System-wide Health Management

Raj Mohan Bharadwaj,

Onder Uluyol

Acknowledgment: "This material is based upon work supported by NASA Award Number NNL09AD44T and the Department of Energy under Award Number DE-EE0001368."

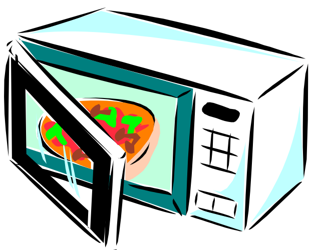
Honeywell

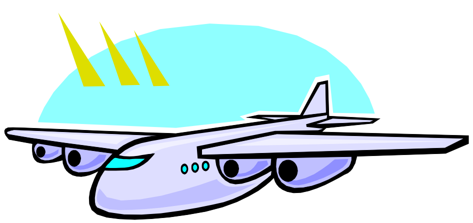
Outline

- Definition: System-wide Heath Monitoring
- Theory of SHM
  - Features
  - Where does system level come from
  - Types of Analysis
  - Types of Evidences
  - Associated Technologies/Technical Areas
- Example 1: Vehicle Level Reasoning System
- Example 2: Wind farm health management

System - Defined

- What is a System?
  - A *system* is defined as a collection of components, which work together to provide a higher level function
  - "The whole is greater than the sum of its parts"





(Ref, Phil Scandura, http://www.nacet.org/wp-content/uploads/2011/04/Presentation1.pdf)

System-Expanded

- It has
  - Structure- made up of member systems
  - Function(s)
  - Inputs, outputs and states
  - Interconnectivity

components ⇨	*subsystems* ⇨	***system*** ⇨	*fleet* ⇨	*enterprise*
Subassemblies	Navigation	Vehicle		Airline
Electronics	Propulsion			
Sensors	Controls			
Bearings				
Actuators	Power generation			
Motors				
Gearbox	Pitch control	Wind Turbine	Wind Farm	Electrical Grid

Integrated

Distributed

Why do it?

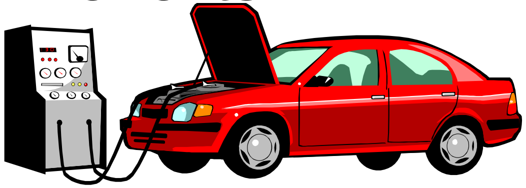
➢ Why?

- Increased complexity
- Higher level of automation

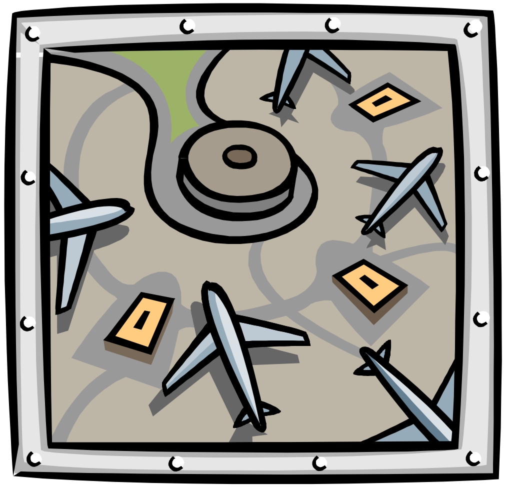
➢ Objectives:

- Safety
- Economics:
  - Aftermarket services
  - Availability
  - Improve customer experience

➢ Benefits



- Easier/quicker to diagnose
- Fix it right the first time
- Lower cost repairs

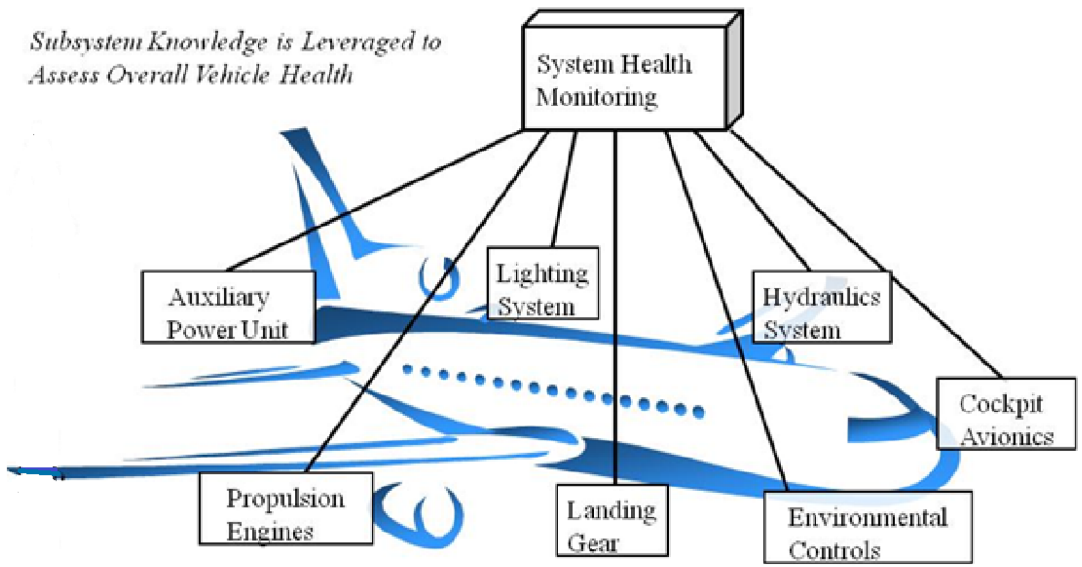


- Faster turn-around times
- Better fleet utilization
- Reduce unscheduled maintenance

(Ref, Phil Scandura, http://www.nacet.org/wp-content/uploads/2011/04/Presentation1.pdf)

System-wide Health Monitoring

- System-wide health monitoring (SHM) detects, infers and manages the health state of the system from the member system health.



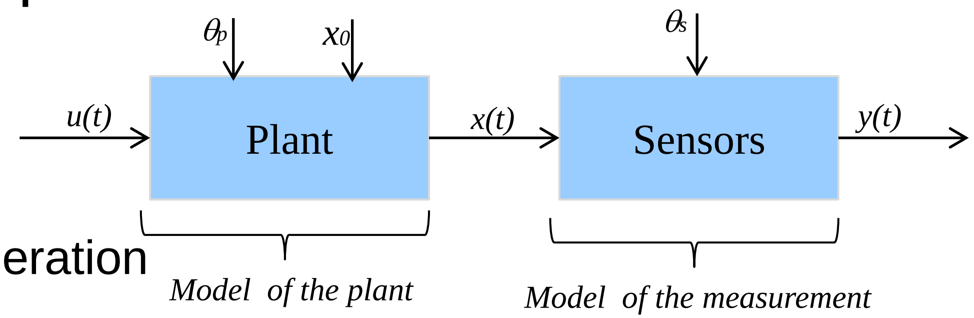
System health monitoring is externally looking. It does not care about individual faults instead has a collective focus.

