (ITC) in Figure 8.

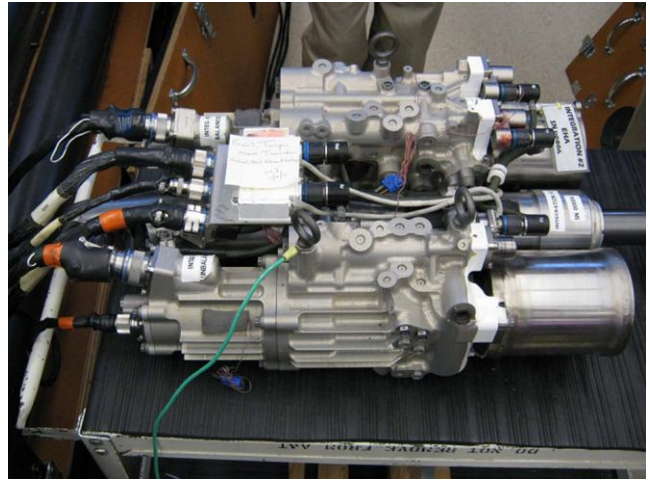


Figure 6. Flaperon EHA used in experiments

The PDE shown in Figure 7 is where the fault-seeded gate driver circuit is housed. Oscilloscope probes, visible on the left side of the PDE, are monitoring the gate driver output.



Figure 7. Power drive electronics (PDE) unit

The ITC shown in Figure 8 controls the operation of the test stand and downloads data from the digital recorder. Motion profiles and test stand digi-rec (digital recording) commands are configured from this workstation.

Figure 9 shows a portion of the gate driver circuit schematic, which was the source of fault seeding. As previously mentioned in Section 2.0, different resistor values were placed in the input diode's cathode branch to simulate degradation of the optocoupler. The resistor values were changed by physically removing a resistor and soldering in a new one with the specified ohmic resistance.



Figure 8. Integrated test computer (ITC)

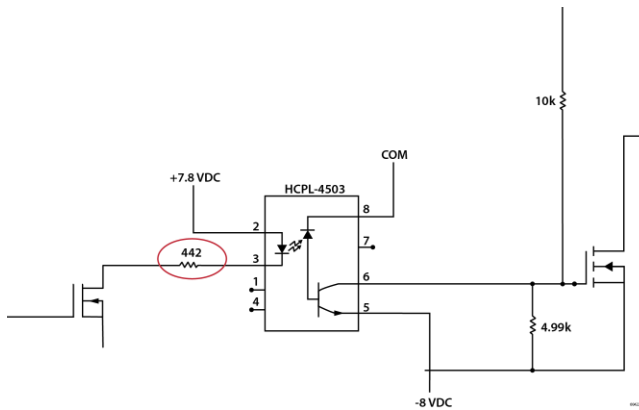


Figure 9. The gate drive circuit with  $R_{gd}$  (gate driver resistor) circled in red

Five different levels of degradation were chosen, simulating an even decline in health from nominal degradation to terminal failure. Note that the actual degradation percentages are slightly different from the previously stated values due to the available resistor characteristics.

Hardware Config.	Actual Degradation	$R_{gd}$ (ohms)
Baseline	0%	442
Degradation 25%	25.70%	681
Degradation 50%	55.10%	953
Degradation 75%	76.30%	1150
Degradation 95%	95.70%	1330
Terminal Degradation	100%	1370

Table 1. Gate driver configurations

### 2.3. Motion Profiles

It was determined that the best plan of action was to collect baseline data for each individual motion, and then string

them into a composite motion of appropriate length to improve test and analysis efficiency. A detailed summary of the motion profiles is shown in Table 2.

Motion	Signal	Frequency – Hz Max Velocity (°/sec)	Duration (msec)	Amplitude (In.)
1	Run	54° / sec	560	3
2	Step (50% duty)	60° / sec	500	.3
3	Sinusoid	4 Hz	500	0.227
4	Sinusoid	6 Hz	500	0.151
5	Sinusoid	8 Hz	500	0.113
6	Linear Chirp	5.5 -> 10 -> 5.5 Hz 20 ms dead time	600 -> 300	Max allowable
7	Geometric Chirp	5.5 -> 10 -> 5.5 Hz 20 ms dead time	600 -> 300	Max allowable
9	Linear Chirp	5.5 -> 10 -> 5.5 Hz 100 ms dead time	600 -> 300	Max allowable
10	Geometric Chirp	5.5 -> 10 -> 5.5 Hz 100 ms dead time	600 -> 300	Max allowable

Table 2. Motion description

### 2.4. Monitored Variables

Several variables were monitored in the test fixture as well as hardware test points inside the gate driver circuit. All software test points were downloaded and all hardware test points were captured with a 2 GS/s oscilloscope.

Table 3 provides a complete listing of collected data points.

Variable	Sampling Frequency	Test Point	tag	//
270 V internal	8064 Hz	Software	ITC	#18 A1
270 V Bus Link Capacitor Voltage	8064 Hz	Software	ITC	#19 A2
Phase A Motor Current	8064 Hz	Software	ITC	#110 A3
Phase B Motor Current	8064 Hz	Software	ITC	#111 A4
Phase C Motor Current	8064 Hz	Software	ITC	#112 A5
Commanded Position (inches)	80 Hz	Software	ITC	#113 A6
Actuator Position (inches)	560 Hz	Software	ITC	#117 A7
Actuator Velocity (rad/sec)	2240 Hz	Software	ITC	#118 A8
Local Motor Velocity CMD	560 Hz	Software	ITC	#119 A9
Quadrature Axis Current Error	8064 Hz	Software	ITC	#120 A10
Phase A Gate Driver Command (GATE_DVR_A_OUT)	2 Gs/s	Hardware	O-Scope	Scope 1
Phase A IGBT Gate (G2_A)	2 Gs/s	Hardware	O-Scope	Scope 2

Table 3. Monitored test points

### 3. DATA ANALYSIS

A significant amount of data was collected during the testing. Approximately 2.4 gigabytes of data were collected through the course of 72 trials. Each motion profile that was tested was recorded three times for repeatability analysis. The data analysis methodology uses data collected while running a composite motion profile with different resistor values, as previously shown in Table 1.

#### 3.1. Analysis Methodology

Data collected with a resistor value of 440  $\Omega$  is used as the “golden,” “healthy,” or “reference” data. The goal of this data analysis methodology is to create a signature that can be used to compute the level of degradation of any future test runs. This methodology computes differences from the

golden signature and sums the differences over time. The summed differences relate to fault degradation. The fault degradation is used to determine a fault-to-failure progression.

The algorithm flow of extracting signatures from a data set is described in Figure 10. There is a basic assumption that there is both a nominal data set and a fault-seeded or degraded data set. There is also an assumption that the degraded data sets are ordered in increasing levels of degradation. We begin with the “golden” data set. Each data set (nominal and degraded) is made up of a certain number of measured parameters, recorded at a certain frequency over a certain time interval.

```

1: For golden data set,
2:   For each time i,
3:
4:
5:   _____
6:   End For
7: End For
8: For each degraded data set,
9:   For each time i,
10:    For j = 1;3, _____
11:
12:   End For
13:   _____
14: End For
15: For each windows, T
16:
17: End For
18: Compute a minimum gap
19: Compute the maximum of the minimum gap
20: End For
    
```

Figure 10. Analysis methodology procedure

A brief explanation of the analysis methodology to identify the actuator motion profile that provided the best prognostic signature follows. From Line 1 to Line 6, three amplitude values and a middle value  $m_i$  is calculated from three amplitude values for each time in the golden data set. This means that the nominal data set produces a total of  $n$  middle values where  $n$  is the number of golden data sets. From Line 8 to Line 14, a degraded data set is considered, and an intermediate distance value,  $D_{T,k}$ , and the average of the three intermediate distance values,  $\bar{D}_{T,k}$ , are computed, where  $k$  refers to the degradation level. Thus a total of  $n$  average intermediate values are computed for each level of degradation. From Line 15 to Line 17, the final average of the distances is computed from the data set consisting of  $n$  time values. A window size is chosen of  $w$  time values where  $w$  is less than or equal to  $n$ . This produces  $T$  final

averages for each of the degraded data sets. Last, we find the maximum of the minimum gaps. A minimum gap is generated by first calculating the difference between the final average for successive degradation levels,  $D_{T,k} - D_{T,k-1}$ . A total of  $k-1$  differences are determined, one less than the total number of degradation levels. The minimum gap is the minimum of these differences and indicates the distances from the nominal relative to degradation. **The maximum of the minimum gaps provides the right edge of the time window  $T$  corresponding to the signature.** This tells you where there is a good separation of the means of the measured parameters that correspond with degradation. Figure 11 shows the maximum of the minimum gaps in green, motion profile in black, and best signature time in red.

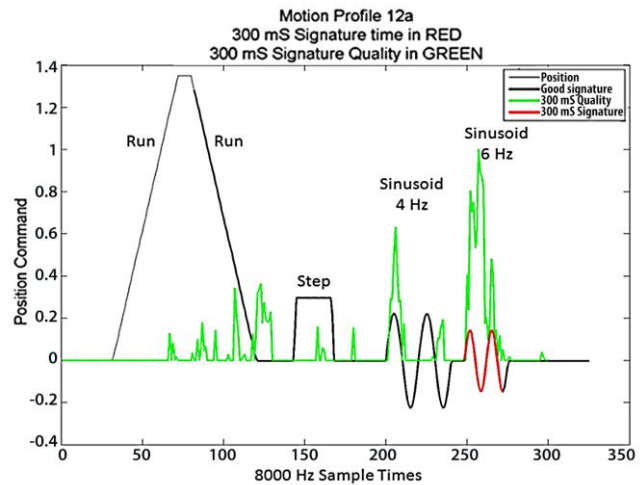


Figure 11. 300 ms signature search motion profile

The application of the methodology identified the quadrature axis current error (QACE) data signal (see Figure 12) as the best failure precursor when combined with a simple 6 Hz sinusoidal motion profile.

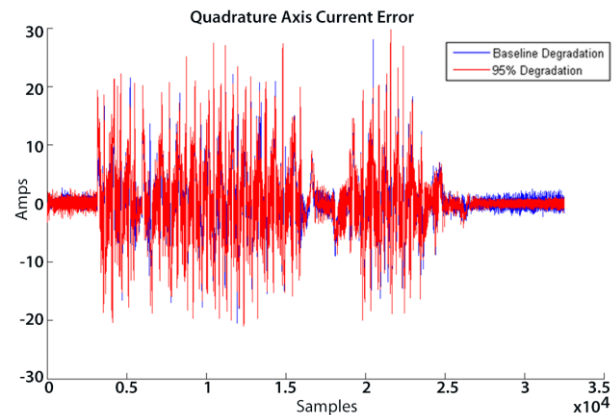


Figure 12. Quadrature axis current error (QACE)

The initial data analysis methodology used all of the data collected at 8 kHz. However, since data are written to the VMS bus in the aircraft at only 80 Hz, the prognostic analysis methodology would need to work at this data rate. Consequently, the analysis was performed again with the data decimated down to 80 Hz. The results are illustrated in Figure 13.

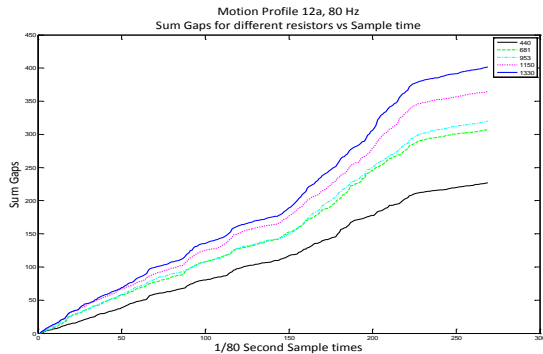


Figure 13. Total sum\_gaps for 80 Hz samples

After the QACE 8 KHz data from all test runs were decimated to 80 Hz, the search for the maximum of minimum gaps produced the same signature as shown in Figure 11 above.

The fault-to-failure signature utilizing the analysis methodology applied to the QACE data is plotted in Figure 14 for all resistor values: 440 Ω (○), 681 Ω (□), 953 Ω (■), 1150 Ω (◇), and 1330 Ω (×). This plot illustrates the increasing prognostic signature as the synthesized degradation increases.

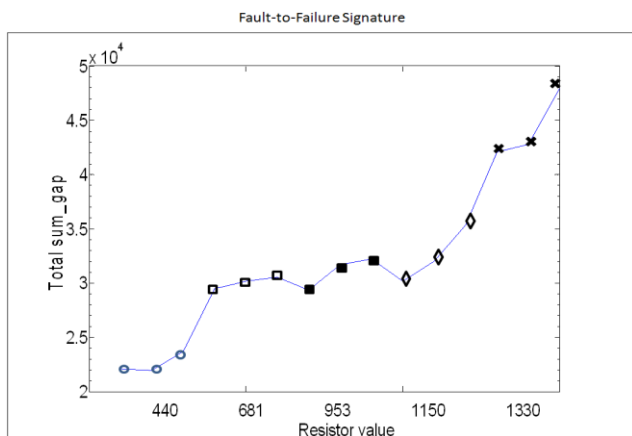


Figure 14. Total sum

SOH assessment is calculated for subsequent tests by comparing the subsequent test measurements with recorded reference signature values. The subsequent total sum gap is compared with totals recorded in the signature file and the SOH is derived by interpolating the new total between the recorded subgaps and percentage degradation for different resistors.

The RUL is calculated from the SOH assessments acquired at various times. A linear RUL estimate is calculated from the assessment times and SOH at the two most recent assessments. The change in SOH per unit of time is assumed to be a constant. The RUL is a linear extrapolation of the two most recent states of health and assessment times. More accurate RUL is attainable by monitoring SOH degradation over real time on a real system.

#### 4. OFF-BOARD PHM DEMONSTRATOR AND REASONER FACTORY

The work on the PHM Demonstrator consisted of establishing the system engineering tasks and activities required for the design of a signal parser and OBPHM Demonstrator. Dell Services provided domain knowledge, software engineering, and business process expertise required to design the OBPHM Demonstrator such that future integration of algorithms into production systems is realistic.

The flow depicted in Figure 15 represents the path the data follow from on-board to off-board systems for processing. The data are taken off the aircraft by means of a portable memory device. The unclassified PHM and signal data are split off from the classified data and transferred to the OBPHM system, which calculates, tracks, and visualizes RUL of various components. Additionally, maintenance work orders can be generated for repair and replace actions as needed. The rationale for the approach that was taken to minimize costs associated with transferring technology from a research and development to a production environment is self-explanatory. To achieve the highest possible level of compatibility between the concept demonstrator and the production OBPHM system, software components were developed with functionality similar to that in a production environment. Also, the same software development toolset that Dell used to support OBPHM development was used for this work.

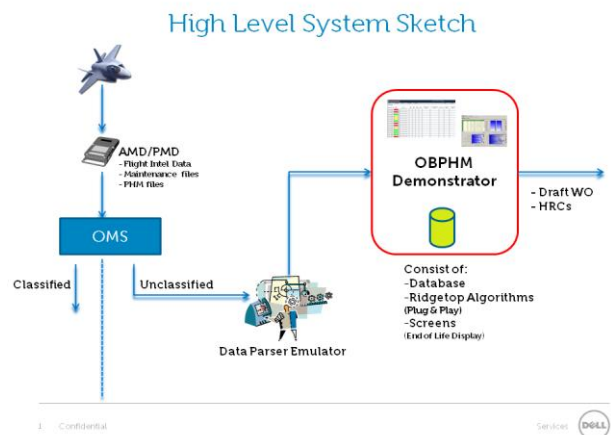


Figure 15. System sketch



The tools developed are outlined in Table 4:

Requirements Tools	
Starteam	<a href="http://www.borland.com/us/products/starteam/">http://www.borland.com/us/products/starteam/</a>
Modeling Tools	
Together Control Center 2007	<a href="http://www.borland.com/together/">http://www.borland.com/together/</a>
XML Spy	<a href="http://www.xmlspy.com/">http://www.xmlspy.com/</a>
Configuration Management Tools	
Subversion	<a href="http://tortoissvn.tigris.org/">http://tortoissvn.tigris.org/</a>
Database Tools	
Oracle	<a href="http://www.oracle.com/">http://www.oracle.com/</a>
Toad Oracle (Tool for Database Administrators)	<a href="http://www.toadsoft.com">http://www.toadsoft.com</a>

Table 4. System design tools

The Demonstrator Software Prototype and Data Parser Software Emulator Prototype runs in a single environment for demonstration purposes. A distributed architecture was not developed as part of the initial capability. Figure 16 represents the architecture for the Demonstrator. The EHAS architecture consists of components and interfaces and supports loose coupling. Each component implements a single related set of functionality.

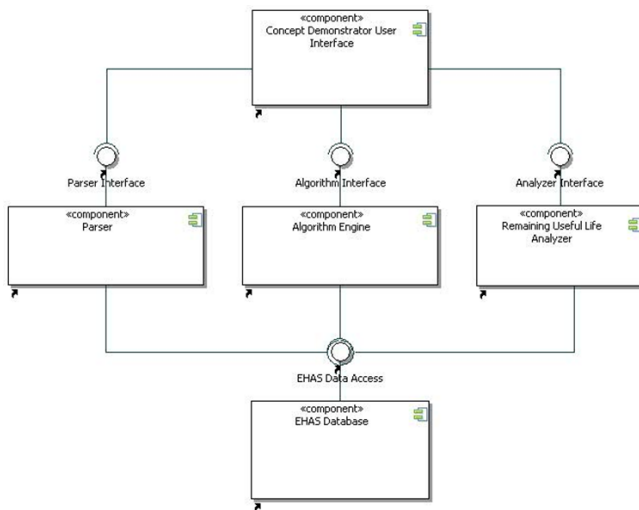


Figure 16. Demonstrator overview

The EHAS architecture consists of the following components:

- **Concept Demonstrator User Interface:** This component represents the interface to the end user for interaction with the Parser component, Algorithm Engine component, and the RUL Analyzer component.
- **Parser Component:** The Parser component has the capability to parse datasets, offers the capability to view the parse results, and the ability to delete these results.

The Parser component has a Parser Interface, which is utilized by the Concept Demonstrator UI component.

- **Algorithm Engine Component:** The Algorithm Engine component has the capability to run algorithms and to view the run results. The Algorithm Engine component has an Algorithm Interface, which is utilized by the Concept Demonstrator UI component. The algorithm engine also ensures that parsed data sets are processed in chronological sequence.
- **Remaining Useful Life (RUL) Analyzer Component:** The RUL Analyzer component offers the capability to view the RUL analysis and has an Analyzer Interface, which is utilized by the Concept Demonstrator UI component.
- **EHAS Database:** The EHAS Database component offers a Data Access Interface to the other components to view, create, update, and delete data required by the various Concept Demonstrator functions.

The architecture puts in place the basic framework of being able to parse signal data, process parsed data, and display RUL. More work is needed to make the system more robust. For example, as we learn more of the performance characteristics of the air vehicle it may be necessary to recalculate remaining life of one or more components. Flight/system data are stored starting, in some cases, during production. There can be cases where all the flight data need to be reprocessed starting from day one or some other point in time during the life of the aircraft.

This basic design principle introduces additional complexity. For example, it does not make sense to recalculate remaining life for components that have already been scrapped. Also, parts may be refurbished, returned to the supply chain and end up on a different aircraft from the first install; the component may even end up on an aircraft belonging to a different country's air force. Therefore the OBPHM system needs to be able to track which component was installed on which air vehicle and the period of time that it was installed. Additionally, it is essential that Performance-Based Logistics contracts are put in place with partner nations or we may not be able to feed the system all the needed flight and performance data to effectively perform prognostics and health management. PHM and remaining life data accompany the component throughout its life so for each period that a component is "on wing" it is known what its start and end RUL characteristics were.

Finally, as the aircraft matures over time, changes will be incorporated. For the PHM demonstrator the signal definitions are most important, as this is a configuration-managed item. So it is not enough to know which part flew in which air vehicle for a given flight of the air vehicle. We must also be able to determine the correct set of signal definitions that are an essential input to the parser function.

Since we expect to tighten tolerances used in the PHM algorithms as the air system matures, it is essential that algorithms do not require to be recompiled with version updates to the OBPHM system. Many copies of the OBPHM system run in multiple locations and countries. Version updates of the OBPHM software are therefore not trivial and are time-consuming to roll out. Hence the design requirement for PHM algorithms to be parameter-driven, which in the current version of the demonstrator they are not.

#### 4.1. Reasoner Development

For the final demonstration, two reasoners were successfully developed and integrated with the OBPHM demonstrator, using the supplied test run datasets presenting both the SOH and RUL. In addition, the system was able to repeatedly discover the signatures of interest as the data were decimated from 8 kHz down to 80 Hz, proving that the same algorithm without change could detect the degradation with fewer data points. Figure 17 and Figure 18 show the final EHAS concept demonstrator using the datasets at 80 Hz with a 300 ms detection window for SOH and RUL.

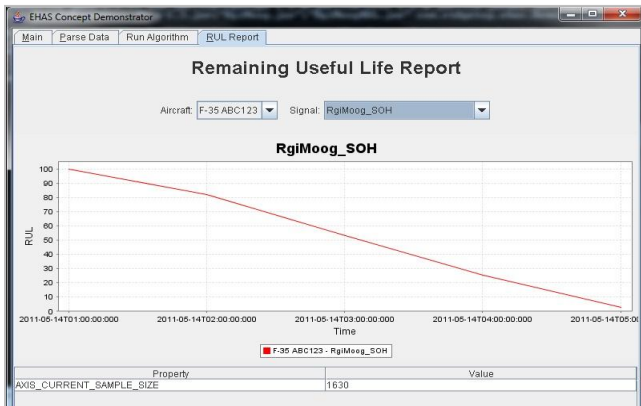


Figure 17. Results SOH report screen

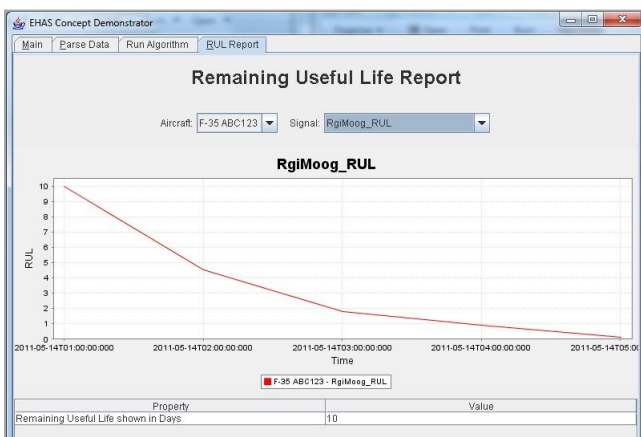


Figure 18. Results RUL report screen

## 5. PROGNOSTIC ARCHITECTURE AND IMPLEMENTATION CONSIDERATIONS

The essential idea of the prognostic methodology presented in this paper is to sum the absolute value of the difference between a degraded signal and a golden (reference) signal. The gaps or differences between different levels of degradation are then computed and the running sums again calculated to obtain sum\_gaps. An average gap is then computed to be used as an indicator of the strength of the prognostic signal. The implementation of two error summations essentially applies a “magnifying glass” to the differences between the degraded signals and the reference value so that the best failure precursor signal, on a relative strength level, can be identified.

The application of this methodology identified the QACE as the data signal that provided the best failure precursor when combined with motion profile 4, a 6 Hz sinusoidal motion.