(Figure, http://www.nacet.org/wp-content/uploads/2011/04/Presentation1.pdf)

Analytical Models

➢ Analytical Models Steps:

- Plant model
- State estimation
- Input-output model
- FDI through residual generation



➢ Limitations

- Unavailability of models for plants and sensors
- Many variables that are available are binary in nature as they represent expert knowledge in rules
- Model inaccuracies

(Ref, Ron J Patton et al., Issues of Fault Diagnosis for Dynamic Systems, Chapter 9)

Structural Models

Structural model and canonical decomposition

- Capture the interconnects and constraints graphically
- Structure of the model is a digraph whose incidence matrix represent link between the variables and constraints

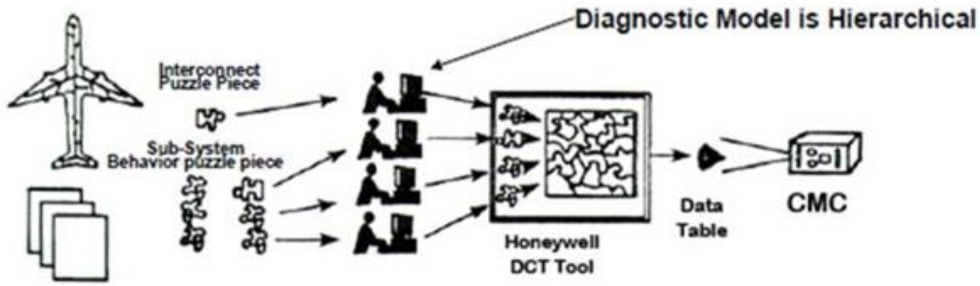
	E1	E2	E3	E4	E5	E6	E7
FM1	1	0	0	0	0	0	0
FM2	0	1	0	0	0	0	0
FM3	1	1	0	0	0	0	0
FM4	0	0	1	1	1	0	0
FM5	0	0	1	0	1	1	1
FM6	0	0	0	1	1	0	1
FM7	0	0	1	0	0	0	0
FM8	0	1	0	0	1	0	0

- Canonical decomposition to diagraph can be applied to simplify
- Boolean signature of the fault, binary word *BS(e)* indicates, the constraints that are violated (*BS(e)*=1) and those that remain true (*BS(e)=0*), in case of fault e;

  BS(OK)=(0,……,0)

(Ref, Ron J Patton et al., Issues of Fault Diagnosis for Dynamic Systems, Chapter 9)

Model-Based Approach



Ack: Felke, NASA presentation 2009

- Each expert is asked to describe only his/her subsystem behavior
- Honeywell modeling tool fits the pieces together and generates a System Reference Model
- The System Reference Model is used by the VIPR inference engine to diagnose aircraft faults

---

(evidence-failure mode relationship)
System Reference Model

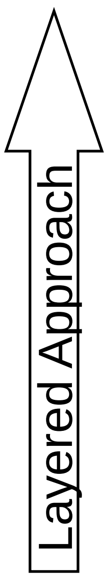
Symptoms (observations) → **Inference (reasoning) Engine** → Fault Conditions, likelihood

(the most plausible explanation)

A model-based approach allows one-time software certification & pushes aircraft specific data to an externally loadable image

How is it done?

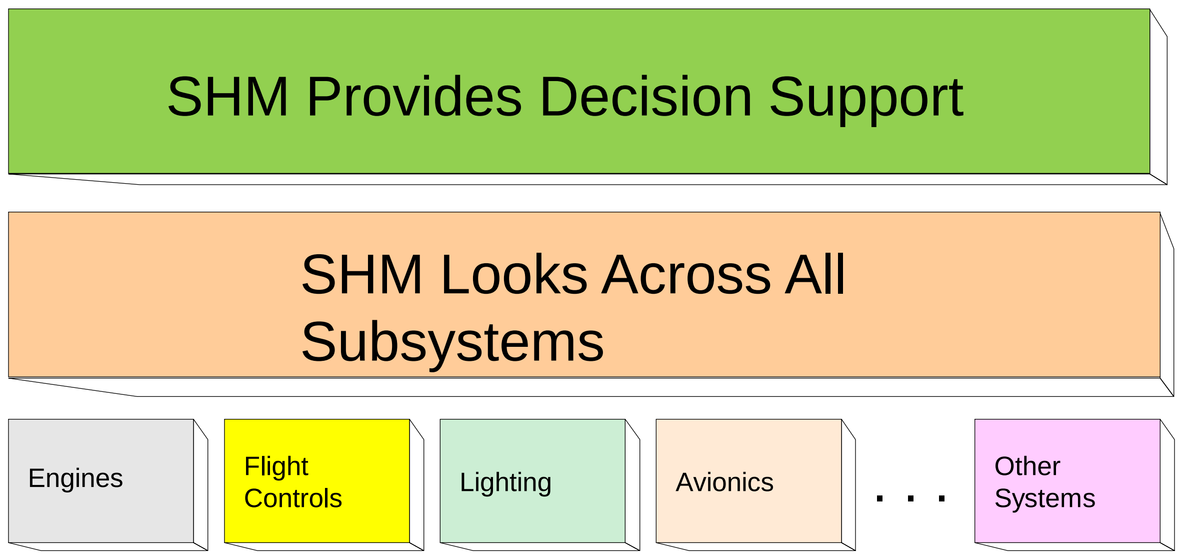
- Useful to view SHM as a series of layers, in which each layer supports the next higher layer, providing a portion of the overall SHM function



System Health Management Provides Decision Support Across the Enterprise

Subsystem Knowledge is Leveraged to Assess Overall System Health

Effective Lower Level Condition Assessment Enables Effective System Health Management

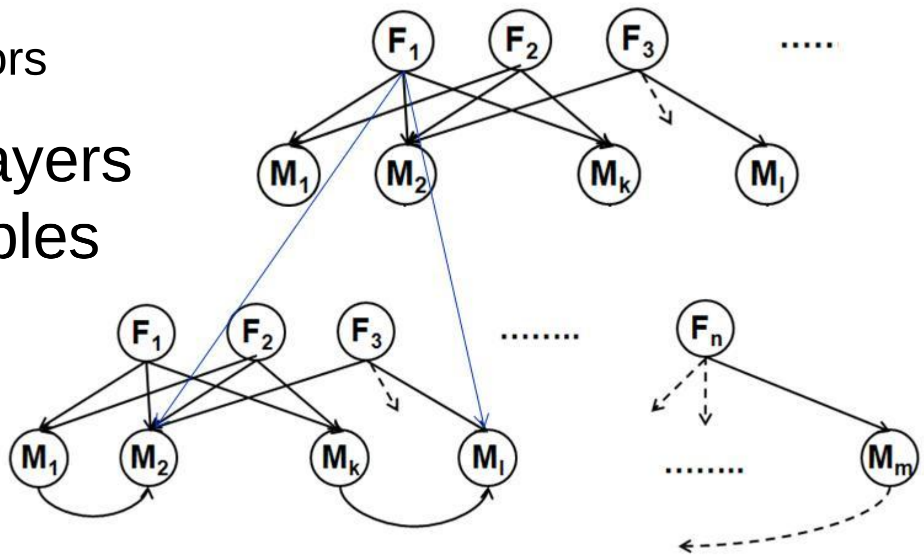


System-wide health monitoring is "designed in" not "added on"

(Ref, Phil Scandura, http://www.nacet.org/wp-content/uploads/2011/04/Presentation1.pdf)

Connectivity

- Member systems are connected loosely
- The relationship is captured in a fixed structure
  - Faults as roots of trees
  - Monitors as leaves
  - Limited Causality between Monitors
- The connectivity between layers is captured such that it enables in isolating faults to a LRU
  - Cascade effects are also captured