### 5.1. Prognostics Architecture Feasibility

There are two primary factors in going forward with technology implementations on the F-35. The first is technical feasibility, and the second is return on investment (ROI). By ROI, we essentially mean that the technology has to earn its way on the aircraft by providing a cost benefit that will result in net cost savings over the life of the program.

#### 5.1.1. Technical Feasibility

The F-35 vehicle systems network (VSN) is the primary means for transfer of data between vehicle system subsystems and components. This includes transfer of data between the VMCs and the EHA EU.

The lab experiments and data analysis indicated that a prognostic algorithm to calculate RUL for a degraded optocoupler is feasible for this particular failure mode when tested in a lab environment. The key question is how would the algorithm perform in an operational environment? Since the motion profile would be implemented during IBIT, which would provide a somewhat repeatable field environment, variability in the EU states and resulting effect on the prognostic algorithm would be minimized. The concern is then the effects of noise and other environmental factors such as temperature on the QACE signal and RUL calculations.

Temperature variations could have a significant effect at the extremes of the operational environment. It is highly probable that the EU behavior would be significantly different at -20 °C than at 23 °C due to lower fluid temperatures requiring more power input to the motor. Temperature and resulting power variations and their effect on the QACE would have to be considered in the prognostic algorithm.

Variations in signal strength or noise due to other external sources such as variability in components would most likely have to be compensated for. The strategy computes SOH utilizing the sum\_gaps obtained from the laboratory testing based on field data obtained during IBIT. The RUL could then be inferred from the current sum\_gap value and its rate of change and projected time to reach the 100% degradation value. However, since the sum\_gap is essentially a “double summation” over time of the errors from the golden values, variations in noise could produce significant deviations from the sum\_gaps established in the laboratory testing. This is illustrated in Figure 19. Four levels of random, Gaussian, zero-mean noise were simulated (with standard deviations of 0.1, 1.0, 3.0, and 5.0). Note that the Iq deviation values have an amplitude mostly less than 10 A. The chosen noise signals thus have standard deviations ranging from 1% to 50% of the amplitude of the signal. The blue line represents the sum gap between the 681 and 440 ohm resistance values, the green line between the 953 and 440 resistance values, etc. There are two reference levels on the plot, one with the reset function (dashed horizontal colored lines), and one without (solid colored horizontal lines). The one without is the reference value of interest since the reset function is neither necessary nor desired for a prognostics implementation based on IBIT data.

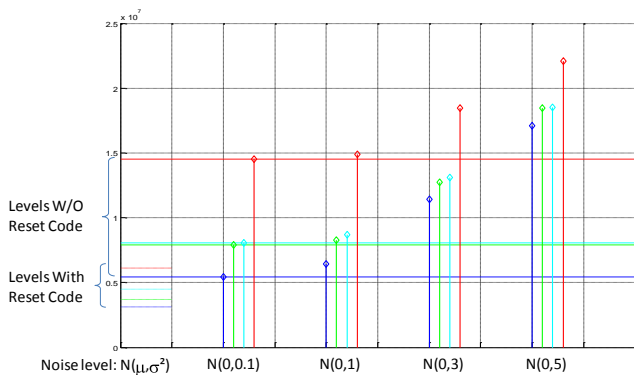


Figure 19. Effect of noise on Sum\_Gap calculations

Would the QACE signal have much variation due to noise or temperature? We do not know the answer to that. But as the plot illustrates, even with a standard deviation of 1, significant errors in the reference values used to estimate RUL would be incurred. Consequently, this would have to be monitored.

### 5.1.2. Return on Investment (ROI)

In order to justify implementation of a prognostic algorithm for the EHAS EU, it has to address a significant issue that could affect the operational costs and mission availability of the fleet. Performing prognostics for the optocoupler only would most likely not meet those criteria. The question is which other failure modes in the EU would also show up in the QACE signal? Another question is how would you isolate between an IGBT failure mode or optocoupler failure

mode or any other failure modes that would affect the QACE signal? Different failure modes would most likely have different sum\_gaps associated with their remaining useful life calculations. Distinguishing between failure modes would most likely require data fusion of different signals, and lab data to establish sum\_gap levels associated with failure.

## 6. CONCLUDING REMARKS

The laboratory testing at Moog on a simulated optocoupler failure proved to be successful with regard to performing degraded electronic component testing, identifying a motion profile that would fit the severe constraints associated with F-35 IBIT and the VMS architecture, and extracting a prognostic signal that showed progressive degradation commensurate with the induced degradation.

Distinguishing which failure mode might be showing up in the QACE signal is probably a more difficult challenge. This would most likely require a detailed circuit and failure modes, effects & criticality analysis (FMECA) for the power circuitry, and additional degraded component test work to identify a prognostic signature associated with a particular failure mode. A recommendation would be to perform testing on degraded EUs that have been returned due to failures so that a prognostic QACE signature could be established for a known component failure. EUs with failed IGBTs would be a significant opportunity for a test program.

One other strategy for implementation would be to start collecting the QACE signal during IBIT and then monitoring it as a precursor that something is going wrong even if the particular failure mode is unknown. Suggested steps in implementing the methodology are shown in Figure 20.

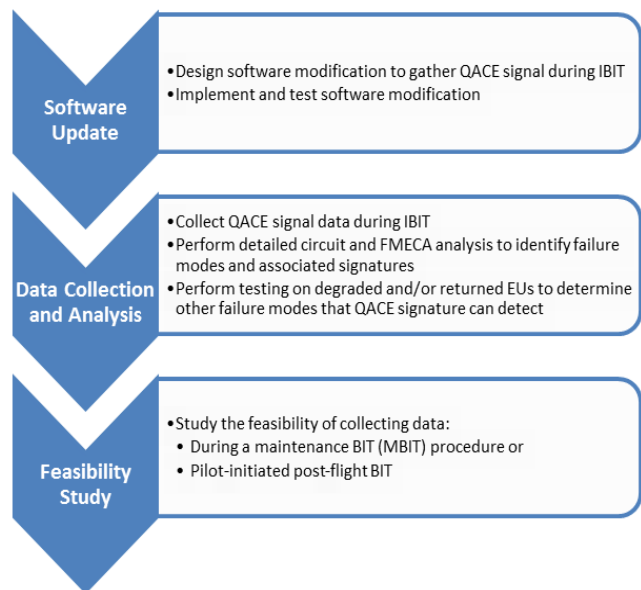


Figure 20. QACE tasks

## BIOGRAPHIES

**Sonia Vohnout** earned her MS in Systems Engineering from the University of Arizona. With a diverse background and experience, Ms. Vohnout is well-suited to manage Ridgetop's Advanced Diagnostics and Prognostics Division as Director. Ms. Vohnout joined Ridgetop Group after successfully building an electronic subassembly business in Mexico, working as a systems engineer at IBM, and handling overseas installations of software with Modular Mining Systems (now part of Komatsu). During her career, she has held executive management and senior technical positions. In addition, she has co-founded several companies. Ms. Vohnout is a board member of the Society for Machinery Failure Prevention Technology (MFPT) ([www.mfpt.org](http://www.mfpt.org)), an interdisciplinary technical organization strongly oriented toward practical applications. Ms. Vohnout has published several papers in the field of PHM.

**David Bodden** is a Senior Fellow at Lockheed Martin Aeronautics Company. His current research is in prognostics for mechanical and electrical systems. Prior to selection as a Fellow, his assignments included six years as Chief of the Control Law design and Analysis Group followed by seven years as the Senior Manager of Flight Control Systems. He has authored numerous papers and technical proposals, and managed numerous technology development programs. He has served on the AIAA Guidance, Navigation, & Control Technical Committee, is former Chairman of the Lockheed Martin GNC Technology Focus Group, is former Chairman of the SAE Aerospace Control and Guidance Systems Committee, and currently serves as Chairman of the Texas A&M Aerospace Advisory Board.

**Byoung Uk Kim, Ph.D.** is a Principal Research Engineer and a project lead for reliability analysis tool development at Ridgetop Group. The field of interest for his doctoral program was fault detection and root cause analysis systems, electronic prognostics, data mining and data analysis, and self-healing algorithms with autonomic computing. His collegiate repertoire also consists of numerous published papers in reliability analysis and autonomic configuration. Dr. Kim worked on a key NASA reliability/prognostics project in 2006 for Ridgetop. He has contributed to the development of innovative solutions that are currently deployed in the NASA ADAPT program at the Ames Research Center.

**Neil Kunst**, BSEE and Principal Systems Engineer, has more than 20 years' experience in product engineering, systems engineering, test engineering, logic design, software development, and project management. At Ridgetop he directs the development of the comprehensive prognostics and health management platform, Sentinel Network, which features a distributed software architecture with an embedded sensor network, centralized data collection, advanced reasoning, and asset management of complex mil-

aero systems-of-systems. Mr. Kunst has been recognized by NASA for outstanding performance on ground-breaking research related to electronic power system and electromechanical actuator prognostics.

**Patrick Edwards** is an Electrical Engineer at Ridgetop Group. He earned his BSEE from the University of Arizona in 2009 and specialized his studies in microcontrollers and embedded system design, computer architecture design, analog and digital control systems, and robotics. Mr. Edwards' undergraduate degree featured advanced studies in system modeling and embedded controller design. He is experienced in electrical and firmware design and integration, as well as PWB layout and embedded systems design. Mr. Edwards played a key role in the successful engineering of a Phase I DOE Small Business Innovation Research (SBIR) program titled "Uptime Improvements for Photovoltaic Power Inverters."

**Robert Wagoner** is a Senior Software Engineer at Ridgetop Group and longtime member of IEEE. At Ridgetop, Mr. Wagoner has been leading the R&D of Ridgetop's foundation prognostics and health management (PHM) application, Sentinel Network. He is also technical lead on a NASA Phase 2 SBIR program, and is contributing to the commercialization of Ridgetop's MAPR technology and construction of an advanced actuator testbed. He has expertise in robotics development and electric propulsion design for unmanned aerial vehicles. He has developed UAV GNC/IMU-6DOF with GPS and ground station software, and created mainstream use of electric ducted fan-jet models.

**Bill Gleeson** is a Senior Electrical Engineer at Ridgetop Group. He earned a BS in Engineering Math from the University of Arizona and an MS in Industrial Engineering from Arizona State University. He has six patents and a technical excellence award from PC Magazine. He developed prognostic algorithms for a number of actuator-related programs at Ridgetop Group and found the degradation signature contained in the QACE. Prior to Ridgetop, Mr. Gleeson was VP of Hardware Engineering at NetMedia Inc., and was Senior Scientist at Hughes Aircraft.

**Dennis Cascio**, MSEE, is a Staff Engineer in the Aircraft Group at Moog Inc. He has over 25 years of experience in power conversion electronics design. Mr. Cascio joined Moog in 2002 after working as a design engineer on switching power supply applications, followed by large industrial AC to DC power conversion systems. At Moog, he has been involved in the design of various motor control power stages including those used in the Joint Strike Fighter, Flight Control Actuation Systems.

**Steve Brzuszkiewicz** is a Staff Project Engineer in the Aircraft Controls Group of Moog Inc. He has 40 years of experience in the electrical engineering and control system fields. Mr. Brzuszkiewicz joined Moog in 1984 and holds

degrees from Kettering University (BEE) and SUNY at Buffalo (MSEE). At Moog, he has been involved in the design and development phases of flight control and vibration control actuation systems, including those used in the JSF F-35 and B-2 aircraft.

**Roy Wagemans** is currently Delivery Director for Dell Services and responsible for coordinating Dell's services capabilities in various countries in the EMEA region. Roy has a background in software systems engineering and holds a business degree from the University of Henley. As of 1999 he has held a number of project and program management positions related to the Joint Strike Fighter program all related to the Prognostics & Health Management domain.

**N. Scott Clements** is a Systems Engineer at Lockheed Martin Aeronautics Company. He is currently researching fault degradation models and associated PHM techniques. He received his bachelor's degree from Mississippi State University in 1996 and his master's and doctoral degrees from Georgia Institute of Technology in 1998 and 2003, respectively. His research interests include PHM, data mining, verification techniques, and fault tolerant control systems.

# Poster Papers

# Performance Based Anomaly Detection Analysis of a Gas Turbine Engine by Artificial Neural Network Approach

Amar Kumar<sup>1</sup>, Alka Srivastava<sup>1</sup>, Avisekh Banerjee<sup>2</sup>, Alok Goel<sup>3</sup>

<sup>1</sup>*Tecsis Corporation, 210 Colonnade Road, Ottawa, ON, K2E 7L5, Canada*

*amar@tecsis.ca  
alka@tecsis.ca*

<sup>2</sup>*Life Prediction Technologies Inc., 23-1010 Polytek Street, Ottawa, ON, K1J 9J1, Canada*

*banerjeea@lifepredictiontech.com*

<sup>3</sup>*OMTEC Inc., 170 Bristol Road East, Mississauga, ON, L4Z 3V3, Canada*

*alok@omtec.ca*

## ABSTRACT

This present work follows our earlier research efforts on fault diagnosis and prognosis solutions considering statistical and physics based approaches. In-service performance analysis and detection of any malfunctioning in an operating small sized gas turbine engine using artificial neural network approach is the central theme of this work. The measured engine operating and performance parameters are used to train two neural network models, namely back propagation and generalized regression. Following the training and validation of the neural network model, simulation results for test data corresponding to various engine usage stages are found to be close by two models. The analysis identifies an anomaly in the simulated and measured data collected 17 months after the engine overhauling which may be attributed to deliberate adjustments in the operating parameters. A threshold for anomaly detection in terms of the probability levels for variation of the rated power capacity of the engine is also studied.

## 1. INTRODUCTION

Gas turbine (GT) configurations typically include single or multi-shafts, closed loop, simple or combined cycle, combined heat and power etc. and is typically used to generate electrical power. In spite of wide variations in the design, complexity, applications, operating conditions etc. the failure mechanisms are generally experienced to be identical. The GT life limiting issues of concern are low cycle fatigue (LCF), high cycle fatigue (HCF), creep, oxidation, corrosion, foreign object damage, etc. For effective and efficient engine health monitoring (EHM), a

host of parameters are usually monitored in modern GT that include speed, power, gas inlet pressure and temperature, exhaust and operating pressure and temperature, gearbox, journal and thrust bearing vibration and temperature (Clifton, 2006; Hoeft et. al., 2003).

On-line condition based monitoring of plant operation is extremely important for plant safety, reliability and availability and maximization of the power output and lowering life cycle costs. In recent years, with the rapid development of condition monitoring and forecasting, information processing, fault detection and artificial intelligence technology, it has been possible and feasible to monitor and forecast equipment condition and assess its health in real-time. It is well recognized that optimized maintenance practices within an industrial setting require the correct blend of condition based maintenance (CBM) strategies. (Hoeft et. al, 2003; Sobanska & Szczepaniak, 2006, Fast, 2010).

With usage, the health of the GT components deteriorates and affects the performance of the engine. The continuous degradation in performance and its rate plays a crucial role in establishing the time intervals between major overhaul (Clifton, 2006; Fast, 2010; Fast & Palme, 2010). The sophisticated technologies being incorporated into new gas turbines allow operations at higher pressures and temperatures with higher efficiencies. The trend of preventive maintenance at regular intervals is being replaced by CBM techniques to further reduce the maintenance costs. This makes intelligent and robust engine health and performance monitoring techniques that are sensitive to changes in the engine condition important for higher safety, reliability and availability of the units (Angeli & Chatzinikolaou, 2004). The capability to detect impending faults from the current engine conditions and issue early warning with minimum false positive is also desirable.

---

Amar Kumar et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

The malfunctioning in a GT can occur from component physical damage, changes in operational control and settings, faults in data monitoring devices, calibration etc. Some of the faults will unavoidably occur in an engine due to wear and/or malfunction of some components affecting system performance, as well as their combinations. Some of the malfunctioning can be rectified by taking appropriate corrective actions and requires fault diagnosis solution. Due to the associated complexity, evolutionary Artificial Intelligence techniques (Russel & Norvig, 1995) are being increasingly applied for diagnosis in GT.

Our current research attempts to investigate the feasibility of developing performance based GT-EHM system using operating data driven artificial neural network (ANN) modeling and techniques. The ANN models are developed such that it can represent the basic working of a freshly rebuilt GT and predict the critical performance metrics like the output power or the driven unit load. Anomaly detection is performed by comparing the predicted behavior of the freshly rebuilt GT obtained through trained ANN, with the actual measurements during the operation. The work related to the ANN as reported here follows our earlier research activities on diagnostic and prognostics solutions using statistical and physics based approach (Saxena, et.al., 2011; Kumar et.al., 2011). This work using ANN approach is our first attempt to look for an alternative solution for anomaly detection.

## 2. ANN FOR ANOMALY DETECTION

Anomaly detection methods are traditionally based on limit value checking of key measurable parameters without simulating the human reasoning activity. Numerous methods of anomaly detection leading to fault diagnostics have been developed and applied effectively to identify the machine/engine faults at an early stage using different performance parameters such as current, voltage, speed, temperature, and vibrations (Russel & Norvig, 1995; Zhu, 2009; Kumara et.al., 2012). A brief discussion is made in this section on ANN approach for anomaly detection. In the case of very complex time-varying and non-linear systems, where reliable measurements are very complicated and valid mathematical models do not exist, a number of different methods have been proposed by researchers. These methods come from the area of Artificial Intelligence and allow the development of new approaches to anomaly detection in dynamic systems like the GT.

Briefly, artificial neural networks (ANN) are massively parallel-interconnected networks that have the ability to perform pattern recognition, classification and prediction. A wide variety of engineering problems can be solved using ANN which is especially useful in situations where the data volumes are large and the relationships among variables are unclear or hidden. The network is trained to learn from the examples and forms an internal representation of the

problem (Russel & Norvig, 1995). For anomaly detection, it is needed to relate the measurement data to the ideal performance, and distinguish between normal and abnormal states. Input vectors are introduced to the network and the weights of the connections are adjusted to achieve specific goals. An adaptive algorithm automatically adjusts the inputs weightage to minimize the mean square of the error between the actual output value and the desired target value. A significant feature of neural networks is that an approximate model may be adequate to internally map the functional relations that represent the process. A recent work classified between normal and abnormal vibration data for a large turbine (210 MW) in a power plant using artificial neural networks (ANN). Self-organization map is trained with the normal data and simulated with abnormal condition data from a test rig (Patel & Prajapati, 2011; Samhouri et. al., 2009). Different unbalanced conditions are introduced on test rig at laboratory and vibration data is collected to simulate the network.

The choice of network architecture is dependent on the problem. Classification, linear or non-linear problems, with or without underlying system dynamics guides the choices of network composition and the topology. A single feedforward network describes a simple mapping network that can be used in classification or for mapping of simple input output functionality. It is defined through a single layer of neurons (Fast, 2010; Russel & Norvig, 1995). Hence, the knowledge storage capacity is restricted and only simple logic relations can be mapped. An extension of this is the multi-layer feedforward network, also found as multi-layer perceptron (MLP). This network architecture is defined through a minimum of one hidden layer of neurons. The number of hidden layers can be increased depending on the problem. However, a MLP with three hidden layers is sufficient to map every continuous function by adding a certain number of neurons to meet required complexity (Riad et.al, 2010; Sprech, 1991; Kaminski, 2010).

### 2.1. Model Selection

Back propagation (BP) algorithm is a steepest descent algorithm, in which the performance index is the mean square error (Russel & Norvig, 1995; Fast & Palme, 2010). It can be used to train multilayer neural networks. Based on the previous experience, a three-layer BP network, with tan-sigmoid activation function in the hidden layer and linear activation function in the output layer has been considered in this work. The training and simulation work is performed using MATLABM and utilizing its NN toolbox functions. As observed by many researchers, the BP model can approximate virtually any function to any degree of accuracy, provided sufficient hidden units and training sets are available. The learning rate and the momentum are two important parameters for training the network successfully. Levenberg-Marquardt (LM) algorithm has been used for network training, validation and testing as it finds the best



weights by minimizing the function. The ANN model consists of three layers of units: a layer of inputs is connected to a layer of hidden units, which is connected to a layer of output units. The activity of the input units represents the raw information that is fed into the network. The activity of each hidden unit is determined by the activities of the input units and the behavior of the output units depends on the activity of the hidden units and the weights between the hidden and output units. Due to the availability of a large number of training data, the number of neurons in the hidden layer is considered as 20 and 10 and validation of the model is done with 20 percent of the input data. In order to check the consistency of results, a generalized regression neural network (GRNN) modeling is also implemented. However, the input-output variables are kept the same as with the back propagation neural network (BPNN).

For most engineering problems, it is difficult to know beforehand how large and complex a neural network should be for a specific application. Besides the input-output variables, the optimum number of hidden units depends on several other factors, like number of training cases, noise in the targets, complexity of the function or classification, network architecture, type of activation function, training algorithm, and regularization. A decent performance of a ANN model may be obtained by setting the number of hidden layers equals one as more hidden layers are generally harder to train. Secondly, the thumb rules available only relate the number of neurons (N) in hidden layer with size of inputs and output variables, ignoring other factors. (Berry & Linoff, 1997, Boger & Guterman, 1997). These rules are generic and provide only a starting guide. Ultimately, the selection of architecture for neural network analysis will come down to trial and error. Furthermore, a rough approximation can also be obtained by the geometric pyramid rule. For the simple three layer network as considered in our work, the number of neurons in hidden layer was first set at square root of  $m \times n$  with  $n$  input and  $m$  output neurons, then increasing the number of nodes to achieve best fit. Most of the engineering analysis suffers from limited available data for analysis, thus restricting to small number of neurons in the hidden layer as offered by thumb rules and pyramidal rule. The risk of over-fitting tends to be more in such situations. Over fitting makes the network learn well from training data set, but performs poorly for test data set. In the present work, number of nodes in hidden layers is chosen by trial and error starting from the small number of 5. Under fitting occurs when there are too few neurons in the hidden layers to adequately detect the signals in a complicated data set. Considering all these aspects and availability of large 459 training data set in our work, several trials found the optimum node size to be 20 in the hidden layer with minimum error. This decision is also supported by the default neuron number of 20 adopted in Matlab™ NN toolbox functions, namely `nntool` and `nftool`.

Larger numbers of neurons adopted here than estimated by thumb rule in the hidden layer give the network more flexibility because the network has more parameters it can optimize. Besides, 10 neurons in a few cases were also considered for comparison of results. A few recent works considered similar number of nodes for vibration based ANN analysis while training was implemented using Matlab toolbox (Samhoury et. al, 2009; Massad, 2009).

## 2.2. Variable Selection and Data Acquisition

Variable selection and data acquisition are the two key elements for successful modeling of systems behavior and analysis. In ANN approach, the training data is crucial for creating a good generalization of network covering a broad range of the systems behavior. Maintenance and operational data from a small size GT engine was collected over three years for a large number of operating and maintenance parameters. However, for the current work, eight of these parameters are used for training and validation of the ANN model as well as testing and simulations. The selections of the input and output variables of the ANN have been made based on the physical significance, working and thermodynamic principles of the gas turbine operation. Four input variables selected are the gas pressure, two fuel control valve angles and air inlet temperature, while the four output variables selected are the speed, load, exhaust and operating temperatures. This approach has been based on performance monitoring guidelines prescribed by a typical turbine OEM (Hoeft et. al., 2003). Figures 1 and 2 show the typical data profile for the four input and output variables for the GT, respectively. The measured data has been scaled with respect to their mean value as shown in the figures.

## 3. MODEL TRAINING AND VALIDATION

As mentioned earlier, LM algorithm is used for the back propagation neural network (BPNN) training and model validation and results are displayed in Figures 3 and 4 for a four input-one output and four input-four output BPNN model, respectively. The training data was selected from measurements from the freshly rebuilt turbine operation immediately after a major overhaul and over the duration of the first three months of steady state operation. In the absence of any system model and reference data, it is assumed that this performance data would represent the healthy state of the turbine. In Figure 3, it is observed that the four input-one output BPNN model training yields very consistent and converging results. The output considered for this plot is the power or the driven unit load.

When compared with Figure 4, it is evident that the number of epochs required to reach the goal during training for the four input-four output BPNN model is much higher than that required for the four inputs-one output BPNN. It is possibly because of the numerically redundant input parameter of VGV angle (Figure 1) and the output speed

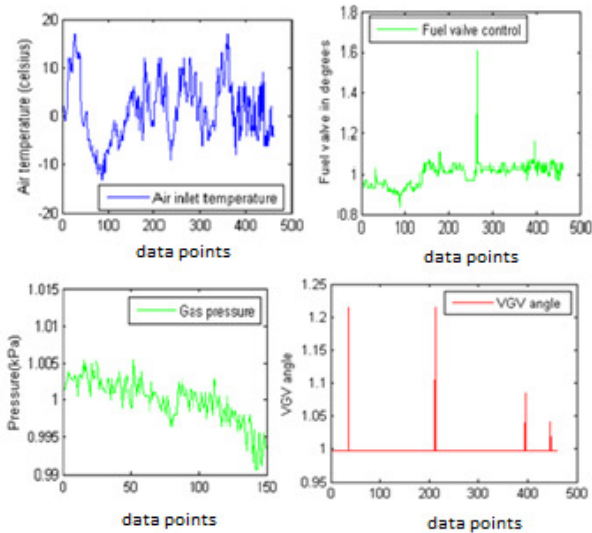


Figure 1: Typical data profile of four Input parameters, Air inlet temperature, FV Angle, Gas pressure and VGV Angle (data scaled)

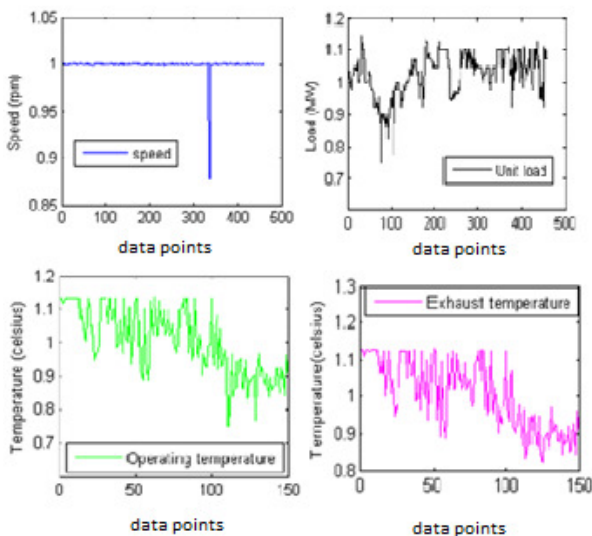


Figure 2: Typical data profile of four Output parameters, namely load, speed, exhaust and operating temperatures (data scaled)

(Figure 2), that are usually very consistent for industrial gas turbines working under steady state conditions. This suggests that the four input-four output model is not suitable for the current set of training data and model selection.

### 3.1. Generalized Regression Model

The present work is extended to general regression neural network (GRNN) model in view of its fast learning and optimal regression convergence abilities avoiding iterative procedures. The BPNN model needs a large number of

iterations to converge to the desired solution. GRNN is similar to probabilistic neural network (PNN) is an

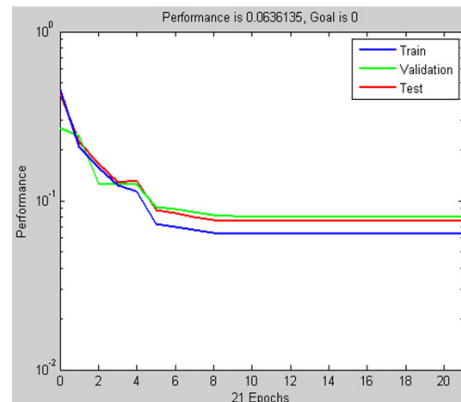


Figure 3: Training of the four input - one output (unit load) BPNN with Levenberg-Marquardt algorithm

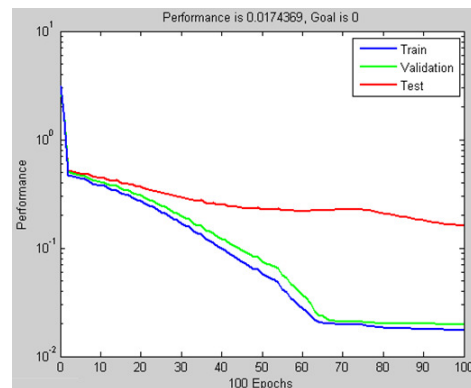


Figure 4: Training of the four input - four output BPNN with Levenberg-Marquardt algorithm

alternative solution when adequate training data are not available in real-time situations. This makes GRNN a very useful tool to perform predictions and comparisons of system performance in practice. It can be used for prediction, modeling, mapping, and interpolating or as a controller (Sprecht, 1991; Russel & Norvig, 1995; Kaminski, 2010).

Figure 5 shows the comparison of the measured data (target) and simulation results (output) obtained using the GRNN during its training with a four input-one output model while predicting the used data set. The figure suggests that the GRNN model may not be predicting like the BPNN model. This can be attributed to the choice of the NN model architecture and the values of the input and output parameters. The figure shows good correlation between the target and output suggesting the effective training of the GRNN model with the unit load output.

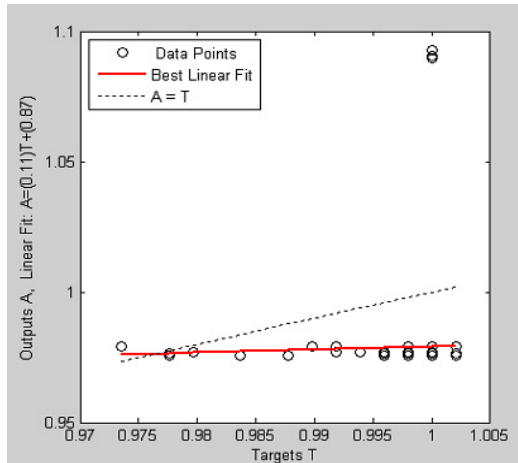


Figure 5: Comparison of measured data and simulation results using GRNN model using training data set with four inputs and one output model

#### 4. SIMULATION

The simulation of the engine performance parameters was done using the trained four input-one output BPNN as well as GRNN models. The different data sets used for the simulation are given below:

- **begin:** This corresponds to a group of 100 data points collected at the 3rd month since overhauling at the beginning of the operation period and represents a freshly rebuilt engine state as the data is close to the training set.
- **intermediate:** This corresponds to a group of 154 data points collected between the 17th-18th month of steady operation of the turbine and represents the 'intermediate' engine state as the data corresponds to the half-way to the end of the operation period.
- **end:** This corresponds to a group of 179 data points collected between 35-36th months of steady state operation and represents the 'used' engine state as the data is close to the end of the turbine design life cycle prescribed by the OEM.

The training data, as discussed earlier corresponds to the operating data during the first three months of operation and represents the performance of a freshly rebuilt system. The aim of the simulation study was to predict the engine performance for a freshly rebuilt engine, and compare it with the measured data to detect any anomaly and inconsistent behavior of the engine performance. The simulation output and the observed target data using BPNN and GRNN models are compared as displayed in Figures 6 and 7, respectively.

In Figure 6, it can be observed that the scatter in the plots (for the 'begin' and 'end' data sets) increases with usage. This is due to the fact that the trained BPNN model predicts the unit load for a freshly rebuilt engine. With usage the

performance of the engine degrades, and the measured performance index in terms of the unit load starts deviating from this bench line level simulated by the trained BPNN. However the maximum spread is observed for the 'intermediate' data which may suggest potential anomaly.

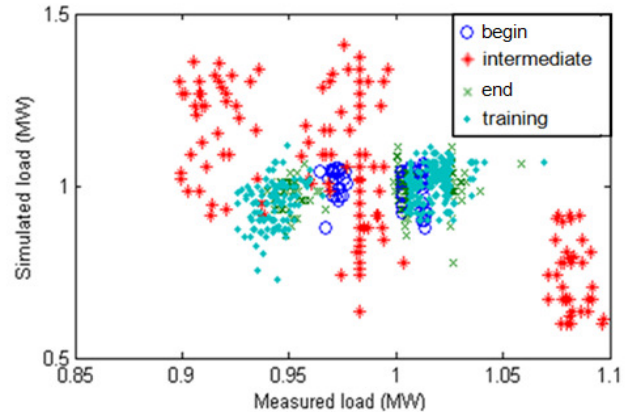


Figure 6: Comparison of the simulated and measured load (scaled) data for different data sets using BPNN model

In Figure 7, a similar trend is observed in the means of the simulated load for different test data sets using the GRNN model. This suggests the possible applicability of both BPNN and GRNN models for engine performance. Also, reducing the number of hidden neurons to 10 does not change the test data mean significantly. Consistently, the simulation results for the 10 neuron hidden layer exhibits a marginally lower value than those obtained with 20 neurons. The mean of the intermediate data is observed to be much lower than the other data sets and will be discussed later. In Figure 8, the comparison of the simulated and simulated data points using the trained BPNN for the 'end' data set is shown. The low correlation reiterates the observation that the deviation between the simulated and measured unit load is high for the 'end' data set.

Under idealistic situations, the simulated and target data points, respectively A and T, and so the linear fit and A=T line should all lie very close. However, for realistic situations, analysis shows that the data points in all cases fit linearly with high correlation coefficients (over 0.90) as typically shown in Figs. 5 and 8. This confirms the modeling - simulation output (A) is consistent with target data (T) as desired. Mismatch between the linear fit and A=T line are also observed and can be explained by the difference in the scattering nature of A and T data. Fig. 8 shows the simulation results have restricted and suppressed scatter band (0.95 to 1.05) when the scaled target load varies over 0.78 to 1.14. In other words, the simulation results seem to be somewhat conservative as compared to target data and tend to lie closely around the mean. This point needs to be examined further at a later stage with more trials and errors with network structure.

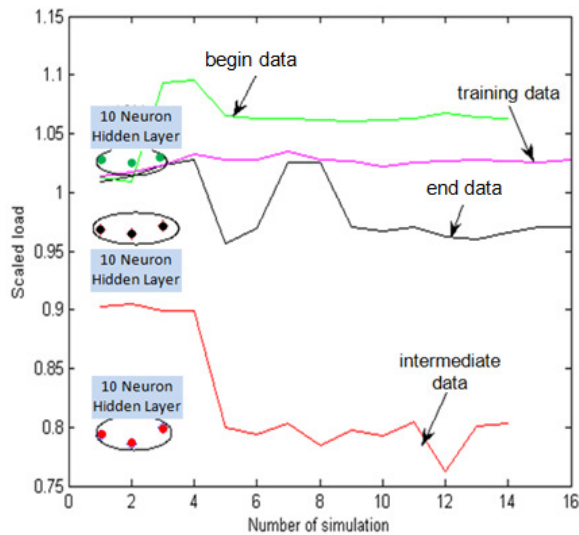


Figure 7: Simulation results for load demonstrating the variations for mean values for different data set using GRNN model

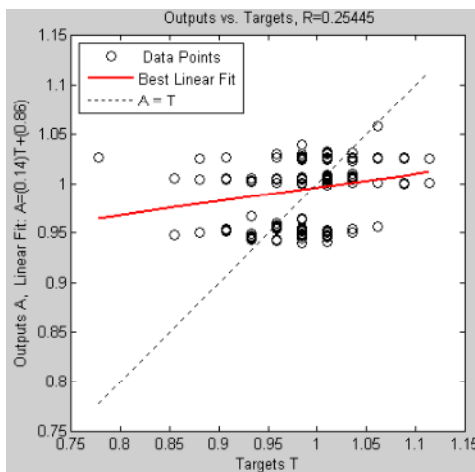


Figure 8: Network simulation output and target data (both scaled) for the 'end' data set with four inputs and load output BPNN model

### 5. RESULTS AND DISCUSSION

Discussion on the qualitative and quantitative analysis of results is made here in the light of engine performance based anomaly detection in the gas turbine using BPNN and GRNN models. The training of the four input-one output BPNN model with 459 data points, resulted in fast and adequate numerical convergence, as shown in Figure 3. However, attempts to train the four input-four output BPNN model using the same training data set did not yield similar satisfactory training performance (Figure 4). When tested after training, two of the four output parameters, namely load and EGT yielded better result compared to the speed and operating temperature. Hence, a four input-one output

GRNN model was trained to predict the unit load for further testing and simulation.

The unit load was simulated using the trained BPNN and GRNN models for different test cases represented by 'begin', 'intermediate' and 'end' engine condition, as displayed in Figure 6 and 7, respectively. The Figure 5 shows that the scatter between the simulated and measured data points is increasing steadily with the usage. This is because the trained BPNN models are simulated the engine performance of a freshly rebuilt system whereas the performance continuously deteriorates with usage. The low correlation of the simulation and measured data for the "end" data case highlights this deviation in Figure 7. Figure 7 reiterates the observations made in Figure 6 and suggests that the GRNN can also be used for the engine performance analysis. The effect of a 10 neuron hidden layer was also found to be insignificant.

Figures 6 and 7 throw meaningful lights on the turbine performance when the simulated load data is compared with the measured value for three data sets, namely 'begin', 'intermediate' and 'end' data sets collected after 3, 17 and 35 months of operating after a major overhaul. Ideally, the data points should scatter around one for healthy and normal state as simulated by the ANN models. The simulated results for different test data sets formed clusters that can be further analyzed for anomaly detection. Interestingly, the predictions are fairly close around one for all cases except 'intermediate' group (17-18 months data). These data points can be seen as widely scattered and seems to represent an anomaly and potentially indicating an unhealthy state. This can be explained by the fact the seasonal variation in the energy requirement and lower gas pressure generates lower unit load during this time period. This is confirmed by the Figure 9 where it is evident that the gas pressure and unit load for this data set are well lower than that used for training the models. Hence the ANN models trained at peak performance may not be able to predict the consistently for lower performance.

An alternative way to reassess the simulation results is by estimating means as shown in Figure 7. These means are based on at least 15 iterations and may be seen to be fairly consistent except the intermediate case. The deviation of means is around 15 to 25 percent lower than the expected values. Figure 7 also includes a few data points that come from GRNN model with 10 neurons at the hidden layer. No significant effect on the mean output is evident as compared to 20 neurons results.

In order to study the feasibility of anomaly detection using the deviation between the measured and simulated unit load, a threshold is introduced based on the measured load variation. A probabilistic level for the measured load will provide the chance of detecting the fault based on the usage state of the engine. The probability for the load yielding variations of 10, 20 and 30 percent of the measured power

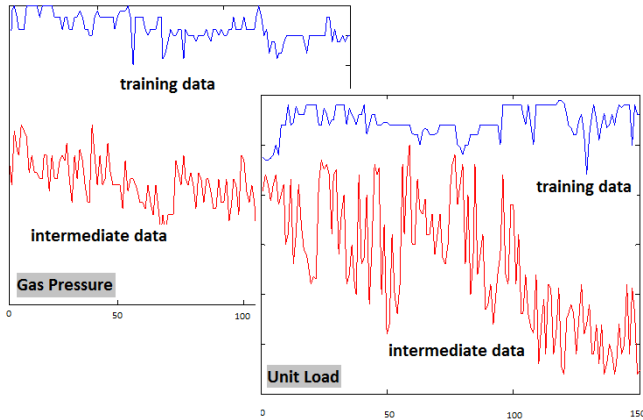


Figure 9: Comparison of the measured input gas pressure and output unit load data (both scaled) for the training and intermediate data set

of the gas turbine is estimated. Table 1 gives the results of the probabilistic analysis supporting our qualitative anomaly/no anomaly observations for predictions with quantitative (probabilistic) measure of load variations. As for example, setting the criterion at 20 percent load variation, the intermediate stage will have around 36 percent probability for anomaly detection. The two usages namely begin and end stages should have almost no chance for anomaly detection. Intermediate test data significantly contributes to the probability of variation dues to its lower magnitude compared to the other data sets and confirm the anomaly and under utilization of the GT capabilities even if it is deliberate.spacing between paragraphs. All papers should use Times Roman 10-point font throughout.

Load variation	begin	intermediate	end	training
10 %	6.0 %	68.5 %	3.9 %	4.8 %
20 %	0 %	36.5 %	0.83 %	0.33 %
30 %	0 %	17.5 %	0 %	0 %

Table 1: Computed probability level for the measured load

## 6. CONCLUSION

Artificial neural network models have been used for the performance based anomaly detection of a small sized gas turbine. Four independent input variables and output variables were selected from the operating and performance data. The health monitoring data was collected over the entire operational time between two major overhauls and used for ANN model training, validation and testing. However, the output power was considered as the major performance index for this study, as the four output models were not effectively trained for the selected data sets. Back propagation (BPNN) and generalized regression (GRNN) neural network models were implemented with MATLAB™ programming for this work. Both the models appear to well capture the behavior if the output unit loads. The comparison of the test data set collected 17 months

after engine overhauling and its ANN based simulation represent an anomalous situation. The difference in the output unit load may be due to the deliberate lower gas pressure adjustments made to accommodate seasonal variation in the power requirement. Otherwise, the scatter between the simulated and measured unit load increases with usage suggesting that the trained ANN representing the performance of a freshly rebuilt system can be used for anomaly detection. A high probability level over 36 percent is estimated for variation of the rated power capacity of the GT engine. However, samples collected towards the end of the entire operation cycle indicate no unhealthy signs.

## ACKNOWLEDGEMENT

Gas turbine engine monitoring data used in this paper for simulation and analysis was received from Life Prediction Technologies Inc. (LPTI), Ottawa, ON. Authors thankfully acknowledge the generous support and encouragement from LPTi.

## REFERENCES

- Clifton, D., (2006). *Condition monitoring of gas turbine engines*, Doctoral thesis, Department of Engineering Science, University of Oxford
- Hoefst, R., Janawitz, J. & Keck, R., (2003). *Heavy duty gas turbine operating and maintenance considerations*, GE Power Systems, GER-3620J
- Sobanska, P. & Szczepaniak, P., (2006). Neural modeling of steam turbines, *Proceedings of the International Multiconference on Computer Science and Information Technology*, (pp. 197–205), November 6-10, 2006, Wisla, Poland
- Fast, M., (2010). *Artificial neural networks for gas turbine monitoring*, Doctoral thesis, Faculty of Engineering, Lund University, Sweden
- Fast, M. & Palmé, T., (2010). Application of artificial neural network to the condition monitoring and diagnosis of a combined heat and power plant, *Journal of Energy*, vol. 35 (2), pp. 1114-1120
- Angeli, C. & Chatzinikolaou, A., (2004). On-Line fault detection techniques for technical systems: A survey, *International Journal of Computer Science & Applications*, Technomathematics Research Foundation, vol. I (1), pp. 12 – 30
- Russell, S. & Norvig, P., (1995). *Artificial intelligence: A modern approach*, New York, Prentice-Hall Inc.
- Saxena, B., Kumar, A., Srivastava, A. & Goel, A., (2011). Real time diagnostic prognostic solution for life cycle management of thermomechanical system, *IEEE Canadian Conference on Electrical and Computer Engineering* (pp. 999-1003), May 8-10, Niagara Falls, Canada
- Kumar, A., Saxena, B., Srivastava, A. & Goel, A., (2011). Physics based prognostic health management for thermal Barrier Coating System, *Proceeding.*

*Prognostic and Health Management Society Conference*, vol. 2 (6), September 25-29, Montreal, Canada

Michael J. A. Berry & Gordon Linoff, (1997). *Data Mining Techniques: For Marketing, Sales, and Customer Support, Database management*, New York, Computer Publishing Wiley

Boger, Z. & Guterman, H., (1997). Knowledge extraction from artificial neural network models, *Proceedings of the IEEE Conference on Systems, Man and Cybernetics* (pp. 3030 - 3035), October 12-15, Orlando, USA

Zhu, J., (2009). Marine diesel engine condition monitoring by use of BP neural network, *Proceedings of the International Multi Conference of Engineers and Computer Scientists*, Vol. II, March 18-20, Hong Kong

Kumara, K.P., Rao, K.V.N.S. , Krishnan, K.R.& Thejaa, B., (2012). Neural network based vibration analysis with novelty in data detection for a large gas turbine, *Shock and Vibration*, vol. 19, pp. 25–35

Patel, P. M. & Prajapati, J.M., (2011). A review on artificial intelligent system for bearing condition monitoring, *International journal of engineering science and technology*, vol. 3 (2), pp. 1520-1525

Samhoury, M. , Al-Ghandoor, A., Alhaj Ali, S., Hinti, I. & Massad, I., (2009). An intelligent machine condition monitoring system using time-based analysis: neuro-fuzzy versus neural network, *Jordan Journal of Mechanical and Industrial Engineering*, vol. 3 (4), pp. 294 – 305.

Riad, A.M, Elminir K.H., & Elattar, M. H., (2010). Evaluation of neural networks in the subject of prognostics as compared to linear regression model, *International Journal of Engineering & Technology IJET-IJENS*, vol: 10 (6), pp. 52 - 58

Specht, D. F. (1991). A general regression neural network, *IEEE Transactions on Neural Networks*, vol. 2 (6), pp. 568-577

Kaminski, M., (2010). General regression neural networks as rotor fault detectors of the induction motor, *IEEE International Conference on Industrial Technology (ICIT'10)* (pp. 1239 – 1244), March 14-17, Vina del Mar, Chile

**Alka Srivastava** has 24 years of research, administrative and industrial experience and has a BAsC degree in Electrical engineering from University of Ottawa. At present she is the manager of the R & D Division and leads several teams working in the fields of Fault Tolerance, Prognostic health management etc.

**Avisekh Banerjee, PhD, PEng**, is Senior Mechanical Engineer at Life Prediction Technologies Inc. (LPTi), Ottawa, ON. He is working on the development of a diagnostics and prognostics tools for turbo-machinery and avionics. His area of research and interests is diagnostics, prognostics and data trending for failure detection and development of PHM framework.

**Alok Goel** has more than 30 years of manufacturing, research, and industrial experience. He has a M.Sc degree in Advance Manufacturing Systems & Technology from University of Liverpool, UK. His research interests are in naval fuel filtration and lube oil separation. He is specialized in manufacturing graded natural fiber as a filler material for plastic reinforcement application.

## BIOGRAPHIES

**Dr. Amar Kumar** has more than 25 years of research and consulting experience in the fields of structural materials characterization and development, fracture mechanics, failure analysis and applications. Dr. Kumar is currently working as senior research scientist in the development projects of diagnostics, prognostics and health management of aeroengine components. He specializes in both data driven approaches and physics based modeling and simulations. Dr. Kumar has published more than 170 research papers in refereed journals, conference proceedings, and technical reports.