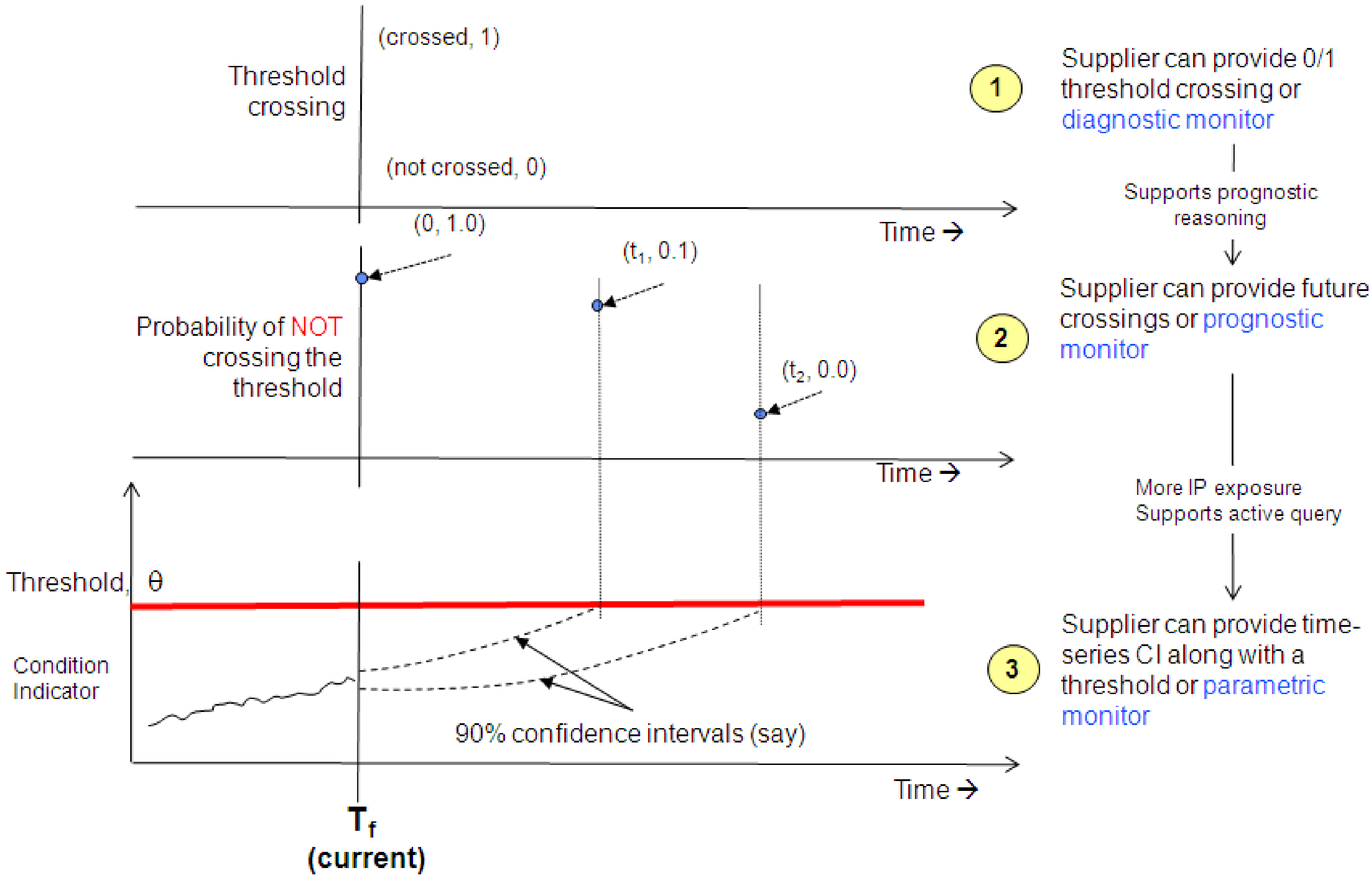


Connectivity is captured in a reference model

Heterogeneous Evidence

- Heterogeneity in data
  - Multiple aircraft
  - Multiple flights under different conditions
  - Failures and adverse events are few and far between
- Time series data
  - Data collected at different rates, different types – have to be merged into common timeline
  - Temporal information has to be abstracted for learning algorithms
- Noisy, uncertainty, and missing data
  - Noisy sensors
  - Unreliable recording/dropouts

Evidence Abstraction



Need to brings in more advanced heterogeneous evidence

Simple (Subsystem) Reference Model …

$$fm_j \longrightarrow e_i$$

$F =$ set of failure modes, $fm_j \in F$

$fm_j = 1 \Leftrightarrow$ failure mode is present

$fm_j = 0 \Leftrightarrow$ failure mode is absent

$E =$ set of evidence. For all $e_i \in E$

$e_i = 1 \Leftrightarrow$ Evidence is present

$e_i = 0 \Leftrightarrow$ Evidence is NOT present

$$d_{ij} \Leftrightarrow P(e_i = 1 \mid fm_j = 1)$$

Probability that the evidence will be present when the failure mode is present in the system

$$\varepsilon_i \Leftrightarrow P(e_i = 1 \mid \forall fm_j = 0)$$

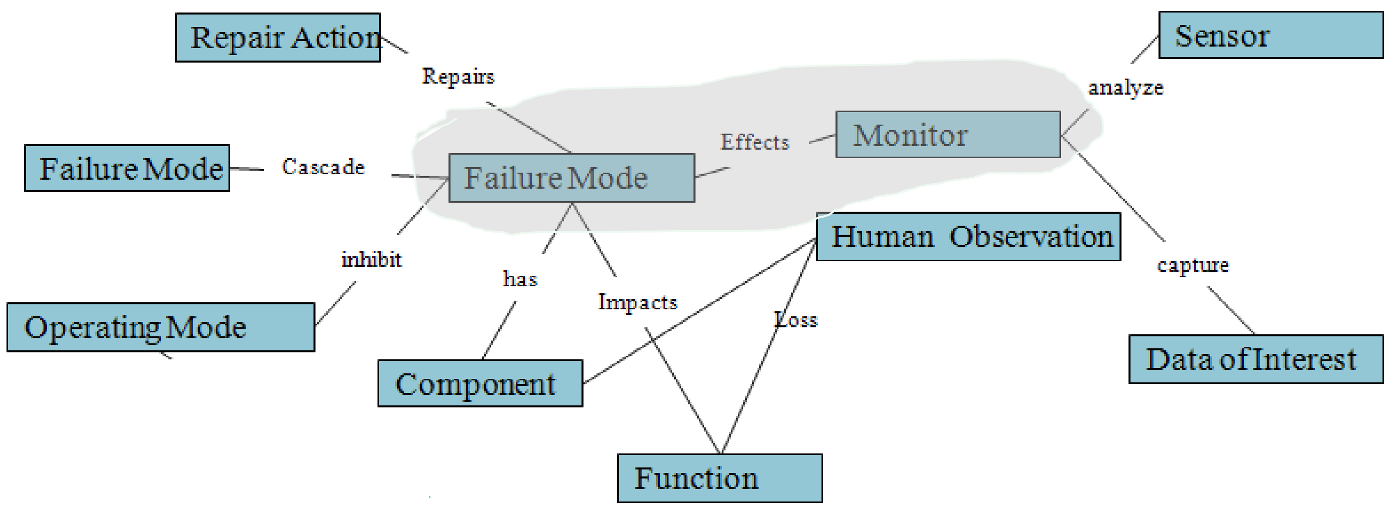
Probability that the evidence will be present when the failure mode is present in the system

There is no notion of time delay in the reference model.

$$AG(e_i) = \{\forall fm_j \mid dij \neq 0\}$$

$$MI(fm_j) = \{\forall e_i \mid dij \neq 0\}$$

System Reference Model



- *Data is provided by individual member system (engines, avionics, landing, etc, …) suppliers and the aircraft model is assembled by an integrator*
- *Accuracy and coverage depends on quality of evidence and completeness of interaction capture*

System Reference Model (static) is a network that captures the specific aircraft configuration

Diagnostic Reasoning Approach

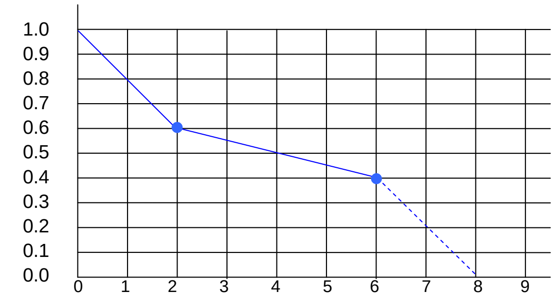
- Capture System relationships
  - Reference Model Database – Consists of two parts:
    - Static – Describes member systems; reusable per system type
      - Definitions, interconnections, failure modes, evidence, action requests, corrective actions
    - Dynamic – Track the current condition of system instance

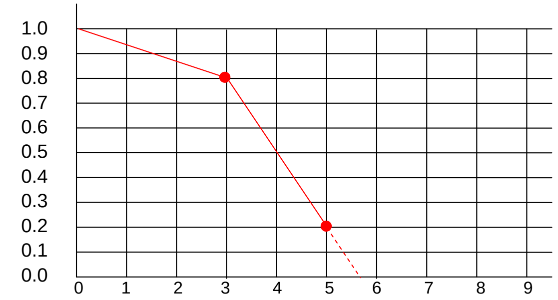
- At run time faults are tracked in "Fault Condition" data structures
  - As evidence arrives, it is assigned to a Fault Condition
  - Fault Conditions group evidence as explained by possible fault hypotheses
  - Fault Conditions are closed after the fault has been corrected
  - Closed Fault Conditions are used to detect repeat and intermittent faults

- Fault Conditions track exactly one fault ("Islands of Single Fault Assumption")
  - If a Fault Condition is believed to be tracking more than fault, it is split into two
  - If two Fault Conditions are believed to be tracking the same fault, they are merged

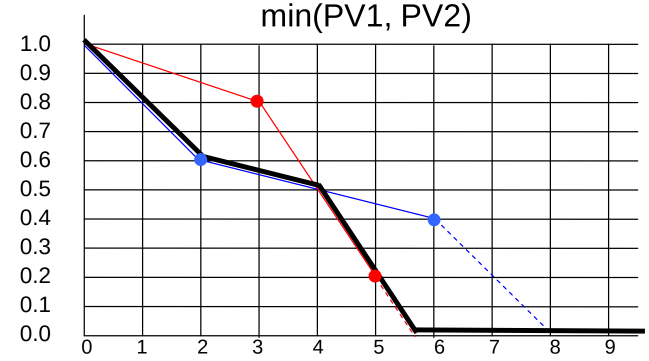
Inclusion of Prognostics

- Extend the diagnostic machinery into prognostics
  - Fault Conditions can contain predicted failure modes and prognostic monitors
  - Prognostic monitors predict the occurrence of a particular diagnostic monitor and whether the DM indicts or exonerates
  - Prognostic monitors contain prognostic vectors (set of time, probability pairs <P, T> – P is probability of *not* failing by time T)
  - Failure modes from the indicated DM are added to the FC's ambiguity set along with the PV
  - PV fusion occurs if the FM is already there





Fused prognostic vector min(PV1, PV2)



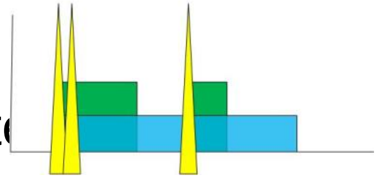
Communications Latency

- Reasoner places a load on the communications system
  - In the absence of faults, reasoner generates few messages
  - The occurrence of a fault triggers a burst of activity
- In safety critical systems such as avionics communication using AFDX and ASCB are periodic and have statically defined schedules for communications
- Reasoner must be statically allocated a slice of communications bandwidth:
  - Narrow slice leads to long latency
  - Wide slice leads to inefficient use of the communications resource

How much bandwidth does SHM need to provide acceptable performance?

Latency and Communication

- Typically Processing time is much shorter than communications time
- Communications time “on the wire” is much shorter than time buffered to send



- Reasoner entity E has two cost parameters
  - Pi : entity’s processing cost for processing an input stimulus and producing its output
  - Ci: entity’s communications cost factor
- For a message of size M bytes, the processing and communications cost for the transaction is:

$$TransactionCost = \frac{Psource + M * Csource}{2} + \frac{Pdestination + M * Cdestination}{2}$$

Goal: for any fault, communications latency should be less than X seconds

Vehicle Integrated Prognostic Reasoner (VIPR)

Acknowledgment: “This material is based upon work supported by NASA Award Number NNL09AD44T and the Department of Energy under Award Number DE-EE0001368.”

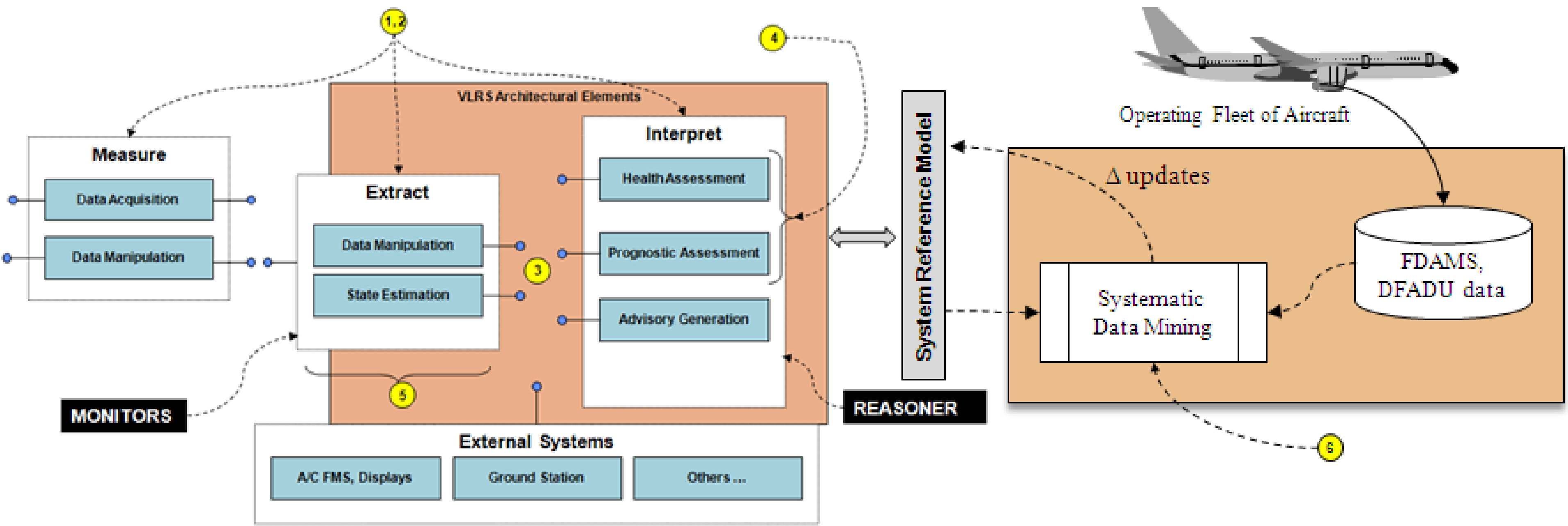
Honeywell

NTSB Safety Incidents

(Ref: Cooper et al., Av Safe Conference, 2009)

- Air France Flight 447 accident on 1st June, 2009 (Bureau d‟Enquêteset d‟AnalysesInterim Report *f-cp090601ae)*
  - Analysis of the series of 24 broadcast maintenance messages concluded that various monitoring processes were triggered, with many of them pointing to an inconsistency in speed measurement
- In-flight upset 154 km west of Learmonth, WA, 7 October 2008 Airbus A330-303 (ATSB Transport Safety Report AO-2008-070 Interim Factual)
  - While cruising at 37,000ft the aircraft autopilot disconnected, various aircraft system failures were indicated. Before the flight crew could deal with them, the aircraft abruptly pitched nose-down and descended 650 ft.
- Loss of Pitch Control During Takeoff, Air Midwest Flight 5481, Raytheon (Beechcraft) 1900D, N233YV, Charlotte, North Carolina, January 8, 2003 (NTSB/AAR-04/01)
  - Post event analysis showed consistent differences in pitch control position values 10 flights before the maintenance check, and the 9 flights after the D6 maintenance check.