# Health Monitoring of a Power Supply Using Multivariate Regression

Leonardo Ramos Rodrigues<sup>1</sup>, João Paulo Pordeus Gomes<sup>1</sup>,  
Takashi Yoneyama<sup>2</sup> and Roberto Kawakami Harrop Galvão<sup>2</sup>

<sup>1</sup>EMBRAER S.A., São José dos Campos, São Paulo, 12227-901, Brazil

*leonardo.ramos@embraer.com.br*

*joao.pordeus@embraer.com.br*

<sup>2</sup>ITA – Instituto Tecnológico de Aeronáutica, São José dos Campos, São Paulo, 12228-900, Brazil

*takashi@ita.br*

*kawakami@ita.br*

## ABSTRACT

Due to the increasing use of electronics in critical aircraft control systems, it has become more and more important in the aerospace industry to understand how the degradation process of electronic devices occurs. Power supplies are devices of special interest since their internal components such as diodes, capacitors, MOSFETs (Metal Oxide Semiconductor Field Effect Transistor) and IGBTs (Insulated Gate Bipolar Transistors) operate under continuous stress conditions and often present elevated failure rates. The aim of this work is to present a methodology for detecting the gradual health degradation of a COTS (Commercial off-the-shelf) power supply. An accelerated aging process for power MOSFETs was conducted. During this experiment, power MOSFETs were subjected to thermal overstress in order to increase die-junction temperature above rated value through large current from drain to source. Multivariate regression analysis was applied to the raw data collected during the tests in order to assess the power supply health status.

## 1. INTRODUCTION

Health monitoring applications for electronic systems are significantly rarer compared to mechanical components. Some well known reasons are: complex architecture of electronics, interdependency of component functionality, and the lack of monitoring sensors (Kumar, Vichare, Dolev, & Pecht, 2011).

Despite the lower availability of electronic health monitoring applications, some work can be found regarding isolated components such as transistors (Patil, Das, Goebel & Pecht, 2008; Celaya, Saxena, Saha & Goebel, 2011), capacitors (Gu, Azarian & Pecht, 2008; Kulkarni, Biswas & Koutsoukos, 2009) and in some cases, integrated

components (Kumar et al., 2011; Zhang, Kwan, Xu, Vichare & Pecht, 2007).

Power supplies play an important role in modern electronic systems and their high failure rate turned these components into great candidates for health monitoring applications. The high level of integration and the lack of sensors turn this component into a challenging problem for PHM applications researchers.

The following work presents a power supply health monitoring application. A real power supply is aged in a testbed to generate degraded data. After that, a health monitoring methodology is proposed using multivariate regression.

This work is described in more details in the following sections. Section 2 presents the system under study. Section 3 presents the degraded data generation procedure. The proposed health monitoring methodology and its results are presented in section 4. In section 5, final conclusions are drawn.

## 2. SYTEM DESCRIPTION

For the present work a COTS power supply for commercial and military applications was used. The chosen power supply is produced by Century Electronics and is shown in Figure 1.

Rodrigues, L. R. et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

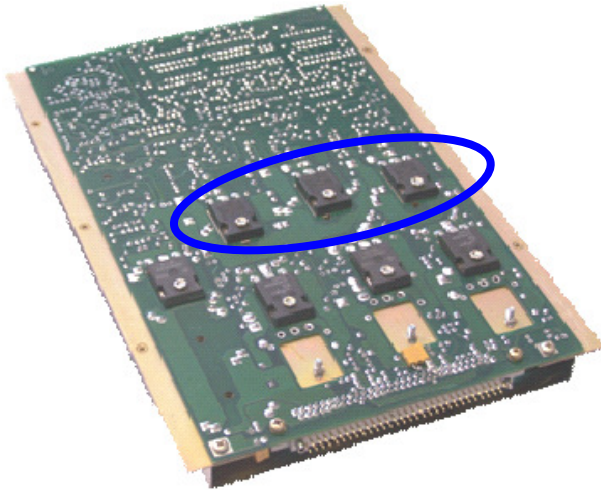


Figure 1. Power supply employed in the present work with indication of the power MOSFETs under consideration.

This is a 200W power supply with three outputs channels. Its input voltage ranges from 15 to 40Vdc with a nominal value of 28Vdc. The electrical specification of each output channel is presented in Table 1.

The chosen power supply was selected for this work due to its electrical and assembly features and typical applications. The power MOSFETs – highlighted in Figure 1 – are located in an accessible position and can be easily disconnected and reconnected. This characteristic is important to the accelerated aging MOSFET procedure used in this work. This procedure will be described in the next section.

Output Channel	Voltage (Vdc)	Current (A)
1	+5.1	8.0
2	+15.0	5.0
3	-15.0	5.5

Table 1. Output channels electrical specifications

### 3. DATA GENERATION

Many works using testbed experiments in order to collect data representing the component lifecycle have been published, like (Kappaganthu, Nataraj & Samanta, 2009) and (Saha & Goebel, 2009). In these works, an accelerated aging procedure is conducted to reduce experiment duration.

#### 3.1 Testbed

The testbed used in this work is composed by a power supply, a programmable DC supply, a programmable DC load and a data acquisition system. The testbed basic architecture is illustrated in Figure 2.

##### Programmable DC Supply

The programmable DC supply is responsible for providing power to the power supply under test according to a predefined profile. As mentioned earlier, the power supply used in the test has an input voltage range from 15 to 40Vdc with a nominal value of 28Vdc. During the experiment, the power supply was submitted to three different input voltage levels: low voltage (16Vdc), nominal voltage (28Vdc) and high voltage (40Vdc). Input voltage levels were defined based on the recommendations obtained from the power supply manufacturer. The input voltage profile is illustrated in Figure 3.

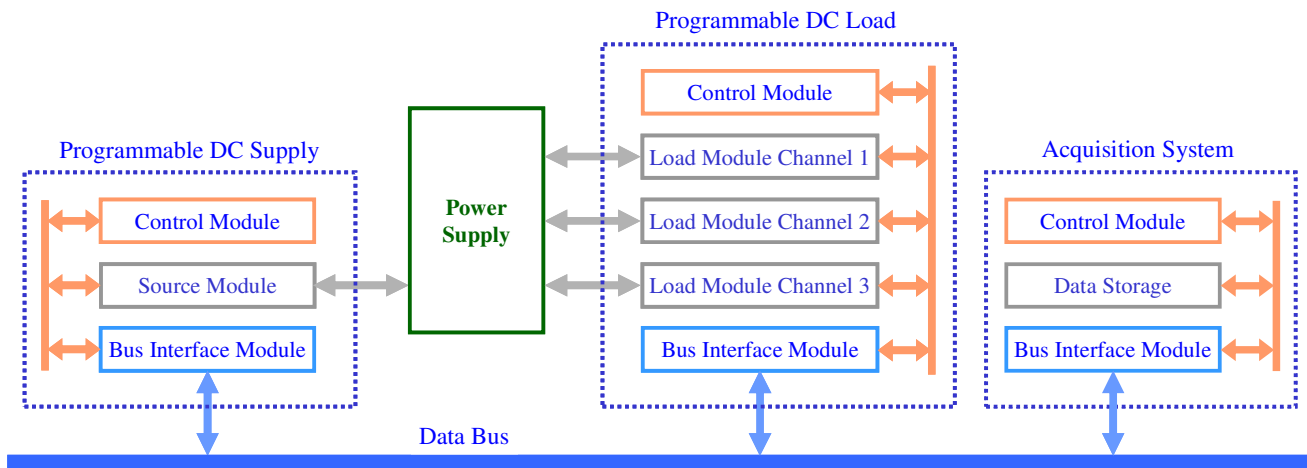


Figure 2. Testbed architecture



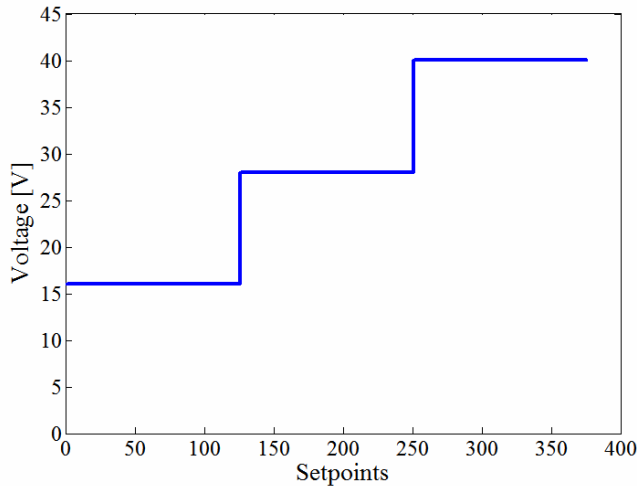


Figure 3. Input voltage profile

### Programmable DC Load

The programmable DC load emulates the desired DC resistance loads for each output channel. For each input voltage level considered in the test, a complete load cycling was performed. In the defined load cycling profile, each output current can assume 5 different values equally distant from zero to its nominal value. The load cycling sequence is composed by all possible combinations of output currents. Since there are three output channels and each channel can assume 5 different values, there are 125 possible output current combinations. Figure 4 shows the load cycling sequence.

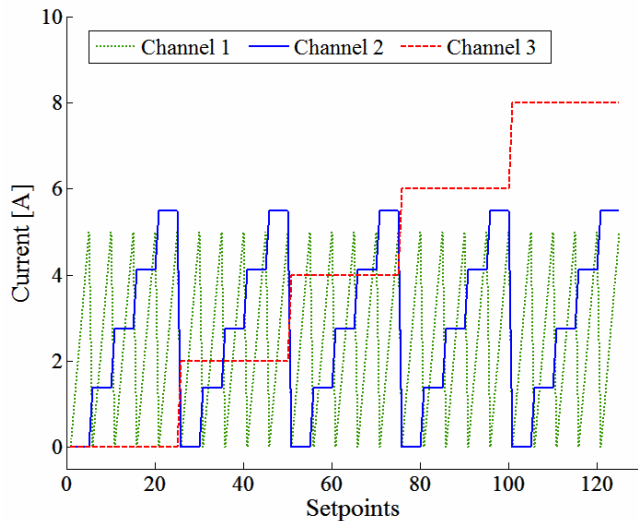


Figure 4. Load cycling sequence

### Data Acquisition System

The testbed comprises a data acquisition system that is responsible for recording the relevant signals during test execution. The variables recorded during the tests that will be used in the proposed health monitoring method are listed below. All variables were recorded with a sample rate of 1 Hz.

- Input voltage
- Input current.
- Output Voltage of each output channel.
- Output Current of each output channel.

### 3.2 MOSFETs Accelerated Aging Procedure

ALT (Accelerated Life Testing) and HALT (Highly Accelerated Life Testing) methodologies are frequently used to assess the reliability of products. These methodologies also play an important role in the development of PHM (Prognostics and Health Monitoring) solutions for electronics components and systems. The typical expected lifetime for electronic devices is in the order of thousands of hours. In such situations, it is not feasible to wait for devices to fail under normal operation in order to collect a dataset that is representative of the component lifetime (Celaya, Saxena, Wysocki, Saha & Goebel, 2010).

In this work, the power supply accelerated degradation process was obtained by the accelerated aging procedure of its power MOSFETs. There are four main semiconductor failure mechanisms that contribute to aging tendencies of MOSFET devices. These mechanisms are listed below (Kalgren, Baybutt, Ginart, Minnella, Roemer & Dabney, 2007):

- Thermal cycling
- Electro-migration
- Hot carrier injection effects
- Time-dependent dielectric breakdown

The accelerated aging process used in this work was based on the hot carrier injection effects principle. A more detailed description of each MOSFET main failure mechanism is provided by Kalgren et al. (2007).

### 3.3 Test Sequence

The power supply test campaign is composed by two main phases: Healthy Testing and Degraded Testing.

### Healthy Testing

A healthy test was performed in order to characterize the baseline response of the power supply. During the healthy testing, no MOSFET aging procedure had been conducted. All MOSFETs had nominal operational condition.

The main goal of running a healthy test is to collect datasets prior to any seeded fault insertion. These will be the datasets upon which all seeded fault datasets can be compared with to determine the performance effects of environmental conditions and MOSFET degradation.

### Degraded Testing

After performing a series of healthy tests, a healthy MOSFET was replaced by a degraded one. It was decided to insert only one degraded MOSFET within the power supply. The output channel 2 MOSFET was chosen to be replaced since channel 2 is the one with the highest nominal output power.

The purpose of performing a seeded fault testing is to collect datasets after a degraded MOSFET has been inserted within the power supply. System level effects caused by the insertion of a degraded component can be investigated and a comparison between a healthy power supply and a degraded one can be made.

## 4. HEALTH MONITORING METHODOLOGY

The insertion of a degraded MOSFET in a channel is supposed to increase its internal resistance. The main idea behind the proposed method is to estimate the internal resistance for each channel and check for variations as the MOSFET is degraded.

For the proposed method, it is assumed that only currents and voltages for each channel are available. With this information, it is possible to equate input and output power for each channel, as described in Eq. (1):

$$V_{IN} \cdot I_{IN} = V_{OUT} \cdot I_{OUT} + T_L \quad (1)$$

where  $V_{IN}$  is the input voltage,  $I_{IN}$  is the input current,  $V_{OUT}$  is the output voltage,  $I_{OUT}$  is the output current and  $T_L$  are the total losses involved in the circuit.

$T_L$  can be expanded, resulting in the expression shown in Eq. (2):

$$P_{IN} - P_{OUT} = R_1 \cdot I_1^2 + R_2 \cdot I_2^2 + R_3 \cdot I_3^2 + L \quad (2)$$

where  $P_{IN}$  is the input power,  $P_{OUT}$  is the output power,  $R_1$ ,  $R_2$  and  $R_3$  are the equivalent internal resistances for each output channel,  $I_1$ ,  $I_2$  and  $I_3$  are the output currents for each output channel and  $L$  is an independent loss. Equation (2) may be rewritten in a vector form, resulting in:

$$[P_{IN} - P_{OUT}] = [R_1 \quad R_2 \quad R_3 \quad L] \begin{bmatrix} I_1^2 \\ I_2^2 \\ I_3^2 \\ 1 \end{bmatrix} \quad (3)$$

Internal resistances can then be estimated using least squares regression according to Eq. (4).

$$\begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ L \end{bmatrix}^T = \left( \begin{bmatrix} I_1^2 \\ I_2^2 \\ I_3^2 \\ 1 \end{bmatrix}^T \cdot \begin{bmatrix} I_1^2 \\ I_2^2 \\ I_3^2 \\ 1 \end{bmatrix} \right)^{-1} \cdot \begin{bmatrix} I_1^2 \\ I_2^2 \\ I_3^2 \\ 1 \end{bmatrix}^T \cdot (P_{IN} - P_{OUT}) \quad (4)$$

### 4.1 Results

The internal resistance of each output channel as well as the independent losses estimated according to Eq. (4) is presented in Figure 5. An estimation of the internal resistance of each channel as well as the independent loss is made for each test run. During a test run, the power supply is submitted to a complete input voltage cycling (as shown in Figure 3) and load cycling (as shown in Figure 4).

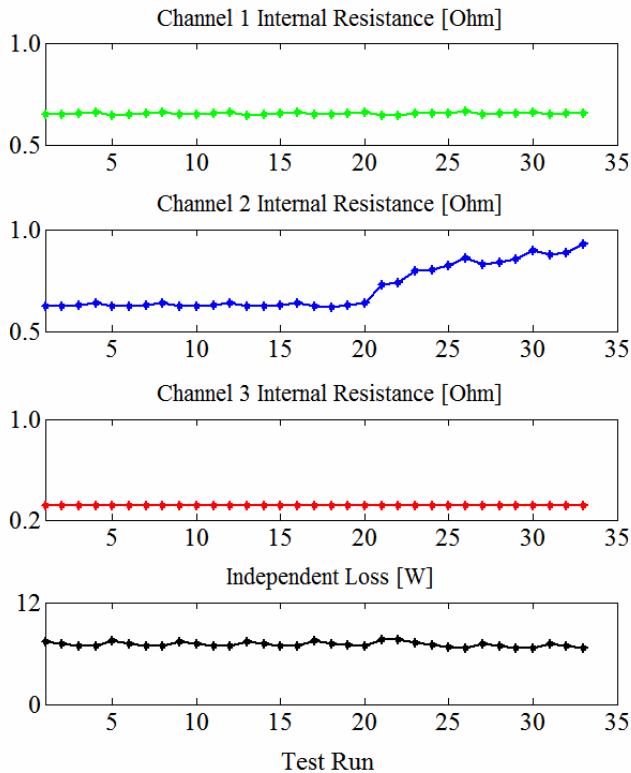


Figure 5. Estimation results

It can be noticed that the estimated internal resistance values of all three output channels are stable during the healthy stage of power supply lifetime. But when a degraded component is inserted into the power supply (approximately at the 20th test run) an increase in the estimated value of the output channel 2 internal resistance can be observed. It is important to remember that the fault was inserted only in the output channel 2 MOSFET. No changes were made in other two MOSFETs.

The multivariate regression analysis showed that the power supply was operating in a faulty condition and correctly identified that the degraded component corresponded to channel 2.

The last point of each curve showed in Figure 5 was calculated using the last dataset collected before the actual failure of the power supply.

## 5. CONCLUSIONS

This paper presented a methodology for detecting the gradual health degradation of a COTS power supply using multivariate regression.

The proposed method was capable of distinguishing a healthy power supply from a degraded one. Also, the results obtained showed that it is possible to monitor the health condition of electronic devices using a low bandwidth signal.

Many of the efforts to monitor the health condition of electronic devices use high bandwidth signals. The reason for that is that electronic devices failure modes, once present, tends to lead the device to a failure condition in a shorter period of time, when compared to other types of failure modes such as hydraulic or mechanical. However, the cost of recording and transmitting signals using high sample rates in aerospace industry is very high. This fact confirms the potential advantages of the proposed methodology.

## ACKNOWLEDGMENT

The authors acknowledge the support of FINEP (Financiadora de Estudos e Projetos - Brazil), CNPq (research fellowship) and FAPESP (grant 2011/17610-0).

## REFERENCES

- Celaya, J. R., Saxena, A., Saha, S. & Goebel, K. F. (2011). Prognostics of Power MOSFETs under Thermal Stress Accelerated Aging Using Data-Driven and Model-Based Methodologies. In *Proceedings of International Conference on Prognostics and Health Management, Montreal*.
- Celaya, J. R., Saxena, A., Wysocki, P., Saha, S. & Goebel, K. (2010). Towards Prognostics of Power MOSFETs: Accelerated Aging and Precursors of Failure. In *Proceedings of International Conference on Prognostics and Health Management, Portland*.
- Gu, J., Azarian, M. H. & Pecht, M. G. (2008). Failure Prognostics of Multilayer Ceramic Capacitors in Temperature-Humidity-Bias Conditions. In *Proceedings of International Conference on Prognostics and Health Management, Denver*.
- Kalgren, P. W., Baybutt, M., Ginart, A., Minnella, C., Roemer, M. J. & Dabney, T. (2007). Application of Prognostic Health Management in Digital Electronic Systems. In *Proceedings of IEEE Aerospace Conference, Big Sky*.
- Kappaganthu, K., Nataraj, C. & Samanta, B. (2009). Model Based Bearing Fault Detection Using Support Vector Machines. In *Proceedings of International Conference on Prognostics and Health Management, San Diego*.
- Kulkarni, C. S., Biswas, G. & Koutsoukos, X. (2009). A Prognosis Case Study for Electrolytic Capacitor Degradation in DC-DC Converters. In *Proceedings of International Conference on Prognostics and Health Management, San Diego*.

- Kumar, S., Vichare, N. M., Dolev, E. & Pecht, M. (2011). A Health Indicator Method for Degradation Detection of Electronic Products. *Microelectronics Reliability*, Volume 52, Issue 2.
- Patil, N., Das, D., Goebel, K. & Pecht, M. (2008). Identification of Failure Precursor Parameters for Insulated Gate Bipolar Transistors (IGBTs). In *Proceedings of International Conference on Prognostics and Health Management, Denver*.
- Saha, B. & Goebel, K. (2009). Modeling Li-ion Battery Capacity Depletion in a Particle Filtering Framework. In *Proceedings of International Conference on Prognostics and Health Management, San Diego*.
- Zhang, G., Kwan C., Xu, R., Vichare, N. & Pecht, M. (2007). An Enhanced Prognostic Model for Intermittent Failures in Digital Electronics. In *Proceedings of IEEE Aerospace Conference, Big Sky*.

#### BIOGRAPHIES



**Leonardo Ramos Rodrigues** holds a bachelor's degree in Electrical Engineering from Universidade Federal do Espírito Santo (UFES, 2003), Brazil, and a Master Degree in Aeronautical Engineering from Instituto Tecnológico de Aeronáutica (ITA, 2008), São José dos Campos, São Paulo, Brazil. He is currently

pursuing his doctorate in Aeronautical Engineering at ITA. He is with EMBRAER S.A., São José dos Campos, São Paulo, Brazil, since 2006. He works as a Development Engineer in an R&T group at EMBRAER performing research on PHM technology for application to aeronautical systems. His current research interests are the application of health monitoring techniques for electronic components and the usage of PHM information for inventory optimization.



**João Paulo Pordeus Gomes** holds a bachelor's degree on Electrical Engineering from Universidade Federal do Ceará (UFC, 2004), Brazil, master's (2006) degree on aeronautical Engineering and doctorate's (2011) degree in electronics engineering from Instituto Tecnológico de Aeronáutica (ITA), São José dos Campos, SP,

Brazil. He is with EMBRAER S.A., São José dos Campos, SP, Brazil, since 2006. He works as a Development Engineer at EMBRAER focusing on PHM technology applications on aeronautical systems.



**Takashi Yoneyama** is a Professor of Control Theory with the Electronic Engineering Department of ITA. He received the bachelor's degree in electronic engineering from Instituto Tecnológico de Aeronáutica (ITA), Brazil, the M.D. degree in medicine from Universidade de Taubaté, Brazil, and the Ph.D. degree in electrical engineering from the University of London, U.K. (1983). He has more than 250 published papers, has written four books, and has supervised more than 50 theses. His research is concerned mainly with stochastic optimal control theory. Prof. Yoneyama served as the President of the Brazilian Automatics Society in the period 2004-2006.



**Roberto Kawakami Harrop Galvão** is an Associate Professor of Systems and Control at the Electronic Engineering Department of ITA. He holds a bachelor's degree in Electronic Engineering (Summa cum Laude, 1995) from Instituto Tecnológico de Aeronáutica (ITA), Brazil. He also obtained the master's

(1997) and doctorate (1999) degrees in Systems and Control from the same institution. Since 1998 he has been with the Electronic Engineering Department of ITA as a full-time academic. Dr. Galvão is a Senior Member of the IEEE and an Associate Member of the Brazilian Academy of Sciences. He has published more than 200 papers in peer-reviewed journals and conference proceedings. His main areas of interest are fault diagnosis and prognosis, wavelet theory and applications, and model predictive control.