Data Driven VLRS+

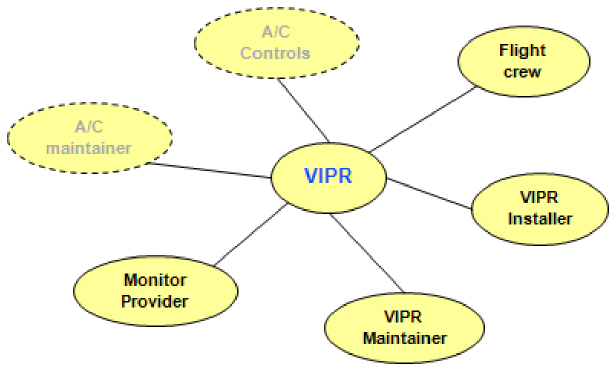


Next Generation VLRS needs to support the following features

- Support temporal and prognostic reasoning
- Active role for fault isolation
- Systematic updates to the reference model using operational data – continual learning

Working with NASA to provide systematic extensions to the field-proven ADMS reasoner to handle next gen safety requirements – called VIPR

User Requirements



	Event Type	Top Level requirements (Flight crew)
Time Evolution	Slow	1. Less important. 2. Important, if and only if it will affect the current flight.
	Fast	1. Very important. Early detection of incipient conditions. 2. …ic control) actions
Impact Propagation	Localized	…g as designed
	Widespread	…usion. …move the evidence.
…ptom …stence	Constant	…rder.
		…tion and establish that intermittency is true. …ot cause may be less important

1. Detect events in real time.
2. If impact is localized, confirm that backup is working as designed
3. Keep track of intermittents

	Top Level requirements (VIPR Installer)
Scalability	1. Separate the reasoning algorithms from aircraft specific configurations. 2. A common code base is easy to validate and makes is easier to certify. 3. Finite set of operations, each of which is bounded computationally.
Deployment	1. Reason… …e. 2. Suppor… …enerating monitors. 3. VIPR sh… …f a monitor provider. 4. Unamb… …tation.
Accuracy	1. Ability t… …s important. 2. Must in… …t can be archived and used as… 3. States a… …' operations 4. Capable of proposing and working with multiple fault hypotheses.

1. Allow member systems to encode proprietary knowledge.
2. Common code base to reduce certification efforts.
3. Work within aircraft HW/SW constraints

VIPR Inputs: Monitors

- Monitor is an observation regarding the presence or absence of evidence e_i

$$m_i = 1 \Leftrightarrow e_i = 1(indicte), \quad m_i = 0 \Leftrightarrow e_i = 0(exonerate), \quad m_i = -1 \Leftrightarrow e_i = unknown$$

- There are several ways of "expressing this observation" at time t_0:

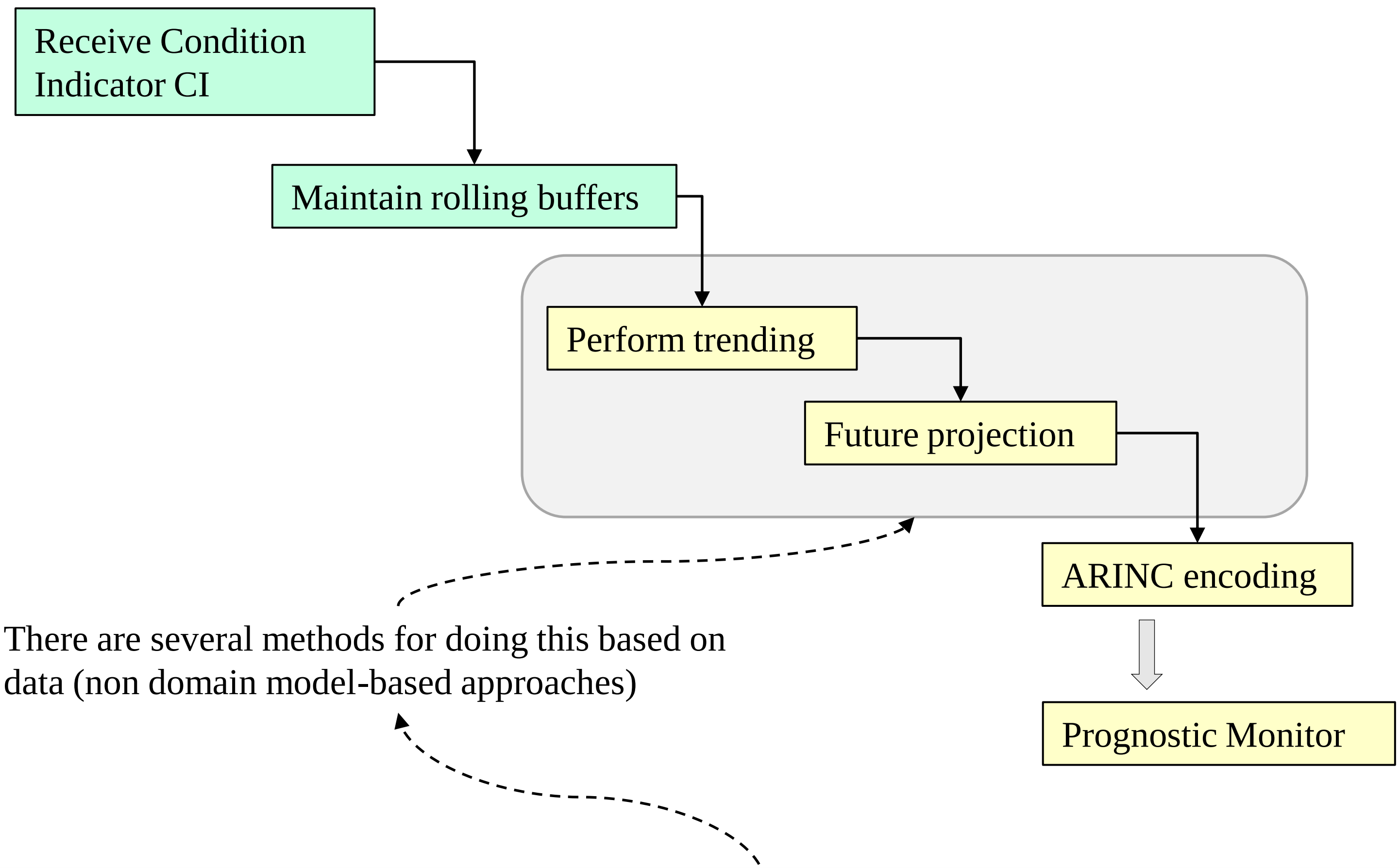
Simplest Diagnostic Monitor: $P(m_i = 1) @ t_0 .Often :P(m_i = 0) @ t_0$ ⇐ State of the art

Prognostic Monitor: $P(mi = 0) @ t_0, t_1, t_2, ...$ ⇐ VIPR Standardizes this

- In later slides we discuss how to generate prognostic monitors …

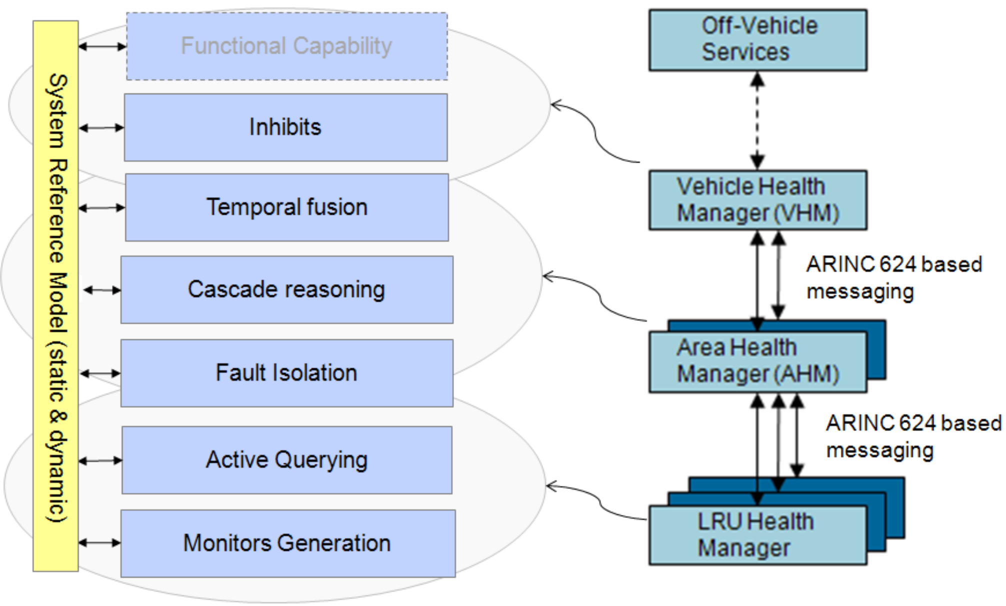
> - *VIPR accomplishment:* 4 mechanism for generating and expressing complex evidence to enable "active participation" to detect incipient events.

Prognostic monitor generation



- *VIPR accomplishment:* Defines four mechanisms for handling progressive, slow and intermittent evolution of an underlying adverse event. VIPR needs to know how to interpret the CI, and NOT how the CI was generated

Layered Computation Architecture



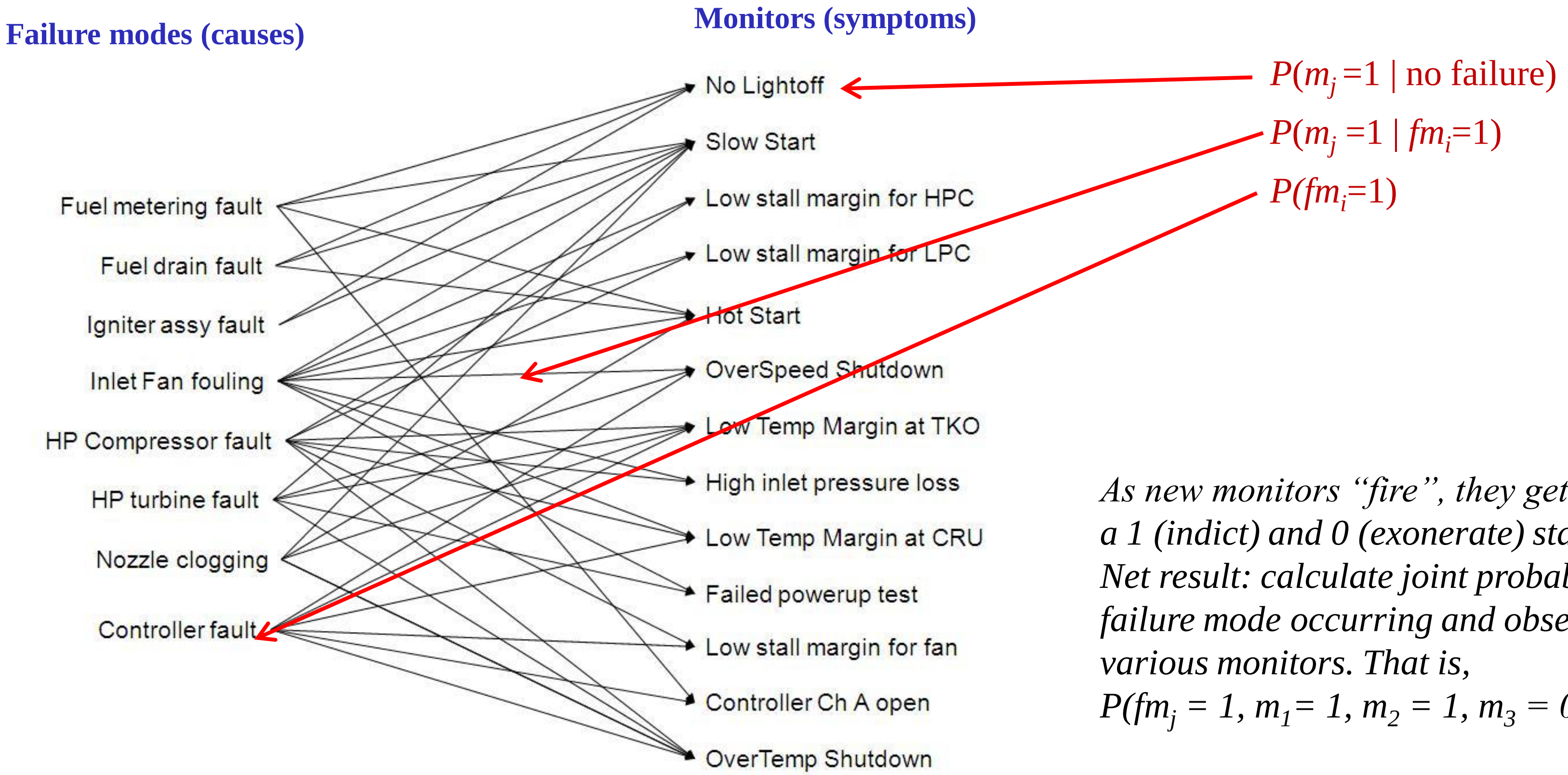
- In an aircraft:
  - A LRU may not be capable of generating monitors
  - VIPR needs to provide computational resource to generate these monitors based on sensor data
  - Hence the need for a LRU health manager tier to support these intensive calculations
  - Area Health Manager does most of the fault isolation
  - Vehicle health manager does inhibits, temporal and functional capability assessment
- Practically:
  - VIPR like any other CBM system needs to buy itself. Customer may only choose one or more functions, rather than the entire thing!

A distributed reasoning architecture allows VLRS to operate within aircraft computation constraints

Three Steps (phase)

- Phase 1: concepts, design, concept of operations
  - Establish initial design and pathway for acceptance within the community, availability of historic data
- Phase 2: detailed design, implementation and validation
  - Demonstration in a simulation environment, tools & methods
- Phase 3: metrics collection
  - Scenario-based cost, prognostic benefit and safety impact metrics calculation

The Reasoner theory

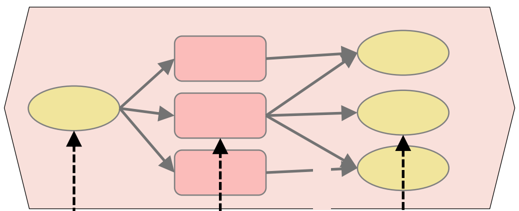


As new monitors "fire", they get assigned a 1 (indict) and 0 (exonerate) state. Net result: calculate joint probability of a failure mode occurring and observing various monitors. That is, $P(fm_j = 1, m_1 = 1, m_2 = 1, m_3 = 0, \ldots)$

Use a noisy-or (Naïve Bayesian update) to calculate the joint probability

Reasoner Engine: States & Operators

Fault Condition FC – VIPR state



Initiating Monitor

Failure modes that could trigger this monitor AG(FC)

Monitors expected to fire if any of the failure mode is active, EoI(FC)

- *Represents a "diagnostic conclusion within VIPR"*
- *Contains an ambiguity set of failure modes*
- *Tracks a single fault i.e. makes a single fault assumption hypothesis*
- *VIPR can contain several fault conditions at any time*

VIPR "state update operators"

Probability update: $P(fm_j = 1, m_1 = 1, m_2 = 1, m_3 = 0, \ldots)$

Isolate: $P(fm_j = 1, \ldots) > \boldsymbol{\delta_I} + P(fm_k = 1, \ldots), \ldots$