# Failure Prognostics of a Hydraulic Pump Using Kalman Filter

João Paulo P. Gomes<sup>1</sup>, Bruno P. Leão<sup>2</sup>, Wlamir O. L. Vianna<sup>1</sup>, Roberto K. H. Galvão<sup>2</sup> and Takashi Yoneyama<sup>2</sup>

<sup>1</sup> *EMBRAER S.A., São José dos Campos, São Paulo, 12227-901, Brazil*  
*joao.pordeus@embraer.com.br*  
*wlamir.vianna@embraer.com.br*

<sup>2</sup> *ITA – Instituto Tecnológico de Aeronáutica, São José dos Campos, São Paulo, 12228-900, Brazil*  
*bruno.leao@gmail.com*  
*kawakami@ita.br*  
*takashi@ita.br*

## ABSTRACT

Hydraulic systems are an important power source in modern aircraft. Most aircraft employ hydraulic power for flight control systems and landing gears actuation. Pumps are a critical component in hydraulic system and monitoring the health of such components may provide economic and operational benefits to aircraft operators. This work describes the use of Kalman Filter techniques for the estimation of remaining useful life of aircraft hydraulic pumps. An empirical model of degradation evolution is employed for this purpose. Low sampling rate measurements of the hydraulic pressure of the aircraft hydraulic systems are the only measurements employed. In order to illustrate and validate the method, two time series of actual run to failure data are analyzed. Results provide evidence that the method can be successfully employed for actual aircraft hydraulic pump failure prognosis.

## 1. INTRODUCTION

Hydraulic power is widely employed in aircraft systems, mainly in the actuation of flight control surfaces and landing gears. One important reason for this is the fact that hydraulic actuators present a high power density, i.e. compact and light actuators may withstand large loads (Merrit, 1976). This makes hydraulic power very suitable for aircraft systems applications. Therefore, hydraulic systems comprise an important power source in modern aircraft.

Although redundancies are employed in the design of aircraft systems in order to guarantee high airworthiness levels, the failure of a hydraulic system commonly represents adverse economic and operational consequences. Therefore, the application of prognosis techniques to avoid failures in hydraulic system components can potentially

provide benefits to aircraft operators. Pumps are the heart of a hydraulic system, since they are the components which transform mechanical power from the engine gearbox or from an electric motor into hydraulic power. This work describes an innovative application of PHM techniques in the estimation of remaining useful life (RUL) of hydraulic pumps in aircraft hydraulic systems. In the proposed solution, low sampling rate measurements of the hydraulic pressure of the hydraulic system fed by the hydraulic pump are the only source of information for the estimation of component health. This kind of measurement is commonly available in modern aircraft. The Kalman Filter (KF) technique (Kalman, 1960) is employed for estimating current degradation and its trend, based on an empirical model of degradation evolution. Such estimates are then extrapolated for performing failure prognosis.

There are previous works in the literature which deal with failure prognosis of hydraulic pumps. For industrial pumps, various works such as (Hancock and Zhang, 2006) present solution. This specific solution is based on vibration measurements which are analyzed using wavelet and neuro-fuzzy techniques. Concerning aircraft hydraulic pumps, (Byington, Watson, Edwards, and Dunkin, 2003) describe the application of neuro-fuzzy classifiers and a Bayesian belief formulation in order to obtain failure prognosis. In this case, measurements of pump pressure, case drain flow and case drain temperature were employed. (Bechhoefer, Clark and He, 2010) presents the use of Kalman Filter for performing prognosis of this kind of component. However, in such work, the condition indicator (CI) is based on vibration measurements, taken from dedicated sensors. The proposed method differs from such technique due to the fact that in the former only system hydraulic pressure measurements are employed, with low sampling rates. Such measurement are used for control and safety purposes and are therefore usually available in modern aircraft with no need for additional sensors or recording capability.

The remaining sections are organized as follows: section 2 describes the hydraulic pump considered in this work;

Gomes et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

section 3 presents the failure prognosis technique employed; section 4 presents the results and section 5 is the conclusion.

## 2. COMPONENT DESCRIPTION

The component under study is a variable displacement axial piston pump with pressure compensation system to deliver constant discharge pressure of 3.000 psi.

The pressure compensation system comprises a piston and valve that controls the swash plate angle. Increased angles provide higher flows while lower angles provide lower flows. If discharge pressure decreases, the compensation piston actuates decreasing the swash plate angle and the discharge flow. This brings the discharge pressure back to nominal values. The compensation system is illustrated in Figure 1.

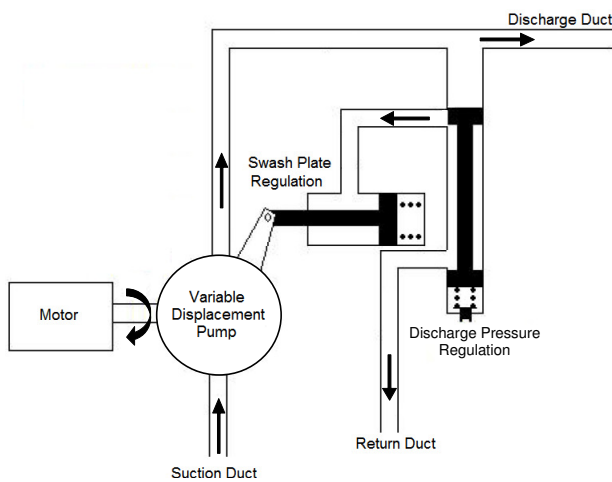


Figure 1. Pressure compensation system schematic

This component contains a spring actuated hydraulic cylinder controlled by a pressure regulator valve. This valve contains a spring whose pre load calibration defines the pump discharge pressure.

One common failure of this type of pump is the looseness of the regulator valve spring which decreases the pump discharge pressure. This failure mode can be sensed by pressure transducer data acquired at the pump discharge duct at no load condition. For the analysis presented in this paper, hydraulic discharge pressure data was acquired for different aircraft operating at different conditions for several flights.

## 3. FAILURE PROGNOSTICS METHODOLOGY

As presented in section 2 the failure mode considered in this work can lead to a reduction in the pressure delivered by the pump. This fact is well known and a failure monitor is commonly available to detect when pressure drops below a given threshold for a certain period of time. Although this

failure detection system is usually available for aircraft hydraulic systems, failure prognosis is not currently found in commercial systems.

The proposed failure prognostics system uses the average of measured system pressure as a CI. The idea is to track this average pressure and to predict when it will reach a level that triggers the designed failure monitor.

Data employed in the present analysis correspond to field data collected from two aircraft that presented pump failures. Data ranging from pumps in normal condition up to failure monitor triggering are available for both aircraft. Figure 2 and Figure 3 show normalized hydraulic pump CIs calculated for these two aircraft.

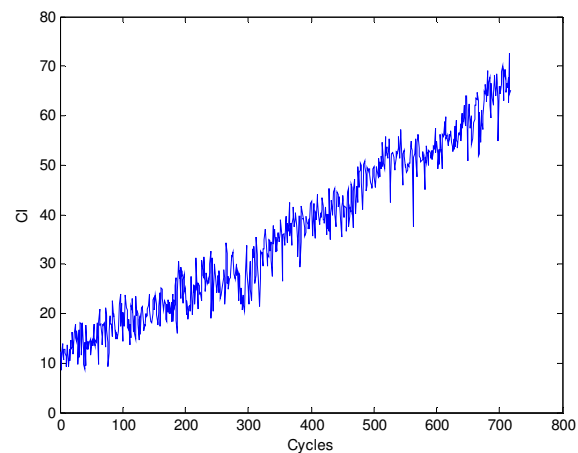


Figure 2. CI for Pump 1

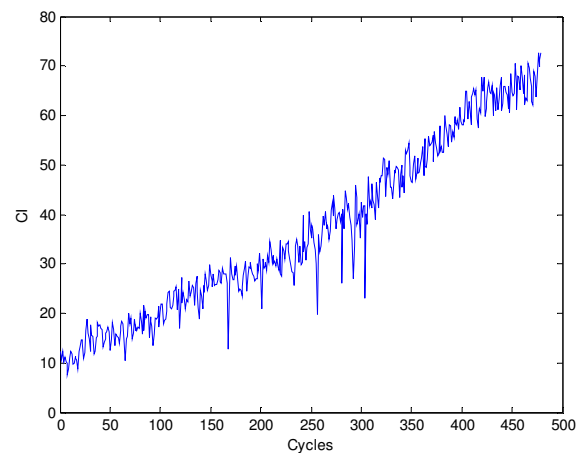


Figure 3. CI for Pump 2

For failure prognostics implementation, a Kalman filter was employed. Concerning the dynamic model necessary for filtering and extrapolation, no first principles model was used. The state space representation of a linear degradation evolution with unknown slope was used for this purpose.

This model was empirically chosen based on the aspect of the CI plots. It may be noticed that the degradation rate is not constant for different pumps. This may be the case even for the same pump under different operating conditions. Therefore, it is necessary to estimate the slope – which describes the degradation evolution trend – along with the degradation estimation itself. The resulting model may be defined as follows:

$$\begin{aligned} d_{k+1} &= a_k + d_k + v_{1k} \\ a_{k+1} &= a_k + v_{2k} \\ CI_k &= d_k + w_k \end{aligned} \quad (1)$$

where  $d$  is the estimated degradation,  $a$  is the slope,  $v_1$ ,  $v_2$  and  $w$  are gaussian noises,  $CI$  is the condition indicator and  $k$  is the discrete time instant. In this case,  $k$  represents aircraft cycles. State noise  $v_1$  and observation noise  $w$  represent actual state and observation noises present in the data, while  $v_2$  is an artificial noise added for the estimation of the fixed parameter  $a$ . A noise adaptation scheme to adapt  $v_2$  variance during filtering (Leão, 2011). In Kalman filter, the information concerning the variance of the parameter estimates at instant  $k$  is contained in the covariance matrix  $P_k$ . Using this information, the variance  $\sigma_{v_2k}^2$  is obtained by :

$$\sigma_{v_2k}^2 = \left( -1 + \frac{1}{\lambda} \right) P_k \quad (2)$$

Using the  $d$  and  $a$  distributions estimated at a given instant and the model presented in Eq. (1), Monte Carlo simulations were performed until  $d$  reaches a failure threshold. Failure thresholds were chosen according to the concept of Hazard Zone (HZ) (Orchard and Vachtsevanos, 2009).

The HZ defines a region, modeled by a bounded distribution, with high probability of failure occurrence. In this work, failure thresholds were sampled according to the chosen HZ distribution. The HZ was defined as a normal distribution with mean of 70 and standard deviation of 3.

The HZ definition was performed empirically. A more systematic approach to choose it could only be developed as more run-to-failure time series are available.

#### 4. RESULTS

The proposed method was tested using the two run-to-failure datasets corresponding to the two mentioned hydraulic pumps. For each pump, five predictions were made in different time-to-failure (TTF) situations. Chosen instants were 250, 200, 150, 100 and 50 cycles before failure. Figure 4 and Figure 5 show the result for each pump. The plots present the real and estimated remaining useful life (RUL) for each pump. Bars indicating the uncertainty of the corresponding RUL estimate were also presented.

In both cases, uncertainty bars were defined as an interval corresponding to 95% of the predicted distribution

Figures 4 and 5 show promising results: in all prognostics attempts, the TTF value was inside the predicted distribution. Another desirable characteristic is the fact that uncertainty in RUL estimates was reduced as TTF reduces.

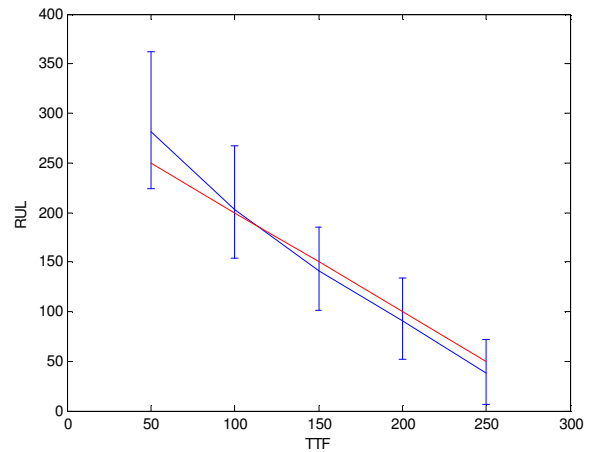


Figure 4. Prognostics for Pump 1

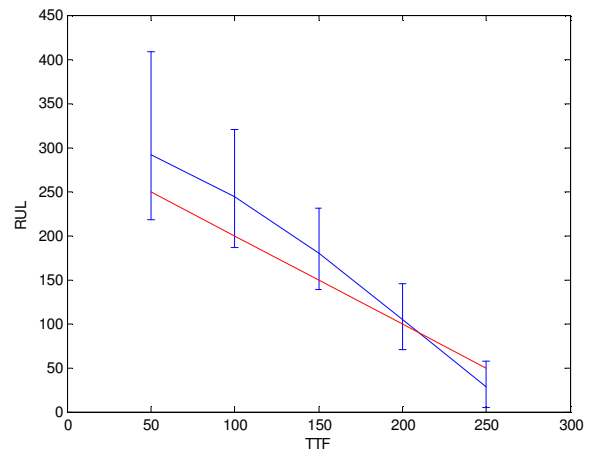


Figure 5. Prognostics for Pump 2

#### 5. CONCLUSIONS

The present work showed a method for aircraft hydraulic pump failure prognostics using Kalman Filter. First principles were not used to model the degradation evolution. A linear model with uncertain slope was empirically employed for this purpose. All calculations were based on a

low bandwidth measurement of the hydraulic pressure in the system fed by the hydraulic pump.

Despite the simplicity and wide applicability of the method, results obtained from its application to two run-to-failure time series of actual aircraft hydraulic pumps showed to be promising. More conclusive results may be obtained as long as more run-to-failure time series for hydraulic pumps become available.

#### ACKNOWLEDGMENT

This work was partially supported by Financiadora de Estudos e Projetos (FINEP), FAPESP (grant 2011/17610-0) and CNPq (research fellowship), Brazil

#### REFERENCES

- Merrit, E. H. (1976). Hydraulic Control Systems. John Wiley, New York.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. Transactions of the ASME - Journal of Basic Engineering, v.82, Series D, p.35-45.
- Hancock, K. M.; Zhang, Q. (2006). A hybrid approach to hydraulic vane pump condition monitoring and fault detection. Transactions of the American Society of Agricultural and Biological Engineers, v. 49 (4), p. 1203-1211.
- Byington, C. S.; Watson, M.; Edwards, D. and Dunkin, B. (2003). In-line health monitoring system for hydraulic pumps and motors. IEEE Aerospace Conference Proceedings, Big Sky, MO.
- Bechhoefer, E.; Clark, S. and He, D. (2010). A state space model for vibration based prognostics. Proceedings of the Annual Conference of the Prognostics and Health Management Society.
- Leão, B. P. (2011). Failure prognosis methods and offline performance evaluation. Ph.D. thesis. Instituto Tecnológico de Aeronáutica.
- Orchard, M; Vachtsevanos, G. (2009) A particle filtering approach for online fault diagnosis and failure prognosis. Transactions of the Institute of Measurement and Control, no. 3-4, p. 221-246.

#### BIOGRAPHIES



**João Paulo Pordeus Gomes** holds a bachelor's degree on Electrical Engineering from Universidade Federal do Ceará (UFC, 2004), Brazil, master's (2006) degree on aeronautical Engineering and doctorate's (2011) degree in electronics engineering from Instituto Tecnológico de Aeronáutica (ITA), São José dos Campos, SP,

Brazil. He is with EMBRAER S.A., São José dos Campos, SP, Brazil, since 2006. He works as a Development Engineer at EMBRAER focusing on PHM technology applications for aeronautical systems.



**Bruno P. Leão** holds a bachelor's degree on Control and Automation Engineering (2004) from Universidade Federal de Minas Gerais (UFMG), Brazil, a master's degree on Aeronautical Engineering (2007) from Instituto Tecnológico de Aeronáutica (ITA), Brazil, and a D.Sc. degree on Electronics Engineering and Computer

Science (2011) also from ITA. He is currently a Lead Scientist with GE Global Research at the Brazil Technology Center in Rio de Janeiro, where he performs research in the field of Aviation Systems. He was formerly with Embraer S.A. in Brazil from 2005 to 2012. During five years he has been with the PHM research group at Embraer researching PHM solutions for aircraft systems. Before that, he has worked as a Systems Engineer on the areas of Flight Controls and Automatic Flight Controls.



**Wlamir Olivares Loesch Vianna** holds a bachelor's degree on Mechanical Engineering (2005) from Universidade de São Paulo (USP), Brazil, and Master Degree on Aeronautical Engineering (2007) from Instituto Tecnológico de Aeronáutica (ITA), Brazil. He is with Empresa Brasileira de Aeronáutica S.A (EMBRAER), São José dos

Campos, SP, Brazil, since 2007. He works as a Development Engineer of a R&T group at EMBRAER focused on PHM technology applications in aeronautical systems



**Roberto Kawakami Harrop Galvão** is an Associate Professor of Systems and Control at the Electronic Engineering Department of ITA. He holds a bachelor's degree in Electronic Engineering (Summa cum Laude, 1995) from Instituto Tecnológico de Aeronáutica (ITA), Brazil. He also obtained the master's

(1997) and doctorate (1999) degrees in Systems and Control from the same institution. Since 1998 he has been with the Electronic Engineering Department of ITA as a full-time academic. Dr. Galvão is a Senior Member of the IEEE and an Associate Member of the Brazilian Academy of Sciences. He has published more than 200 papers in peer-reviewed journals and conference proceedings. His main



areas of interest are fault diagnosis and prognosis, wavelet theory and applications, and model predictive control.



**Takashi Yoneyama** is a Professor of Control Theory with the Electronic Engineering Department of ITA. He received the bachelor's degree in electronic engineering from Instituto Tecnológico de Aeronáutica (ITA), Brazil, the M.D. degree in medicine from Universidade de Taubaté, Brazil, and the Ph.D. degree in electrical engineering from the University of London, U.K. (1983). He has more than 250 published papers, has written four books, and has supervised more than 50 theses. His research is concerned mainly with stochastic optimal control theory. Prof. Yoneyama served as the President of the Brazilian Automatics Society in the period 2004-2006.

# Fleet Wide Asset Monitoring: Sensory Data to Signal Processing to Prognostics

Preston Johnson

*National Instruments, Austin, Texas, 78759, USA  
preston.johnson@ni.com*

## ABSTRACT

Next generation fleet wide asset monitoring solutions are incorporating machine failure prediction and prognostics technologies. These technologies build on signal processing of vibration time waveforms, process parameters, and operating conditions of the machine. For prognostics algorithms to work well, the signal processing algorithms need to be applied correctly and the results need to be reliable. This paper provides a survey of signal processing techniques as applied to specific machine component with a focus on the output and use with prognostics technologies. With properly organized outputs, prognostics algorithms transform the fleet condition and health management challenge into a deployable fleet health management solution. To arrive at the deployable fleet management solution, a systematic approach in the design of the prognostics system is preferable. This approach includes data and model driven failure patterns, sensory data connectivity from deployed assets, prognostics analytical applications, and advisory generation outputs which guide the asset owners and maintainers.

## 1. INTRODUCTION

As costs decline to collect sensory data from industrial assets, it is more practical than before to implement an asset health management system for critical and balance of plant assets. Sensory data is available from supervisory control systems, and from low cost embedded data acquisition systems supporting specialized surveillance such as vibration or electrical power monitoring. To transform this abundance of data into actionable scheduling and maintenance activities, a systematic approach in design and

implementation of a prognostics solution is recommended (Lee, 2009).

There are several steps to consider when implementing a Fleet-wide health management system, Figure 1. The first is to identify the assets within the fleet for which a business case exists that justifies the expense of gathering, analyzing, and advising operations and maintenance. There are many sources of business benefit including uptime impacts on revenue, safety of workers, productivity, or even improvements in asset design (Hollingshaus, 2011). In the case of power generation plants, assets selected for advanced monitoring and prognostics include circulator water feed pumps, coal pulverizers, gas turbines, steam turbines, generators, and transformers.

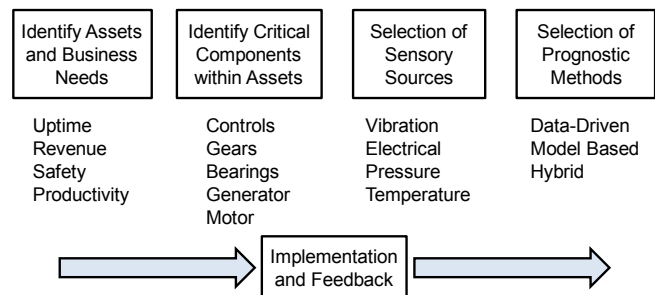


Figure 1. Four steps in design of prognostics systems

A second step in the design and implementation of a fleet-wide health management system is selection of critical components within an asset class that impact the ability of the asset to perform its function to acceptable standards. A typical methodology is the Failure Mode and Effects Analysis (FMEA) and Failure Modes, Effects, and Criticality Analysis (FMECA). In several industries, the process is formalized and includes published standards (Reliability 2004).

For a given asset, there may be several components whose failure will prevent the asset from performing its function.

Preston Johnson. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Further, within these components, some may warrant automated surveillance, while others may warrant periodic replacement. Relating failure modes to business benefit helps to refine the selection of the components for application of a monitoring and prognostic application.

The third step in the implementation process is sensor selection. Sensor selection builds on selected asset component, expected failure and degradation modes, and availability of proven sensory data interpretation algorithms. Selection of sensors and sensory data is impacted by availability of built-in sensor data from existing systems, cost of additional sensors and installation, and impact of data storage requirements. Further, some experimentation may be necessary to fully determine whether specific sensory data and analytics lead to information that is useful in predicting machinery component failure (Lei, 2004).

A fourth step in the implementation process is selection of the prognostic method for the asset class and business environment. Data driven methods require historical operational data to use as comparison using statistics and probability functions to derive estimates and predictions of health and reliability for a given asset. (CALCE, 2012), (NASA, 2012). Even if failure data patterns are not available, data driven methods can be used to compare current machinery surveillance data with historical normal operation. Any deviation from the normal, can be considered an anomaly and worthy of additional study by subject matter experts. Given a degradation is detected, a new pattern can be added to the collection of fault signatures for future use.

Physics driven prognostics often involve a model, or accepted standards for surveillance monitoring outputs. The Physics-of-failure (PoF) approach relies on knowledge of the assets life cycle and the impact of loading, operational conditions, geometry, materials, and failure mechanisms. For example, there are a number of standards for interpretation of vibration signatures including acceptable vibration levels for specific machine components and classes of assets. A bearing vibration analysis incorporates geometries of the bearing, speed of the machine (operational condition), loading, alignment of the shaft, and perhaps the L10 design life of the bearing. And of course, both data drive and physics driven methods can be combined to form a hybrid approach to fleet-wide asset monitoring applications.

Finally, most fleet-wide implementations begin with a selected few assets in the fleet. With an initial deployment step, costs can be contained and the deployment strategy validated. Many questions or challenges are investigated during this pilot phase of implementation. These include the ability to make sensory measurements under consistent conditions, ability to reduce sensory data using embedded computations, and the ability of analytics on collected and historical data to predict patterns and rates of degradation.

This paper expands on each of these steps, and introduces a specific case study in pilot phase implementation.

## 2. SIX CLASSES OF MACHINES AND ASSETS

There are many parameters to evaluate in determining whether a collection of assets deserve monitoring for degradation and automatic processing of degradation indicators. It is up to the owning organization to determine whether financial, safety, or environmental merits exists to justify an expense of condition monitoring and prognostics. The FMECA methodology mentioned earlier serves as a model for making these evaluations. Given merits for monitoring and prognostics, it is desirable classify the asset and identify critical components within the asset family.

There are many types of machines and assets. Assets can often be grouped into a class of machinery with similar condition monitoring techniques, sensory uses, and recommended condition monitoring practices. In beginning the implementation process, it is useful to categorize the assets into one of the following classes, Figure 2.



Figure 2. Six common classes of machines

For each class of machine, there are specific commercial and experimental techniques for condition monitoring that offer methods for predicting mechanical and functional degradation. For example, motor driven machines may incorporate electrical power sensors and signature analysis. Moving machines may require special load and speed sensors to organize sensory data into operational cycles. Each class of machine brings with it a traditional approach of condition based maintenance and specific and accepted sensors and signal processing techniques.

Within each asset, there are multiple critical components common to mechanical function of the machine. In rotating machinery for example, several component failure modes are common, Figure 3.

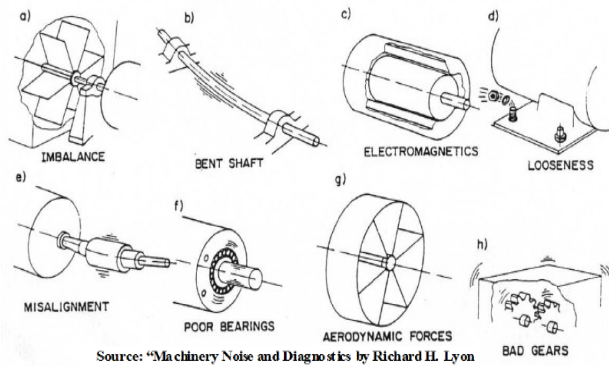


Figure 3. Common mechanical failure sources

With understanding of the asset type within the fleet, it is prudent to identify the key components whose failure directly or significantly impacts the function of the asset. The classes of assets and common mechanical failures are widely used categories for breaking down the asset monitoring and prognostics problem in rotating machinery.

Since most industrial assets have rotating components, it is appropriate to consider common sensors and analytics used to derive parameters that can indicate machinery degradation and failure.

**3. SENSOR SELECTION AND ANALYTICS**

There are many sensors available for monitoring and control of machinery assets. Many exist in the machine as a control related sensor, while others are added to the industrial asset for performance or mechanical health indicators. Common sensor types are shown in Table 1.

Sensor	Graphic	Use
Temperature		Heat as friction indication
Flow		Flow of fluids or gas
Speed		Rotational speed
Acceleration		Vibration
Displacement		Shaft movement
Pressure		Pressure (cylinder)
Electrical Power		Motor Current / Machine Load

Table 1. Common sensors use for asset monitoring

Other sensory information reported from the control system may include error codes, torque, cycle step, and so on. These control system parameters are often useful in correlating the machine’s work and operating condition with measurements from the common sensors in Table 1. It is important to sort measurements into operating modes or

regimes to improve correlation of on-line measurements to historical data patterns.

Analysis of sensory data allows the fleet asset monitoring system to transform data into information useful in determining amount and pace of degradation, and therefore in predicting a failure of the asset to perform its intended function. The output of analysis algorithms reduces the raw sensory data into features which describe the original measurement. These features or descriptors are the numeric inputs which prognostic algorithms use to perform association of an asset’s current state of health with historical machine health patterns, or models of machinery health. Figure 4 offers several analytic techniques and the feature results these analysis techniques may produce.

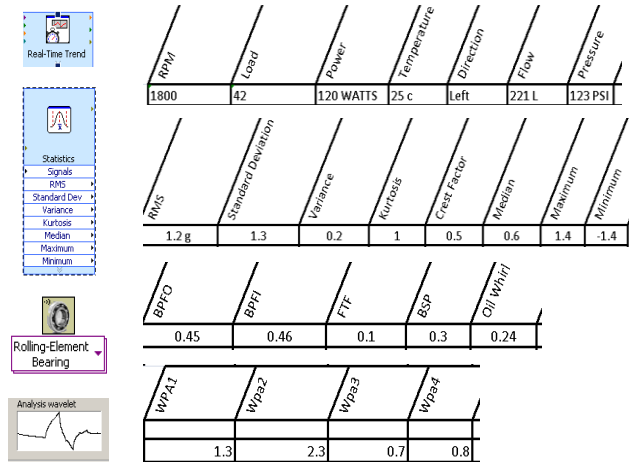


Figure 4. Analysis with feature and numeric results

In Figure 4, the top row lists a series of time series trend analysis that yield averages, rates of change, and current values. The second row lists several statistical measures of a time series vector or trend which indicate shape and distribution of a series of sequential measurements from a single sensor. In the case of a roller bearing, frequency analysis of a high sample rate vibration snapshot can reduce the sensory data to characteristic fault frequency amplitudes indicative of defects in the roller bearing. An advanced analytical technique, wavelet analysis, reduces a high sample rate snapshot from a dynamic sensor to wavelet packet coefficients which indicate presence of transient phenomenon in the measured signals. Transients may be indicative of impacts in the case of a roller bearing, pulsation anomalies in the case of flow or pressure, and so forth.

Knowing that vibration sensors are common sensory measurements used in rotating machinery applications, we may consider taking a closer look at frequency analysis of vibration signatures recorded by accelerometer or displacement probe sensors. Figure 5, depicts the Fast Fourier Transform (FFT) of a vibration sensor signature from the bearing on the input side of a gearbox.

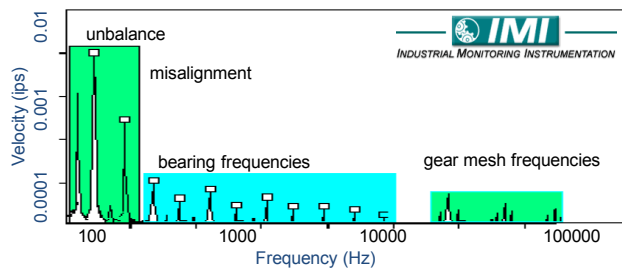


Figure 5. Fault frequencies of machine vibration

In fact, many mechanical faults are detectable using vibration amplitude analysis or extensions to frequency analysis, Table 2 (Jayswal, 2008).

Item	Fault
Gears	Tooth messing faults misalignment cracked and/or worm teeth eccentric gear
Rotors and shaft	Unbalance Bent shaft Misalignment Eccentric journals Loose components Rubs Critical speed Cracked shaft Blade loss Blade resonance
Rolling element bearings	Pitting of race and ball/roller Spalling Other rolling elements defect
Journal/bearing	Oil whirl Oval or barreled journal Journal/bearing rub
Flexible coupling	Misalignment Unbalance
Electrical machines	Unbalanced magnetic pulls Broken/damaged rotor bars Air gap geometry variations
Structural and foundation faults	Structural resonance Piping resonance Vortex shedding

Table 2. Typical faults detectable with vibration analysis

Using the results of the FFT, analytically it is typical to measure the amplitude shown in Figure 5 as Velocity in inches per second (IPS) and compare the amplitude for specific fault modes to historical norms, or to vibration severity standards such as the ISO 10816. When the amplitudes of specific vibration frequencies exceed historical norms, or a standard recommended warning level, the machine component where the vibration sensor is located is considered degraded to the extent its vibration level has exceeded the historical norm or standard warning level.

However, depending on the indicative features of interest, there are a number of signal processing algorithms used to

alternatively or subsequently analyze the time waveform or the results of the FFT. Some of these are listed in Table 3. Advanced analytics build on the FFT creating additional numerical features and augment the degradation status of the machine.

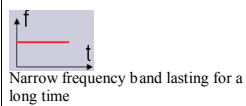
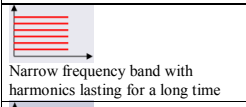
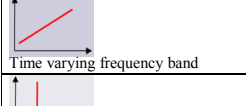
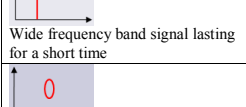
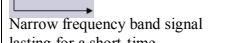
Signal Characteristic	Analysis Methods	Machine Example
 Narrow frequency band lasting for a long time	<ul style="list-style-type: none"> <li>Frequency Analysis</li> <li>Fast Fourier Transform</li> <li>Power Spectrum</li> </ul>	Unbalance in a single speed machine
 Narrow frequency band with harmonics lasting for a long time	<ul style="list-style-type: none"> <li>Quefrency</li> <li>Cepstrum</li> </ul>	Damaged bearing in a machine with roller element bearings
 Time varying frequency band	<ul style="list-style-type: none"> <li>Time-frequency analysis</li> <li>Order analysis</li> </ul>	Unbalance in a variable speed pump
 Wide frequency band signal lasting for a short time	<ul style="list-style-type: none"> <li>Wavelet analysis</li> <li>AR Modeling</li> </ul>	Low speed machine with compressor valve impacts
 Narrow frequency band signal lasting for a short time	<ul style="list-style-type: none"> <li>Wavelet Analysis</li> </ul>	Electrical motor driven machine with rub and knock noise.

Table 3. Signal processing options for dynamic sensors

Specific use cases of advanced signal processing include time synchronous averaging (TSA) to isolate non-synchronous signals from synchronous signals. Additional advanced techniques include Cepstrum which is a frequency type analysis of the FFT (Zhang, 2008). Wavelets and order analysis are additional examples. Each of these advanced signal processing techniques works to clarify specific features found in the original FFT, by removing or isolating those specific dynamic signal amplitudes that best indicate the asset component’s degradation trend or pattern.

#### 4. PROGNOSTIC METHOD SELECTION

There are two general methods of prognostics applications; data driven and model based (Sankavaram, 2009). Data driven methods work best with historical data sets indicating common failure and normal operation of the entire asset as well as its individual components. Model driven methods use mathematical models to describe the relationship between measurements and expected asset behavior.

With data driven methods, historical data is pre-processed to reduce the sensory data to a set of calculated features that describe the normal and various failure conditions. Once this reduction is complete, the data-driven model then relies on one or more health assessment algorithms. These health assessment algorithms work to evaluate the fit of current measurement data to the normal and failure condition feature sets. Example health assessment algorithms include logistic regression, statistical pattern matching, Hidden Markov Models, and Gaussian mixture models.

With model driven methods, a system model is derived from first principle analysis and simulations. Measured data and features, along with system state variables become inputs to the model equation where the outputs map to normal operation or a failure condition. To adapt to specific machinery or operation conditions, often an adaptive learning or model update process is required. The model is derived from system physics and expected behavior. The model then serves as a reference to normal and various failure conditions. Health assessment algorithms then work to fit current measurements to model outputs under similar operating conditions.

As a comparison, data driven methods often require run-to-failure data which may be expensive or impractical to obtain. Model driven methods require accurate modeling and the ability to execute and tune the model in real-time leading. Developing models can be expensive from an engineering perspective. Executing models in real-time as measured data arrives may require extensive computational resources.

Often, a combination of approaches is desirable. Test cell data from design verification testing or factory acceptance testing can provide normal behavior data sets. Many mechanical components have accepted limits on calculated features such as vibration severity levels for which a simplified model can be inferred. By combining data driven, and macro model driven approaches, a basic automated degradation detection and trending system becomes possible.

It is not the intent of the author to imply the prognostics process is easy. Identification of critical assets and selection of sensory information to monitor are well established practices. Yet, development of data driven failure signatures and physics of failure modes are much more difficult. Many Small Business Innovation Research (SBIR) grants are made each year to small prognostic domain expert companies to fit a particular prognostic method to a specific class of industrial machines. Much research at the university level and in industry continues in efforts to formalize algorithms and methodologies for prognostics. One may conclude then, that prognostic is not an absolute science, yet one with much interest and activity in both research and industry. Further experimentation and case studies promise to document successful approaches to make the prognostics system design easier for future implementations.

**5. PILOT IMPLEMENTATION WITHIN A FLEET (CASE STUDY)**

Given a solid understanding of the assets in the fleet, availability of sensors and operational data, as well as historical data sets and any models of asset behavior, it is then possible to design a pilot implementation where

baselines and preliminary prognostic results can be evaluated.

The Electrical Power Research Institute (EPRI) (Hollingshaus, 2011) continues to sponsor a fleet wide asset monitoring project within a special working group, the Fleet-Wide Monitoring Interest Group (FMWIG). This program aims to articulate a condition based maintenance and prognostics solution for its power generation members. The applications framework leverages data available within power generation plants, a fault signature database, and traditional monitoring and analysis techniques for rotating machinery, (Hussey 2006) Figure 6.

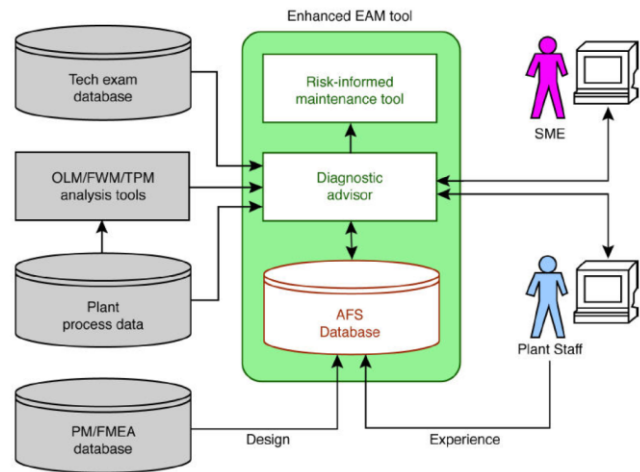


Figure 6. Overview of EPRI FMWIG pilot

Acronyms from Figure 6 are defined here:

- EAM: Enterprise Asset Management System
- AFS: Asset Fault Signature database
- OLM: On-Line Monitoring Systems
- FWM: Fleet Wide Monitoring
- FMEA: Failure Mode and Effects Analysis
- TPM: Thermal Performance Monitoring
- SME: Subject Matter Expert (ex: vibration analyst)
- PM: Preventative Maintenance records

Figure 6, outlines EPRI’s vision for Smart Monitoring and Diagnostics. Currently, existing EPRI pilot projects have included anomaly detection systems such as General Electric’s SmartSignal, and Instep Software’s Prism. These anomaly detection systems operate from plant historian data such as an OSI Soft PI database. These anomaly detection systems are able to develop normal trend patterns and provide notifications when expected operating parameters do not match measured operating parameters. While these trend analyzers provide anomaly detection, it is the technical exam data (vibration, motor current signature, etc) which leads to specific maintenance actions and schedules.

To build on the anomaly detection, EPRI and Progress Energy (now Duke Energy) have embarked on a project to

automate the technical exam, especially vibration. By automating the vibration data collection, the current vibration SMEs will move towards 80% of their time reviewing vibration analysis, as compared to 80% of their time collecting data. (Johnson, 2012), (Cook 2012). To make this shift from data collection to data analysis, the cost of installing permanent vibration monitoring systems had to reduce significantly.

By leveraging high volume commercial off the shelf vibration measurement equipment, and competitively priced vibration sensors, Progress Energy is able to afford installation of over 300 vibration monitoring systems using both wired and wireless Ethernet communications technologies. These systems cover the majority of “balance of plant” equipment including circulating water pumps, pulverizers, fans, transformers, and so forth.

To provide for remote vibration diagnostics, Progress Energy and EPRI are working with vibration analytics software providers to develop an on-line and off-line vibration analytics, which meet the de-facto industry standards for vibration analytics. With hardware and software in place, data storage, aggregation, mining, and fault signature association will become future challenges for the EPRI/Progress team.

Both EPRI and Progress Energy have seen millions of lost dollars in loss of power generation capabilities. The belief is that broader coverage of on-line monitoring along with automated analytics for diagnostics and prognostics will predict and prevent future losses.

As the project moves forward, both data-driven and physics of failure prognostics will be employed as part of the EPRI diagnostic advisor to extend its capabilities to include predictive features. However, data mining, fault signature association, and related prognostics algorithms must be validated to become a universal solution for power generation applications.

The EPRI diagnostic advisor will use the asset fault signature database (AFS) along with on-line monitoring (OLM), trend analysis, any technical exam results, and subject matter experts (SME) to advise plant maintainers and operations of any specific next steps.

As the EPRI project moves forward, with additional pilots, the asset fault database (AFS) will grow and the prognostic methods will improve. The on-line monitoring options including sensors and embedded data acquisition devices will also evolve. There is much to learn from this pilot, yet the opportunity in power generation applications is promising.

In EPRI’s summary, condition based maintenance, diagnostic advisories, and prognostics using asset fault databases will lead to actionable information in time to economically benefit plant operations (Hollingshaus, 2011).

The current pilots are working to validate the prognostics implementation and financial benefits.

## 6. CONCLUSION

The implementation of a fleet wide asset monitoring and advisory system combines several disciplines. These range from traditional condition based maintenance practices, to development of fault models, to implementation of hybrid prognostic systems. There are potentially many benefits derived from a systematically developed prognostic system. These benefits may pay well to electrical power generation and other industries that employ many mechanical assets of similar types and function.

Similar pilots are occurring in the Oil and Gas industry for land based drilling and extraction equipment. Other pilots are just beginning in mining industries, centered on haul trucks, swing shovels, and drag lines. These pilots are similarly challenged by the cost of sensors and data acquisition hardware, cost effective analysis, cost effective data storage, and the development of data driven and physics of failure fault signatures.

## REFERENCES

- Lee, J., Chen, Y., Al-Atat, H., Abuali, M. and Lapira, E. (2009). A systematic approach for predictive maintenance service design: methodology and applications. *International Journal of Internet manufacturing and Services*, Vol. 2, No. 1, pp. 76-94, 2009.
- Reliability Hotwire (2004), Reliability Basics, *Reliability HotWire*, Issue 46, December 2004, <http://www.weibull.com/hotwire/issue46/reliasics46.htm>
- Center for Advanced Life Cycle Engineering (CALCE), University of Maryland (2012). Introduction <http://www.prognostics.umd.edu/tutorials.htm>
- National Aerospace and Space Administration (NASA), (2012). Data-Driven Prognostics <http://ti.arc.nasa.gov/tech/dash/pcoc/data-driven-prognostics>
- Lei, Y., Djurdjanovic, D., Workman, G., Xiao, G. and Lee, J. (2004) Basic prognostics in industrial automation systems., *Proceedings of the 6th International Conference on Frontiers of Design and Manufacturing*. June 21-23, Xi'an, China
- Sankavaram, C., Pattipati, B., Kodali, A., Pattipati, K., Azam, M., Kumar, S., and Pecht, M. (2009), Model-based and Data-driven Prognosis of Automotive and Electronic Systems. *Proceedings of 5th Annual IEEE Conference on Automation Science and Engineering*., August 22-25, Bangalore, India [http://www.prognostics.umd.edu/calcepapers/09\\_C\\_haitanyaSankavaram\\_prgnosis\\_automotiveElecsyst\\_ems.pdf](http://www.prognostics.umd.edu/calcepapers/09_C_haitanyaSankavaram_prgnosis_automotiveElecsyst_ems.pdf)

- Hollingshaus, B. (2011). Program 69: Maintenance Management and Technology. *Electrical Power Research Institute Descriptions of Past Research*. Catalog number 1022681, May. 2011 www.epri.com
- Hussey, A., Hesler, S., and Bickford, R. (2010). Automation of Troubleshooting and Diagnostics of Power Plant Equipment Faults. *Proceedings of 53rd ISA POWID Controls & Instrumentation Conference*. 6–11 June, Summerlin, Nevada
- Bechhoefer, E., and He, D., (2012) A Process for Data Driven Prognostics. *Proceedings of the Prognostics and Health Management Solutions Conference*. 24-26 April, Dayton, OH
- Jayaswal, P., Wadhvani, A. and Mulchandani, A. (2008). Machine Fault Signature Analysis. *International Journal of Rotating Machinery*. Volume 2008, Article ID 583982, 10 pages doi:10.1155/2008/583982
- Zhang, N. (2008) Advanced Signal Processing Algorithms and Architectures for Sound and Vibration. *National Instruments NI-Week Conference*. Presentation TS 1577. August 4-8, Austin, Texas
- Johnson, E. and Johnson, P. (2012) Fleet Wide Asset Monitoring, Status Report from Progress Energy. *EPRI Combined CBM Meeting*, July 16-20, San Diego, California
- Cook, B. (2012) Smart Monitoring and Diagnostics for Power Generation. *National Instruments NIWeek conference*. August 7-9, Austin, Texas

#### BIOGRAPHY

Preston Johnson is the Global Program Manager for Asset Monitoring Systems at National Instruments (NI) in Austin, Texas. NI creates innovative computer-based products that aid engineers in the design, prototyping, and deployment of instrumentation systems for test, control, and embedded applications. He has worked for National Instruments for 25 years in roles of Field Sales, Sales Management, Automation Business Development, Sound and Vibration Segment Manager, and Platform Manager for Condition Monitoring Systems. In his current role as Asset Monitoring Systems Program Manager, his interest lies in embedded signal processing and data acquisition systems and architectures. Preston works with NI OEM and End User customers to deploy fleet-wide asset monitoring systems that lower operation costs, improve machinery reliability, and ultimately increase revenue. He earned his BSEE in Electrical Engineering and Computer Science from Vanderbilt University in 1985 and his MBA in Information Technologies from the University of Texas in 1987.



# Prognostics Assessment Using Fleet-wide Ontology

Gabriela Medina-Oliva<sup>1</sup>, Alexandre Voisin<sup>1</sup>,  
Maxime Monnin<sup>2</sup>, Flavien Peysson<sup>2</sup>, Jean-Baptiste Leger<sup>2</sup>

<sup>1</sup> *Centre de Recherche en Automatique de Nancy (CRAN), Université de Lorraine, UMR 7039 CNRS-UHP-INPL, Faculté des Sciences-1er Cycle - BP239, 54506 Vandoeuvre-Les-Nancy Cedex – France*

*gabriela.medina-oliva@univ-lorraine.fr  
alexandre.voisin@univ-lorraine.fr*

<sup>2</sup> *PREDICT 19, Avenue de la Forêt de Haye, CS 10508, 54519 Vandoeuvre-Lès-Nancy, FRANCE*

*maxime.monnin@predict.fr  
flavien.peysson@predict.fr  
jean-baptiste.leger@predict.fr*

## ABSTRACT

Large complex systems, such as power plants, ships and aircraft, are composed of multiple systems, subsystems and components. When they are considered as embedded in operating systems such as a fleet, mission readiness and maintenance management issues are raised. PHM (Prognostics and Health Management) plays a key role in controlling the performance level of such systems, at least on the basis of adapted PHM strategies and system developments. Moreover considering a fleet implies to provide managers and engineers a relevant synthesis of information and to keep this information updated in terms of the global health of the fleet as well as the current status of their maintenance efforts. In order to achieve PHM at a fleet level, it is thus necessary to manage relevant knowledge arising from both modeling and monitoring of the fleet. In that way, this paper presents a knowledge structuring scheme based on ontologies for fleet PHM management applied to marine domain, with emphasis on prognostics modeling.

## 1. INTRODUCTION

Nowadays, due to the high competitiveness, industrial enterprises need to aim at higher performances, i.e. higher quality of products/services, lower costs, sustainability, etc. (Kleindorfer et al., 2005). In that way, the importance of maintenance has increased due to its key role on improving system availability, performance efficiency, products quality, etc. (Alsyouf, 2007). These requirements promote the evolution of maintenance strategies from a “fail and fix”

to “predict and prevent” approach. This new vision is supported by condition-based/Prognostics and Health Management (PHM) maintenance. Despite this proactive approach, failures still occur. This could be explained since prognostics involved the prediction of the future which is uncertain (Provan, 2003). Furthermore the whole acquisition and treatment algorithms could fail leading sometimes to some errors such as false alarms or non-detections (Barros et al., 2006).