Splitting: $P(fm_j = 1, fm_k = 1, \ldots) > \boldsymbol{\delta_S} + P(fm_j = 1, \ldots), P(fm_k = 1, \ldots)$

Merging: $EoI(FC_1) = EoI(FC_2)$

FM Addition: $AG(FC) \leftarrow AG(FC) + fm_j$,

FM Removal: $AG(FC) \leftarrow AG(FC) - fm_j$,

Active Query: ? m_i, m_i in $EoI(FC)$

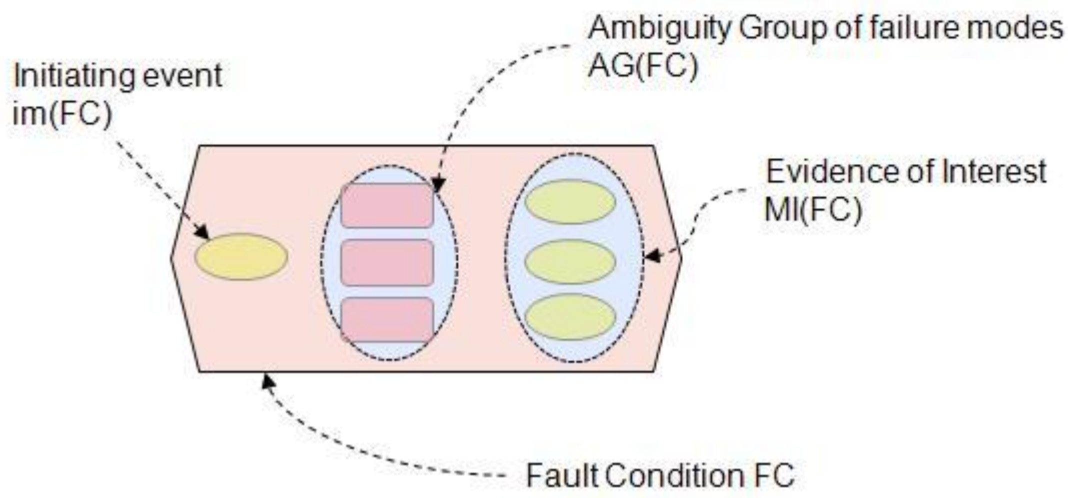
Closing: $P(fm_j = 1, \ldots) < \boldsymbol{\delta_0}$

Ranking: sort($P(fm_j = 1, \ldots)$)

Deletion: $time(P(fm_j = 1, \ldots)) >$ ***NTE***

- *Reasoner can track multiple simultaneous faults*
- *Update is "event driven" – triggered by arrival of new monitor*
- *A finite (deterministic) set of operators per update cycle*
- *Contains several user-tunable knobs or constants to trade-off sensitivity (highlighted in bold)*

VIPR States: Fault Condition



Notation: FC

- FC is a data structure with the following elements:
  - A fault condition has one and only one initiating evidence; it is merely an element of set E
  - The fault condition contains an ambiguity group of failure modes. The ambiguity group contains elements from the set F.
  - The fault condition has a property called evidence of interest.

- *VIPR accomplishment:* FC is a necessary and sufficient “data packet” to support hierarchical reasoning. An ARINC 624 protocol to communicate a compact conclusion to the CAS and Maintainer.

Interpretation of FC

- A given FC represents a hypothesis that any of failure mode in the AG(FC) is occurring within the system.
  - Depending on how many failure modes may be occurring, an FC can assert several hypothesis regarding failure modes occurring in the system.

Given fault condition FC such that $AG(FC) = \{fm_1, fm_2, fm_3\}$.

$$\textit{FC Null Hypothesis: } \wedge(fm_1 = 0, fm_2 = 0, fm_3 = 0)$$

$$\textit{FC Single Fault Hypothesis: } \veebar(fm_1 = 1, fm_2 = 1, fm_3 = 1)$$

$$\textit{FC Two Fault Hypothesis: } \veebar(fm_1 = 1 \wedge fm_2 = 1,\ fm_2 = 1 \wedge fm_3 = 1, fm_3 = 1 \wedge fm_1 = 1)$$

$$\textit{FC Three Fault Hypothesis: } fm_1 = 1 \wedge fm_2 = 1 \wedge fm_3 = 1$$

VIPR aims for "islands of single fault assumptions". Hence it splits a 2-fault hypothesis into two FC each with one 1 fault hypothesis.

Reasoner Main Loop

Firing or occurrence of monitors drives the reasoner process (event driven).

The state update step is a sequence of four steps:

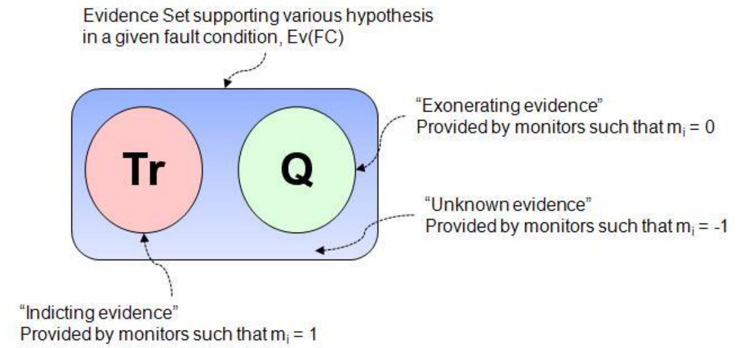
1. Evidence allocation.
2. Probability and Likelihood update. Updating the log likelihood for all fault hypothesis asserted by $FC \in X(n)$.
3. Apply Tests. Application of tests for fault isolation and false alarm suppression.
4. Message Passing. The net outcome from the above steps is a new heath state or $X(n+1)$.

Likelihood calculation: naïve Bayesian update

We define the evidence function of a fault condition $Ev(FC)$ as follows:

$$
\begin{aligned}
\text{FC evidence: } & Ev(FC) \Leftrightarrow \{Tr, Q\} \\
\text{FC evidence(indicting): } & Tr \Leftrightarrow m_i = 1, \forall i | e_i \in \mathrm{EoI}(FC) \\
\text{FC evidence (exonerating): } & Q \Leftrightarrow m_i = 0, \forall i | e_i \in \mathrm{EoI}(FC)
\end{aligned}
$$

$$\text{Relative likelihood } L(h_p) \Leftrightarrow \frac{P(h_p, Tr, Q)}{P(NF, Tr, Q)}, \quad h_p \in \Theta(FC) \qquad (6)$$



Applying the chain rule and using fact that FC evidence Tr and Q are independent of each other, we get:

$$L(h_p) = \frac{P(h_p)}{P(NF)} \frac{P(Tr|h_p)}{P(Tr|NF)} \frac{P(Q|h_p)}{P(Q|NF)}, \quad h_p \in \Theta(FC) \qquad (7)$$