Implementing a proactive approach at a system level requires the consideration of failure rates of different equipment built on different technologies (mechanical, electrical, electronic or software natures) (Verma et al., 2010) whose behaviour can vary all along the different phases of their lifecycle (Bonissone and Varma, 2005). However to improve PHM processes for large and complex systems such as power plants, ships and aircrafts, one possible approach is to take advantage of the fleet dimension. This dimension can provide knowledge and data to improve diagnostic and prognostics models.

A fleet shall be viewed as a set of systems, sub-systems and equipment. In this paper, the naval domain is addressed. Hence, in the following an unit of a fleet will be considered as a system (e.g. ship), a sub-system (e.g. propulsion or electric power generation) or equipment (e.g. diesel engine, shaft...) depending on the nature of the study. To be in accordance with the need of improving PHM at the fleet level, an original methodology is proposed in this paper wherein individual knowledge (of each unit) is capitalized for reuse purpose in order to improve PHM activities such as prognostics. To take advantage of the individual knowledge at the fleet level, a semantic model is proposed for the PHM activities in the naval domain. Such a semantic model enables to reuse particular data, such as maintenance history, reliability analysis, failure analysis, data analysis at

Gabriela Medina-Oliva et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

a fleet level in order to provide knowledge. As data become available, prognostics models could benefit from more contextual information.

## 2. PHM AT EQUIPMENT AND SYSTEM LEVEL

PHM activities can give warning about failure events before they occur, reduce the life cycle cost of a product by decreasing inspection costs, downtime and inventory (Pecht, 2008), (Vichare and Pecht, 2006).

The prognostics process consists on treating, via algorithms, a set of input information to produce a future estimation. Mathematical models are used for the extrapolation of value of the degradation indicator. These mathematical models need as an input (Voisin et al., 2010):

- Past data: feedback about past failures on the system, as well as historic data about the evolution of degradation indicators under different circumstances (mission, environment, etc.). This data allows to identify the characteristics and parameters of the prognostics model.
- Current data: on-line data in order to provide the values in real-time of the monitored indicators (variables). This data is captured and must be treated and analyzed. This data warn maintenance engineers about the current state of the unit and it should be used to feed/adjust the current prognostics model.
- Future data: information and/or hypothesis about the future usage of the unit should be provided such as the missions, the operational context, future maintenance interventions, etc. As mentioned in (Peysson et al., 2009) the prognostics of a complex systems (S) is described by three levels (1):

$$S = \langle M, E, P \rangle \quad (1)$$

M is the mission that defines the use of the system during a time period; E is the environment that represents the area where the mission is accomplished and where the process evolves and P is the process that gives the necessary means to accomplish the mission. The process is decomposed according to different resources that are monitored.

This set of information/data allows to refine the evolution of degradation indicators in different situations and simulations.

While analyzing these inputs, different sources of uncertainty will appear such as measurement and sensor errors, future load and usage uncertainty, prediction under conditions that are different from training data and so on. However uncertainty could be reduced when more data becomes available (Pecht, 2008). In these cases the notion of fleet becomes very interesting. It can provide more capitalized data and information coming from other

members of the fleet. for the improvement/development of the prognostics models

The following section presents a review about the use of the fleet notion in the PHM domain.

## 3. PHM VS. FLEET-WIDE APPROACH

### 3.1. Fleet integrated PHM review

A fleet generally refers to a gathering of group of ships and the term is extended also to any kind of vehicle (e.g. trains, aircrafts, or cars). For industrial systems, the term fleet designs a set of assets or production lines. In general, a fleet refers to the whole assets of an owner's systems. Hence, the fleet here is only an abstract point of view to consider a set of objects for a specific purpose (e.g. a unit maintenance planning), for a given time (e.g. before the end of the current mission). Indeed, the fleet can be viewed as a population consisting of a finite set of objects (individuals) on which a study is ongoing. In this context, a fleet is generally a subset of the real fleet under consideration, i.e. a sub fleet related to the aim of the study. Individuals making up the fleet/sub fleet may be, as needed, systems themselves (Bonissone and Varma, 2005), (Patrick et al., 2010), subsystems or equipment (Umiliacchi et al., 2011). In the following, systems, sub-systems or equipment constituting the fleet, according to the study purpose, will be referred to as units.

In fact, fleet's units must share some characteristics that enable to group them together according to a specific purpose. These common characteristics may be of technical, operational or contextual nature (Monnin et al. 2011a). They allow to put data or information related to all the fleet units on the same benchmark in order to bring out pertinent results for monitoring, diagnostics, prognostics or maintenance decision making. Common characteristics among units allow to define three types of fleet composition: identical, similar or heterogeneous fleets.

Based on these three types of fleet, some relevant works are reviewed below:

- Fleet composed of identical units: When considering maintenance operator's point of view, fleet management aims at making decisions that affect asset life extension and performance, operational costs and future planning (Wheeler et al., 2009), (Bonissone and Varma, 2005),(Williams et al., 2008). In (Patrick et al., 2010), the authors notice that thresholds indicative of condition indicators limits could be derived from statistical studies of fleet wide behaviors and known cases of faults. (Reymonet et al., 2009) propose to apply to the failed system the technical solution corresponding to a similar incident already solved with a comparable asset. (Wang et al., 2008) present a similarity-based approach for estimating the Remaining

Useful Life (RUL) in prognostics using data from a fleet composed by the same type of units. Nevertheless, knowledge derived from the fleet arises from the same kind of units. In a domain where customized units are common, these approaches may give poor results.

- Fleet composed of similar units: the fact of comparing similar units has rarely been addressed as a whole in the literature. In that sense, (Umiliacchi et al., 2011) show the importance of having a standard format for the diagnostic data in order to facilitate their understanding across several subsystems and trains within a railway fleet.
- Fleet composed of heterogeneous units: to fully exploit the knowledge issue of the fleet dimension, we propose in this paper to consider the heterogeneous units that compose the fleet level for decision making (e.g. prognostics, maintenance purposes). Situations (i.e. prognostics results, signal evolution) issued from an historical data of a fleet of heterogeneous units are searched based on some similar characteristics to the units in study (e.g. current situation under investigation) (Monnin et al., 2011a), (Monnin et al., 2011b). The originality of the proposed method is to enlarge the search to heterogeneous units (and not identical ones) where the similarity will be defined on-line by the user, according to the results of the search, in order to find relevant information to be reused.

### 3.2. But how could be used the fleet dimension to improve PHM processes?

One of the industrial realities is the lack of capitalization of knowledge and model reuse which represents high costs and efforts for the enterprises (Weber et al., 2011) (Medina-Oliva et al., 2012). In some fields such as the naval one, units are very customized leading to heterogeneous units. These facts limits mainly:

- Historical data exploitation for identical units. Due to the exposition of industrials systems to different and uncertain missions and environmental conditions.
- Knowledge capitalization about the evolution of the degradation trajectory of an unit under an identical context (identical mission and environment).

To tackle this issue the fleet dimension could provide enough information and data about diagnostic and prognostic models. In that sense, when searching non-identical units but similar ones a higher volume of data becomes available to reduce diagnostic/prognostics models uncertainty. This data could be obtained through the identification of “similar contexts” or “similar individuals”. For example, in the naval field, a technical similarity for diesel engines, which are critical equipment for propulsion and power generation, could be the membership to “4-strokes engines” or “high speed engines” features.

The objective of our proposition is to create an iterative investigation process that will allow to define a sub-fleet. The sub-fleet is defined by grouping a set of units (i.e. systems, sub-systems or equipment) based on “similar characteristics”. Figure 1 presents the main steps of this process. When prognostics models are implemented on a new unit or when the unit is merged in new operational conditions, the prognostics must be first characterized. Characterization consists on the description of the unit to be pronosticated (e.g. type, age, usage, operating environment) as well as data on which an analysis will be carried out (Figure 1-A). To guide this process, the fleet-wide application proposes different criteria based on the technical features of units as well as on the mission and on the environment description. According to this, a targeted population is defined within the whole fleet (Figure 1-B). Within the targeted population, the potential similar prognostics models and data concerning the evolution of the degradation indicators are investigated in order to complete current knowledge for prognostics (Figure 1-C). In an iterative process of steps B and C, the targeted population can be refined if results (i.e. potential similar degradation indicator data) are too far from the current situation or conversely if targeted fleet points out too many similar situations. Then, the prognosticated unit benefits from the past analysis results (Figure 1-D) for building the corresponding prognostics model. Using this original approach could bring several benefits as presented in (Monnin et al., 2011a), (Monnin et al., 2011b), (Peysson et al., 2012).

Toward this goal, the knowledge corresponding to the fleet domain must be well formalized and structured in order to facilitate the manipulation of the multidimensional aspects of the fleet and the heterogeneous data among all fleet units’ databases

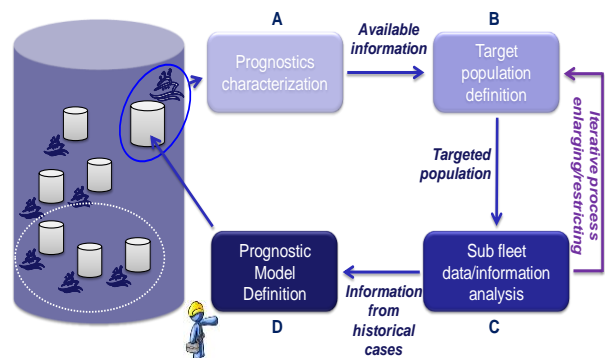


Figure 1. Main steps for fleet case re-use

### 3.3. Sub-fleet characterization

Nowadays, fleet managers/engineers query separately ship databases for identical units to obtain data/information. However, units (e.g. diesel engine) in the naval domain are

very specific and customized. This fact leads to dispose of few identical units. For this reason the proposed methodology leads to search non-identical units but similar ones. Furthermore, another issue emerges. At a fleet level, engineers must treat different databases of different units. These databases might be heterogeneous, in the sense that the database structure might be different, they might have different names for the primary keys and foreign keys, etc.

To tackle these issues, a common semantic becomes necessary (Figure 2). A semantic model provides a high level definition of terms that is common to all databases allowing to query them. It allows to define characteristics of similarities among units and contexts. For instance, to define common characteristics in the technical, operational and contextual domains.

In the following sections a semantic model specifying the similar characteristics necessary to obtain data/information to perform prognostics is presented. The semantic model is specified by means of ontologies.

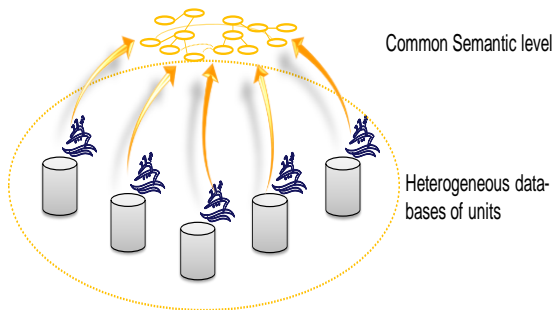


Figure 2. Semantic level to query heterogeneous databases

#### 4. PROPOSITION OF ONTOLOGY FOR FLEET-WIDE SEMANTIC

##### 4.1. Ontology for providing semantic

An ontology explicits formal specifications of knowledge in a domain by defining the terms (vocabulary) and relations among them (Gruber, 2009). To represent knowledge and to explicit semantic (vocabulary and relations), the ontology is coded in Web Ontology Language (OWL) supported by Protégé\* ontology editor. OWL is composed of classes, properties of the classes and instances. These elements are explained as follow:

- Classes describe concepts in the domain. In PHM domain, an example of classes could be “equipment” or “degradation indicator”.
- Properties of the classes describe the attributes of the concepts. For example, the class equipment has a property “is monitored by” the class degradation

indicator. The property “is monitored by” link the class “equipment” with the class “degradation indicator”.

- Subclasses represent concepts that are more specific than the superclass (mother class). When a superclass has a subclass, it means that they are linked by a subsumption relation, i.e. “is a” relation, allowing a taxonomy to be defined. Hence, a hierarchy of classes is established, from general classes to specific ones.
- Instances are the set of specific individuals of classes. For example, the engine Baudouin 12M26.2P2-002 is a specific individual that is part of the class “equipment”.

OWL allows to establish taxonomies. This capability is a useful for example to represent systems, subsystems and equipment. Moreover, OWL provides inference capabilities with plugged reasoners. Inference is based on open-world reasoning. Explicit and manually constructed classes that belong to taxonomy constitute an asserted hierarchy. But thanks to OWL reasoners, an inferred hierarchy is automatically computed allowing to emerge new knowledge. For example if an engine has an internal electrical degradation, the ontology could induce that it is an electrical engine. Moreover, OWL reasoners shall performed consistency checking. Hence, one shall guarantee that the ontology has been built correctly in the sense that no syntactic and inconsistency remain in the ontology. For example, if a fuel-engine is associated to an electrical degradation, an inconsistency will be point out by the reasoner.

Ontologies seem to be a suitable modeling method to provide common semantic and to query heterogeneous databases. Some of the capabilities provided consist on: sharing common understanding of the structure of information among people or software agents, making domain assumptions explicit, defining concepts and knowledge (i.e. a high speed engine 1000rpm, subjectivity is limited) and making domain inferences to obtain non-explicit knowledge (Noy and McGuinness, 2001).

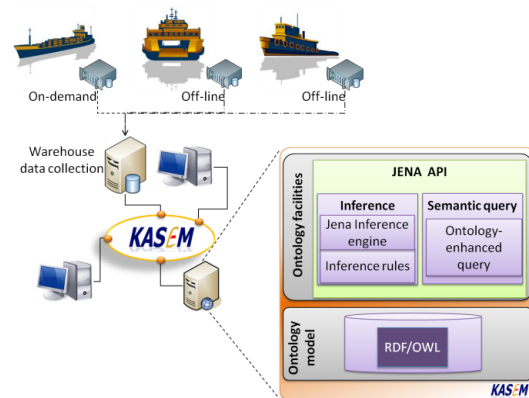


Figure 3: Typical architecture of Fleet-wide PHM system, (Monnin, 2011a)

\* <http://protege.stanford.edu/>

The proposed approach based on PHM ontology is not a goal itself. This ontology is a support for PHM software applications through the KASEM software platform (Leger, 2004), (Monnin, 2011c) Figure 3.

In the next section, the formalization of an ontology to support prognostics activities in the naval domain is proposed.

#### 4.2. Ontology-based PHM assessment

##### Ontology-based fleet-wide PHM

In the section 2.1. key elements to perform prognostics were identified. These elements are technical characteristics of the system/sub-system/equipment, degradation modes, degradation indicators, the mission and the environment.

A semantic model to these elements will be provided in order to obtain formalized knowledge that allows the definition of “similarities” among these units. Based on (Monnin, 2011a) different contexts are defined: a technical context (e.g. characteristics of the system/sub-system/equipment), a dysfunctional context (e.g. degradation modes), an operational context (e.g. mission and environment), a service context (e.g. usage of units) and an application context (e.g. degradation indicators). For a graphical representation of the ontology for these contexts, classes are represented as ovals and relations are represented as links between the classes. Once the ontology model is formalized comparisons of heterogeneous units shall be performed on the basis of context similarity.

##### Technical context

One might think that the definition of every model of equipment could be enough to take advantage of the fleet dimension. But this modeling choice narrows the manageability to strictly identical units. For this reason, a technical context is proposed. The technical context integrates the technical features and characteristics of the system/sub-system/equipment. This model allows the comparison of heterogeneous units for instance when seeking relevant characteristics such as “4-strokes engines” or “direct injection engines”.

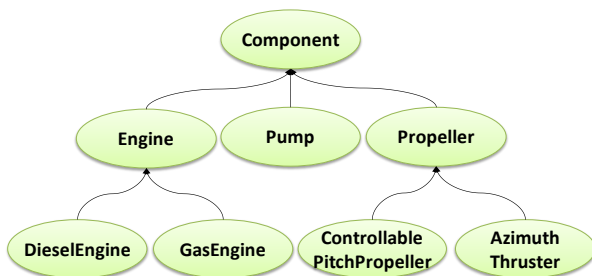


Figure 4: Part of the equipment taxonomy

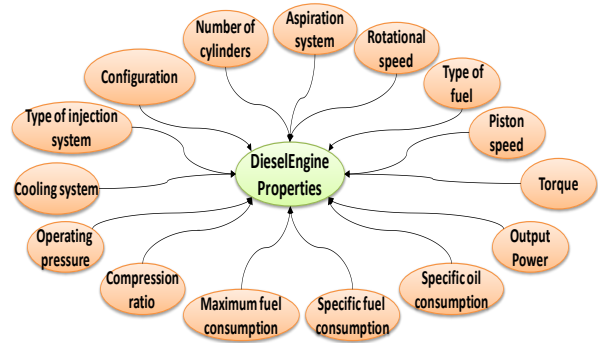


Figure 5: Part of the equipment properties

To model the technical context, “Equipment” (Figure 4) classes are specified as well as “Properties” classes (Figure 5) which define all their features. Hence, units with similar technical properties could be clustered according to their technical properties such as the power output, the rotation speed, the number of cylinders, etc. (Figure 5) in order to retrieve data/information.

##### Dysfunctional context

The dysfunctional context takes into account the information about the degradation modes on the units. It considers the generic degradation modes. Generic degradation modes are taken from the standard (IEC 60812, 2006). Classes include electrical degradation modes, mechanical degradation modes, hydraulic degradation modes, etc. Degradation modes are linked to units (Figure 6).

Furthermore, in this context, it is considered that one degradation mode could be caused by another degradation mode. In that sense, this context allows to describe information about the causality chain of degradation modes that produced an undesirable event. This knowledge is very valuable to retrieve information/data for troubleshooting and corrective maintenance issues. This modeling choice allows to explore the main causes of similar degradation modes. For example to explore common causes of pumps failures regardless of the use of the pumps.

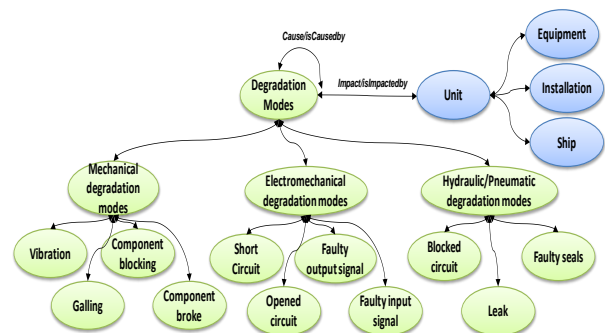


Figure 6: Part of the dysfunctional analysis on units

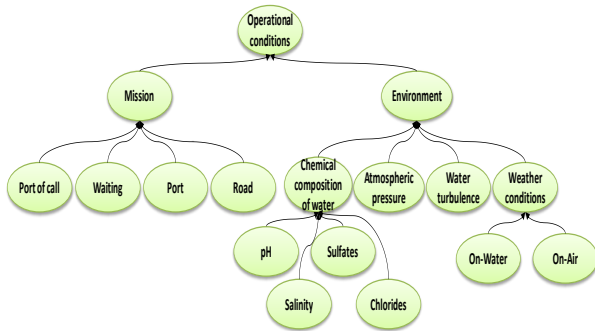


Figure 7: Part of the operational context

### Operational context

Even if units are identical, the operational conditions lead to different units' behaviors. For this reason, one could need to cluster information/data according the operational and service contexts.

The operational context integrates the operational conditions to which the units are exposed to. As explained in section 2, for prognostics the mission (*M*) and the environment (*E*) are considered. The operational conditions are given by the mission to be performed for units as well as the environment that surround them (Figure 7). In the naval domain, the mission is a sequence of dated tasks in a geographical area (e.g. Port of call mission) (Peysson et al., 2009). Hence, similar missions on similar units could be compared (Figure 7). On the other side, the environment takes into account the weather conditions, the chemical composition of water (pH, salinity...), the pressure, water turbulence, etc. (Figure 7) which might impact degradation mechanisms and units' functioning behavior.

The mission and the environment could affect equipment, sub-system and system performances. For this reason the mission was formalized at different abstraction levels (system, sub-system, equipment).

### Service context

The service context deals with the usage of units. Even when units are similar they are exposed to different usages according to the corresponding mission tasks. Hence a service context is formalized in order to differentiate behaviors of the evolution of degradation indicators. In that sense, usage could be divided according to the operating steps, operating phases and the configuration of units (Figure 8).

In order words, degradation behavior can be analyzed according to different abstraction levels.

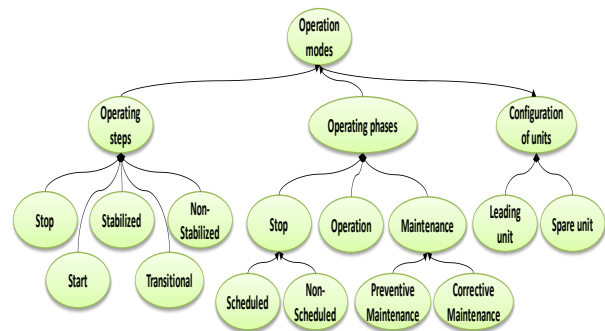


Figure 8: Part of the service context

### Application context

The application context is related to the needs of PHM optimization. Within the application context, the optimization aims the capitalization of knowledge to perform health assessment. Health assessment deals with the definition of indicators such as functional, dysfunctional and environmental indicators at different levels as well as the treatment (processing) of these indicators, etc (Figure 9). This context enables data/model retrieval of the monitored unit with its corresponding context defined in the ontology.

The ontology-based knowledge formalization provides the basis to capitalize knowledge with contextual information. In that sense, one might define similar units, i.e. high speed engines with similar missions and under similar context in order to compare signal evolution of indicators. The next section illustrates the benefits of the proposed methodology within the PHM context.

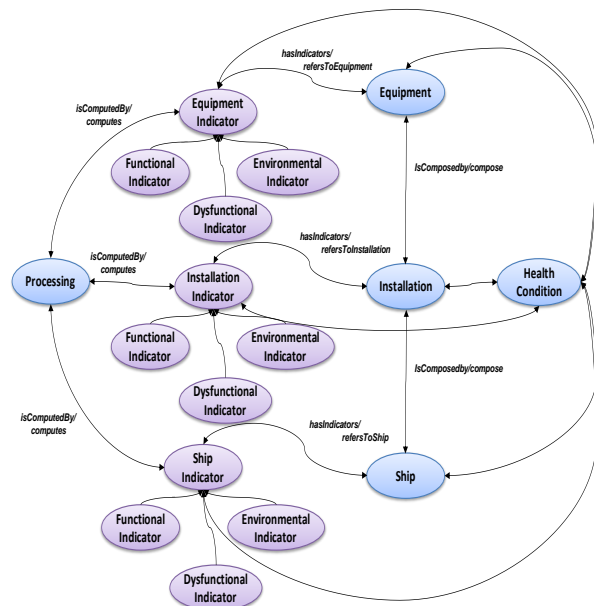


Figure 9: Part of the application context

ID	Engine Ref	Output power (kW)	Nb. of Cylinder	Configuration	Engine Speed (rpm)	Type of cooling system	Engine cycle	Tag related to the ontology
Eqpt1	Wärtsilä 12V38	8 700	12	V	600	Dry water cooling system	Four-stroke engine	Fuel engine
Eqpt2	Wärtsilä 12V38	8 700	12	V	600	Dry water cooling system	Four-stroke engine	Fuel engine
Eqpt3	Baudouin 6M26SRP1	331	6	L	1800	Sea water cooling	Four-stroke engine	Fuel engine
Eqpt4	Man V8-1200	883	8	V	2300	Desalination systems	Four-stroke engine	Fuel engine
Eqpt5	Man V8-1200	883	8	V	2300	Desalination systems	Four-stroke engine	Fuel engine
Eqpt6	Wärtsilä RT-flex50	13 960	8	L	124	Sea water cooling system	Two-stroke engine	Fuel engine
Eqpt7	Wärtsilä RT-FLEX82T	40 680	9	L	80	Sea water cooling system	Two-stroke engine	Fuel engine
Eqpt8	Wärtsilä 12V64	23280	12	V	400	High and low temperature separated circuits	Four-stroke engine	Fuel engine
Eqpt9	ISOTTA v11716hpcr	2750	16	V	2100	High and low temperature separated circuits	Four-stroke engine	Fuel engine
Eqpt10	Baudouin 12M26P1FR	357.94	12	V	1800	High and low temperature separated circuits	Four-stroke engine	Fuel engine

Table 1: Extract of engine fleet technical features stored in the data bases

### Prognostic-model retrieval from fleets composed of heterogeneous units

The proposed approach could be very useful for PHM processes for new units. In the case of new equipment there's neither degradation indicators defined nor historical data about their evolution. But when disposing of an ontology-based approach, knowledge and data could be gathered from heterogeneous units and contexts. To illustrate the proposed methodology, To tackle the fleet dimension three possibilities are shown:

1. *Consideration of identical fuel engines to the Baudouin 8M26SRP engine*
  2. *Consideration of all fuel engines composing the fleet*
  3. *Consideration of the heterogeneous units composing the fleet using the ontology-based approach*
1. *Consideration of identical fuel engines to Baudouin 8M26SRP*

The first step to capitalize data from the fleet dimension would be to consider identical units to Baudouin 8M26SRP. Nevertheless, if the same kind of units is considered, then any unit would match to the results as shown in Table 1. For these reason, other approaches should be investigated.

2. *Consideration of all fuel engines composing the fleet*

Another approach to take advantage of the fleet dimension would be to consider all the fuel engines composing the fleet. This approach allows the capitalization of data. However, the fleet is composed of a wide variety of different engines. If one take a look to Table 1, it is possible to notice the fleet is composed

let's consider finding a prognostics model for the degradation trajectory of a diesel engine within a costal surveillance mission in the south Atlantic for a new ship. In this example, we suppose the new ship is propelled by a diesel engine Baudouin 8M26SRP.

For the purpose of this example, the fleet is limited to diesel engines. Ten engines are considered and briefly presented in Table 1. The table shows an extract of the technical features of the engines.

of big size, high power, two-stroke engines such as the Wärtsilä RT-FLEX82T engine and of other engines with lower output power and four-stroke cycles such as the Wärtsilä 12V38 engine. These units are very different and thus their degradation behavior is very different as well. For this reason considering all the fuel engines might not be appropriated to study the degradation behavior of the Baudouin 8M26SRP engine.

3. *Consideration of the heterogeneous units composing the fleet using the ontology-based approach*

As mentioned previously, the developed application based on ontologies guides the definition of "similar characteristics" in order to define the sub-fleet of units to be used for prognostics purposes (Figure 1-A). To guide the sub-fleet definition some questions will be asked, automatically by the application, based on the relations between the different contexts defined in the ontology. One question concerns the *application context*, if one seeks a degradation indicator, a treatment or mainly a prognostics model. For this example, a prognostics model is searched. Then another question deals with the *technical context*. The application poses questions about the type of unit (i.e. engine, thruster, pump...), the application domain (i.e. marine, land, aeronautics, etc.) and the subsystem (i.e.

propulsion and power generation). For this example, an engine located on a ship (marine) for propulsion purposes is sought.

In the ontology the definition and relations of prognostics have been established. For this reason, another question is asked about the mission and the environment of the engine (*operational context*). The mission in this example is costal surveillance mission (Figure 1-B). One can start searching data found in the sub-fleet definition (Figure 1-C).

Afterward, the application proposes to keep finding other similarities criteria for example those related to the geographical area (Figure 1-A). It proposes several choices such as the south Atlantic, north Atlantic, Indian, south Pacific oceans, etc. For this example the environment is located in the south Atlantic. The sub-fleet evolves automatically when other criteria are chosen.

The ontology embeds that hot oceans impact the performances of the cooling systems. For this reason, another question is asked about the type of cooling system such as high and low temperature separated

circuits, sea water cooling, dry water cooling system or desalination systems (*technical context*). The answer, in this case, for the Baudouin 8M26SRP engine is a high and low temperature separated circuits. The sub-fleet is shown automatically. Three similar equipment with similar contexts are found: Baudouin 12M26P1FR, Wärtsilä 12V64 engine and ISOTTA v1716hpcr (Figure 1-D).

Figure 10 shows the updated degradation prognostics models of these three engines. The time horizon represents the mission that the engines have already experienced. Capitalizing this information allows to build a prognostics model for the Baudouin 8M26SRP engine (Figure 13-orange line). To build the estimated degradation trajectory, histograms for each monitored time issues from retrieved trajectories are built. Then the estimated trajectory is computed as the mean on each of the histograms. The orange line shows the estimate of the degradation trajectory of the Baudouin 8M26SRP engine for the given mission. However, more sophisticated methods could be used to estimate the degradation trajectory such as proposed by (Liu et al., 2007) and (Wang et al., 2012).

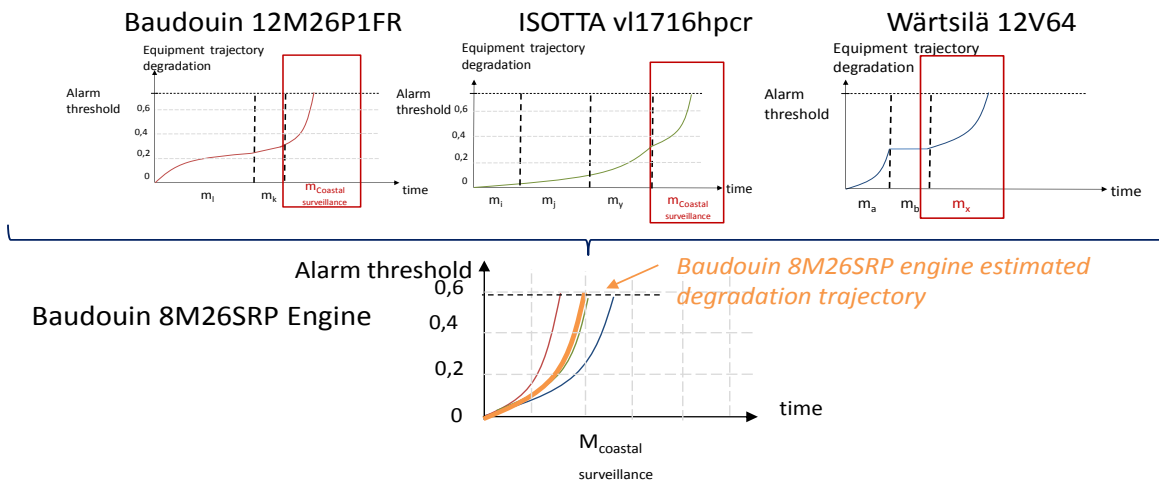


Figure 10. Degradation trajectory estimation for the Baudouin 8M26SRP engine based on similar units behaviors for a costal surveillance mission

## 5. CONCLUSION

Fleet-wide PHM requires knowledge-based system that is able to handle contextual information. Prognosis and maintenance decision making processes are improved by means of semantic modeling that deals with concepts definition and description. In this paper, a knowledge model based on ontologies is proposed. Contextual information is structured by means of specific contexts. These contexts allow to consider fleet unit similarities and heterogeneities. Data of the monitored unit are considered within their

context and enhance the identification of the corresponding health condition.

From a prognosis point of view, the analysis of a degradation variable evolution could improve the prognostics model and precision can be improved through the capitalized data.

The fleet knowledge model has been structured for a marine application. The resulting ontology has been integrated in the KASEM industrial PHM platform. Some experimentation has already been done however; other



experimentations should be tested to show the feasibility and the added value of this methodology. This paper arises some perspectives related to the manipulation of the uncertainty of prognosis (Petch, 2008), as well as the definition of the trajectory model based on retrieved models (Liu et al., 2007) and (Wang et al., 2012).

## REFERENCES

- Alsyouf, I. (2007). The role of maintenance in improving companies' productivity and profitability. *International Journal of Production Economics*, 105, 70–78.
- Azam M., Tu F. and Pattipati, K. R. (2002) "Condition Based Predictive Maintenance of Industrial Power Systems", SPIE conference on Fault Diagnosis, Prognosis and System Health Management, Orlando, April 2002.
- Barros A., Bérenguer C., Grall A. (2009). A maintenance policy for two-unit parallel systems based on imperfect monitoring information. *Reliability Engineering and System Safety* 91 (2006) 131–136.
- Bonissone, P.P., Varma, A. (2005). Predicting the Best Unit within a Fleet: Prognostics Capabilities Enabled by Peer Learning, Fuzzy Similarity, and Evolutionary Design Process. In *Proceedings of the 14th IEEE International Conference on Fuzzy Systems*, IEEE, pp. 312-318.
- Gruber, T. (2009), *Ontology*. In: the *Encyclopedia of Database Systems*, Ling Liu and M. Tamer Özsu (Eds.), Springer-Verlag.
- IEC 60812. Analysis techniques for system reliability – Procedure for failure mode and effects analysis (FMEA). 2006
- ISO 13381-1:2004. Condition monitoring and diagnostics of machines- Prognostics- Part 1: General guidelines
- Kleindorfer PR, Singhal K, Van Wassenhove LN. (2005). Sustainable operations management. *Production and Operations Management*; winter, 14(4):482–492.
- Léger J-B. (2004). A case study of remote diagnosis and e-maintenance information system, Keynote speech of IMS'2004, International Conference on Intelligent Maintenance Systems, Arles, France.
- Liu J., Djurdjanovic D., Ni J., Casoetto N., Lee J. (2007). Similarity based method for manufacturing process performance prediction and diagnosis. *Computer in Industry* 58, Pages 558-566.
- Medina-Oliva G., Weber P., Levrat E., Iung B. (2012) Using probabilistic relational models for knowledge representation of production systems: A new approach to assessing maintenance strategies. *CIRP Annals - Manufacturing Technology*. in press. DOI: 10.1016/j.cirp.2012.03.059
- Monnin M., Abichou B., Voisin A., Mozzati C. (2011b). Fleet historical cases for predictive maintenance. *The International Conference Surveillance 6*. October 25-26. Compiègne, France.
- Monnin M., Voisin A., Leger JB., Iung B. (2011a). Fleet-wide health management architecture. *Annual Conference of the Prognostics and Health Management Society*. Montreal, Quebec, Canada.
- Monnin, M, Leger, J-B., Morel, D. (2011c). KASEM@: e-Maintenance SOA Platform, in *Proceedings of 24th International Congress on Condition Monitoring and Diagnostics Engineering Management*, 29th May – 1st June, Stavanger, Norway.
- Noy N. F. and McGuinness D. L. (2001). *Ontology development 101: A guide to creating your first ontology*. Technical Report SMI-2001-0880, Stanford Medical Informatics.
- Patrick, R., Smith, M J., Byington, C S., Vachtsevanos, G J., Tom, K., Ly, C. (2010). Integrated Software Platform for Fleet Data Analysis, Enhanced Diagnostics, and Safe Transition to Prognostics for Helicopter Component CBM, in *Proceedings of Annual Conference of the Prognostics and Health Management Society*, October 10-16, Portland, Oregon.
- Pecht, M. (2010). A prognostics and health management roadmap for information and electronics-rich systems, *IEICE Fundamentals Review*, vol. 3, no. 4, pp. 25 – 32.
- Peysson F, Ouladsine M, Outbib R, Leger JB, Myx O, Allemand C (2009). "A generic prognostics methodology using damage trajectory models." *IEEE Trans. Reliab.*, 58(2) (277 - 285), June.
- Peysson F., Léger J-B., Allemand C., Ouladsine M., Iung B. (2012) New approaches for ships fleet-wide management and naval mission prognostics. *MFTP 2012: The Prognostics and Health Management Solutions Conference*. April 24-26. Dayton, Ohio, USA.
- Provan, G., (2003). Prognosis and Condition-Based Monitoring: an open systems architecture. *5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, 57--62, Washington, USA.
- Rajkumar and Bardina. Training data requirement for a neural network to predict aerodynamic coefficients. *Proceedings of SPIE*, 2003.
- Reymonet, A., Thomas, J., Aussenac-Gilles, N. (2009). *Ontology Based Information Retrieval: an application to automotive diagnosis*, in *Proceedings of International Workshop on Principles of Diagnosis*, June 14-17, Stockholm, Sweden, pp. 9-14.
- Umiliacchi, P., Lane, D., Romano, F. (2011). Predictive Maintenance of railway subsystems using an Ontology based modelling approach, in *Proceedings of 9th world Conference on Railway Research*, May 22-26, Lille, France.
- Verma, A. K. and Srividya, A. and Ramesh, P. (2010). A systemic approach to integrated E-maintenance of large engineering plants, *International Journal of Automation and Computing*, vol. 7, pp. 173-179.
- Voisin, A., Levrat, E., Cochetoux, P., & Iung, B. (2010). Generic prognosis model for proactive maintenance

decision support: Application to pre-industrial e-Maintenance test bed. *Journal of Intelligent Manufacturing*. 21 (2): 177-193.

Wang P., Youn B., Byeng D. A generic probabilistic framework for structural health prognostics and uncertainty management. *Mechanical Systems and Signal Processing*. 28, Pages 622–637

Wang T., Yu J., Siegel D., Lee J. (2008). A similarity-based prognostics approach for Remaining Useful Life estimation of engineered systems. *International Conference on Prognostics and Health Management*. Denver, USA.

Weber P., Medina-Oliva G., Simon C., Iung B. (2012). Overview on Bayesian networks Applications for Dependability, Risk Analysis and Maintenance areas. *Engineering Applications of Artificial Intelligence*, vol. 25 (4), (671-682).

Wheeler, K., Kurtoglu, T., Poll, S.D. (2009). A survey of health management user objectives related to diagnostic and prognostics metrics, in *Proceedings of International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, August 30–September 2, San Diego, California, USA.

Williams, Z., Gilbertson, D. & Sheffield, G., (2008). Fleet analysis and planning using CBM+ open architecture, in *Proceedings of IEEE International Conference on Prognostics and Health Management*, Denver, CO.

## BIOGRAPHIES

**Dr Gabriela MEDINA-OLIVA** is a Research Engineer at CRAN. She received his PhD from the University of Nancy and her M.S. in Reliability and Risk Analysis from the University of Las Palmas Gran Canarias in Spain. She has experience in maintenance within the oil industry. Moreover, she has worked in the formalization of knowledge with probabilistic tools for maintenance strategies assessment. Her current work focuses on fleet-wide health management. He is member of the French DIAG 21 Association ([www.diag21.com](http://www.diag21.com)) and of the PHM Society.

**Dr Alexandre VOISIN** was born in Metz, France, in 1969. He obtained an engineering degree in Electrical Engineering from the EFREI in 1992 and he received his Ph. D degree in Electrical Engineering from the Université de Lorraine. He is currently assistant professor at the Université de Lorraine. His primary research were in the field of fuzzy logic and information processing where he applied these techniques to subjective evaluation in the area of car seat comfort. Since 2003 he is involved in e-maintenance research project. His research deals with dependability, maintenance decision in a proactive maintenance strategy, prognostics and monitoring. He is member of French and International projects/groups on e-maintenance such as the CNRS MACOD working group (Modeling and Optimization of Distributed vs. Collaborative Maintenance), the French scientific interest

group 3SGS on “Dependability of Complex Systems” in the project DEPRADDEM (Degradation and Prognosis Modeling for Maintenance Decision Making), the European Integrated Project DYNAMITE (Dynamic Decision in Maintenance), and the international project DEPEN-IMPRO (Modeling Policies for the improvement of Production Systems’ Dependability). He is involved in industrial projects with EDF, DCNS, ALSTOM. He is member of the French DIAG 21 Association ([www.diag21.com](http://www.diag21.com)) and of the PHM Society.

**Dr. Maxime MONNIN** is a Research Engineer at PREDICT Company where he is responsible of the R&D Team. In 2007, he received his PhD from the University of Valenciennes. This research funded by the French Procurement Agency (DGA), was conducted in collaboration with NEXTER and the Nancy Research Center for Automatic Control (CRAN), and addressed system of systems availability modelling and simulation. His research interests focus on fleet-wide health management. He is member of the French DIAG 21 Association ([www.diag21.com](http://www.diag21.com)) and of the PHM Society.

**Dr Flavien PEYSSON** is a Research Engineer at PREDICT company, France, with the experience in the development of Diagnostic/Prognostic strategies for PHM systems. He received his M.S. in Industrial Engineering from the Marseilles University Polytechnic School, and earned his PhD, in 2009, from the Aix-Marseilles University, France. His current work is on the development of a Prognostic Framework based on experience and data analysis. He is member of the French DIAG 21 Association ([www.diag21.com](http://www.diag21.com)) and of the PHM Society.

**Dr. Jean-Baptiste LEGER** is CEO and co-founder, in 1999, of the PREDICT company, France. He is graduated from Nancy University, France and his PhD thesis, presented in 1999, was on Formal Modelling Framework for Proactive Maintenance Systems mainly based on Monitoring, Predictive Diagnosis and Prognosis. He has more than 17 years of experience on CBM and PHM systems. His experience includes fault detection and isolation, condition monitoring, fault tolerant control, prognostic, health management, intelligent maintenance and e-maintenance. He is currently working on formal approach of CBM, PHM and Therapy Systems. He is member of the management board of the French DIAG 21 Association ([www.diag21.com](http://www.diag21.com)) and of the PHM Society.



## Author Index

- Allen, David L., 249  
Alonso, Juan J., 396  
An, Dawn, 204, 224  
Argenna, Garrett, 111  
Armstrong, Jeffrey B., 134
- Balaban, Edward, 396  
Banerjee, Avisekh, 449  
Banks, Jeffrey, 122  
Bartram, Gregory, 265  
Bechhoefer, Eric, 63  
Bednar, Jonathon, 122  
Biswas, Gautam, 152  
Bodden, David, 437  
Bole, Brian, 175  
Boubakar, Lamine, 17  
Boucher, Bruno, 63  
Boukabache, Hamza, 215  
Bregon, Anibal, 71, 416  
Brown, Douglas, 163  
Brown, Nathan K., 362  
Brzuskiewicz, Steve, 437  
Byington, Carl, 111
- Camci, F., 147  
Cascio, Dennis, 437  
Celaya, Jose R., 152, 377, 427  
Cerda, Matias, 1  
Choi, Joo-Ho, 204, 224  
Clements, N. Scott, 437
- Daigle, Matthew, 71, 326, 416  
Darr, Duane, 163  
Do Van, Phuc, 256  
Du, Xinyu, 83
- Edwards, Patrick, 437  
Escriba, Christophe, 215  
Evans, John W., 285
- Fischer, Kenneth, 122  
Fourniols, Jean-Yves, 215
- Gallo, Luis D. Jr, 285  
Galvao, Roberto K. H., 458, 464  
Gerez, Valerio, 46  
Gleeson, Bill, 437  
Goebel, Kai, 152, 175, 326, 377, 427  
Goel, Alok, 449  
Gomes, Joao P. P., 458, 464  
Gordon, Grant, 388
- Gouriveau, Rafael, 17
- He, Jingjing, 377  
Hickenbottom, Chris, 388  
Hines, J Wesely, 311  
Hu, Chao, 191
- Irving, Phil, 28  
Iung, Benoit, 256
- Jennions, Ian K., 28, 147  
Johnson, Preston, 469
- Kaminskiy, Mark P., 285  
Karakas, M., 147  
Kim, Byoung U., 437  
Kim, Kyusung, 93  
Kim, Nam H., 204, 224  
Klein, Renata, 183  
Kulkarni, Chetan S., 152  
Kumar, Amar, 449  
Kunst, Neil, 437
- Lacaille, Jerome, 46  
Laskwoski, Bernard, 163  
Leao, Bruno P., 464  
Lebold, Mitchell, 122  
Lee, Wook-Ryun, 191  
Leger, Jean-Baptiste, 476  
Letourneau, Sylvian, 103  
Levrat, Eric, 256  
Li, Jiaming, 369  
Liu, Datong, 336, 343  
Liu, Yongming, 276, 377  
Long, Bing, 319  
Lu, Tsai-Ching, 249  
Luo, Yue, 336
- Mackos, Nicholas, 111  
Mahadevan, Sankaran, 265  
Masson, Patrice, 103  
Medina-Oliva, Gabriela, 476  
Modest, Christian, 233  
Monnin, Maxime, 476  
Morse, Jefferey, 163  
Mulligan, Kyle R., 103  
Muskopf, Paul, 362  
Mylaraswamy, Dinkar, 93, 388
- Naveed, Abdul, 369  
Niculita, Octavian, 28

Olivares, Benjamin, [1](#)  
Orchard, Marcos E., [1](#)

Palladino, Andrew, [111](#)  
Parthasarathy, Girija, [93](#)  
Parthasarthy, Girija, [292](#)  
Pecht, Michael, [319](#), [336](#)  
Peng, Tishun, [377](#)  
Peng, Xiyuan, [336](#), [343](#)  
Peng, Yu, [336](#), [343](#)  
Peysson, Flavien, [476](#)  
Pflumm, Scott, [122](#)  
Pisu, Pierluigi, [353](#)  
Placet, Vincent, [17](#)  
Poll, Scott, [9](#)  
Putic, Mateja, [362](#)

Quaegebeur, Nicolas, [103](#)

Ramasso, Emmanuel, [17](#), [302](#)  
Reichard, Karl, [122](#)  
Reimann, Johan, [111](#)  
Rodrigues, Leonardo R., [53](#), [458](#)  
Rounds, Matthew, [437](#)  
Roychoudhury, Indranil, [71](#), [416](#)

Saha, Bhaskar, [369](#)  
Salman, Mutasim, [83](#)  
Salman, Mutasim A., [249](#)  
Saxena, Abhinav, [326](#), [369](#), [377](#), [427](#)  
Schmitigal, Joel, [111](#)  
Serir, Lisa, [302](#)  
Sevkli, M., [147](#)

Sharp, Michael E., [311](#)  
Silva, Jorge F., [1](#)  
Simon, Donald L., [134](#)  
Srivastava, Alka, [449](#)  
Stempnik, Joseph, [122](#)

Teubert, Christopher, [9](#)  
Thielecke, Frank, [233](#)

Uluyol, Onder, [93](#), [292](#)

Vachtsevanos, George, [175](#), [369](#)  
Vasan, Arvind S. S., [319](#)  
Vianna, Wlamir O. L., [464](#)  
Vohnout, Sonia, [437](#)  
Voisin, Alexandre, [256](#), [476](#)

Wadham-Gagnon, Mathew, [63](#)  
Wagemans, Roy, [437](#)  
Wang, Pingfeng, [191](#)

Xu, Yong, [343](#)

Yang, Jian, [276](#)  
Yoneyama, Takashi, [53](#), [458](#), [464](#)  
Yoon, Joung T., [191](#)  
Youn, Byeng D., [191](#)

Zedek, Sabeha, [215](#)  
Zerhouni, Nouredine, [17](#), [302](#)  
Zhang, Wei, [276](#)  
Zhang, Xian, [353](#)  
Zhang, Yilu, [83](#), [249](#)