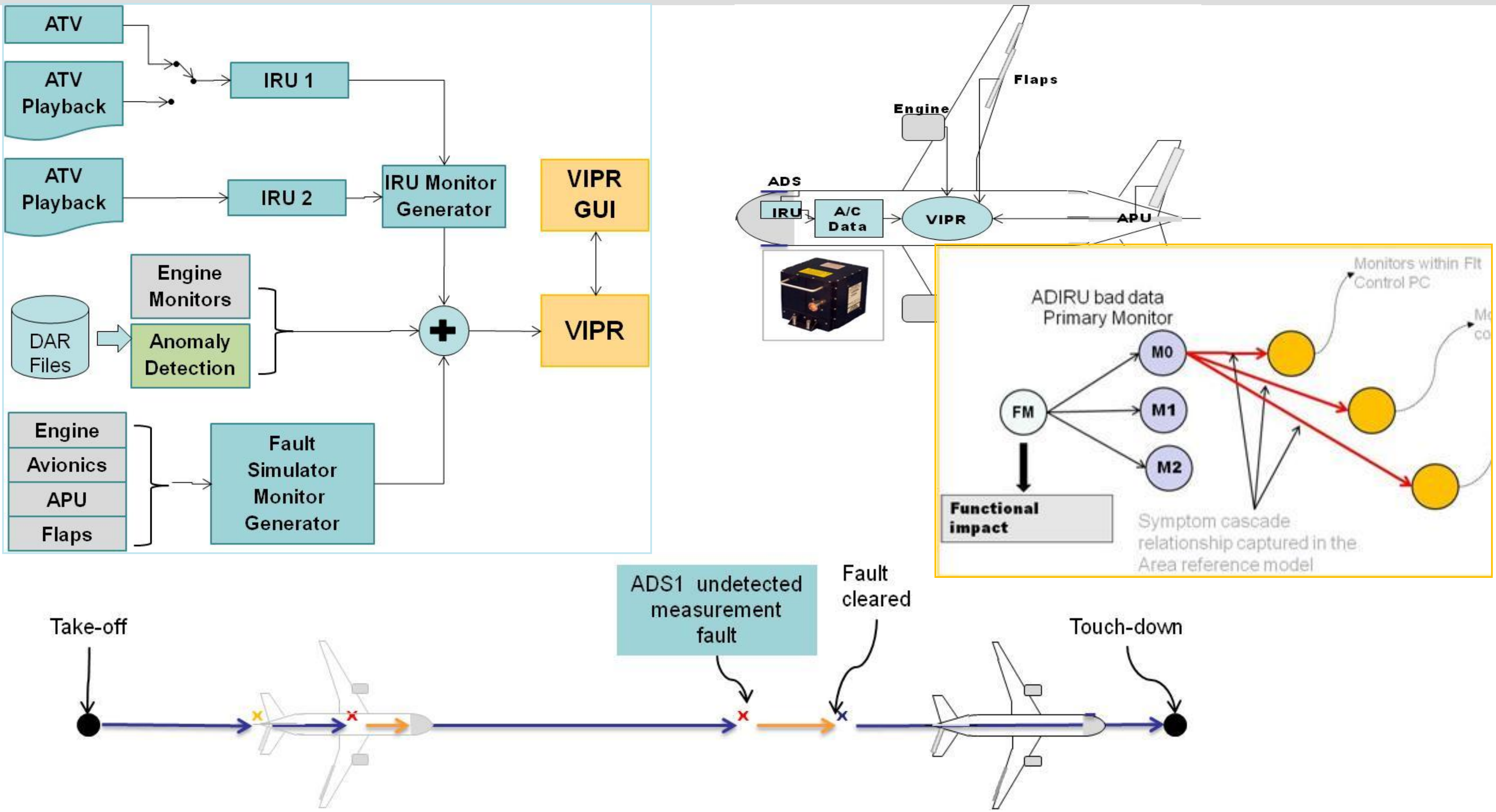
$$\frac{P(Tr|h_p)}{P(Tr|NF)} = \prod_{\substack{i|m_i=1,\\ e_i \in \mathrm{EoI}(FC)}} \left(\frac{P(m_i = 1|h_p)}{\epsilon_i} \right), \quad h_p \in \Theta(FC)$$

$$\frac{P(Q|h_p(FC))}{P(Tr|NF)} = \prod_{\substack{i|m_i=0,\\ e_i \in \mathrm{EoI}(FC)}} \left(\frac{P(m_i = 0|h_p)}{(1-\epsilon_i)} \right), \quad h_p \in \Theta(FC)$$

The DELTA increment each time a new monitor associated with the i'th evidence occurs or fires

- *VIPR accomplishment:* A $O(N^2)$ algorithm for updating the likelihood. N = number of elements in each FC.

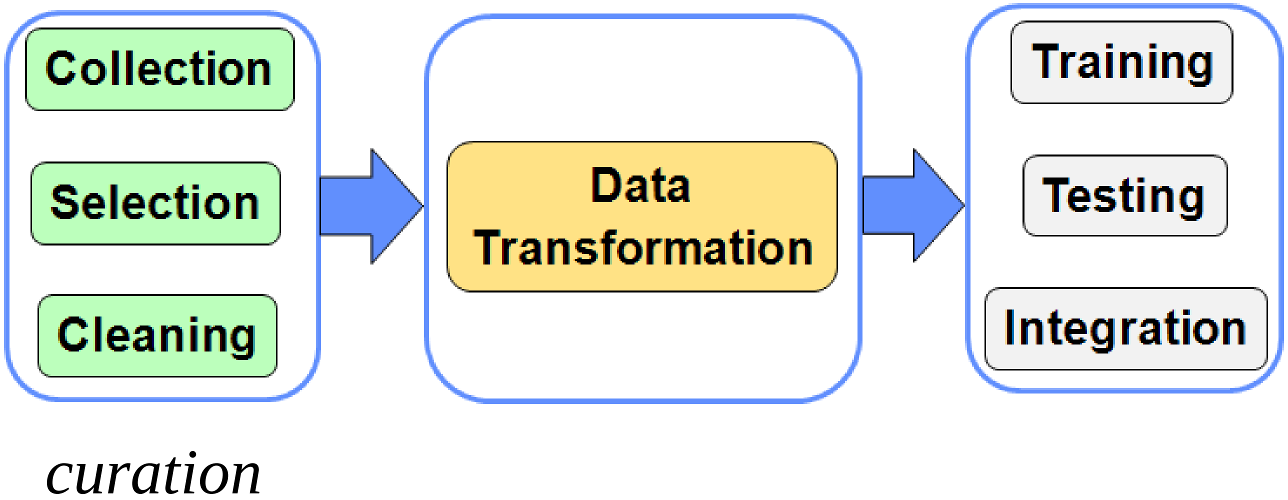
HIL Integrated Demo-ADS Safety Incidence



- Multiple data streams are integrated in the VIPR demo
- Lref6-ATV demo shows need capture subsystem relationships within the reference model to fault prevents cascade

Goals of the Data Mining Work

- Demonstrate a systematic approach for continual improvement in the VIPR performance
  - Exploit data from past adverse event occurrences and known fault situations
  - Semi-automated data-driven processes
  - Selective Data mining operations



Aircraft Data

- We instrumented aircrafts to record 180+ parameters at 1, 2, 4, 8 and 16 Hz over the entire the flight cycle
  - Fleet consisted of 30+ identical airplanes and flies 2—3 flights each day
  - Access to 3000+ consecutive flights

Event Date	Safety Incident	Event Date	Safety Incident	
	Loss of oil and engine shutdown		Pilot error	✕
	Vibration, engine shutdown, Turbine damaged		Hydraulic leak, smoke in the cabin	?
	Over speed temperature and engine shutdown		Incipient ice formation	
	Hydraulic leak. Take off aborted		Runway incident. Hit a pole	✕
	Intermittent engine on fire. Traced to fuel problems		Runway incident, hit a catering truck	✕
	False alarm of engine on fire. Fuel leakages			

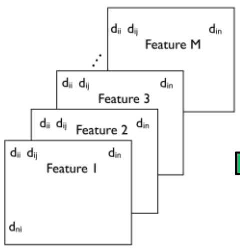
ASIAS (FAA's safety reporting website) incidents and 1—16 Hz aircraft parametric data surrounding these incidents

Anomaly Detection

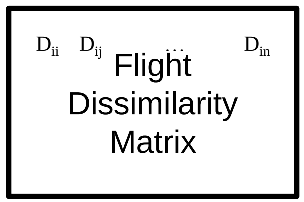
➢ Offline Analysis

- Derive nominal model using entire flight data
  - Method: K-complexity measure:

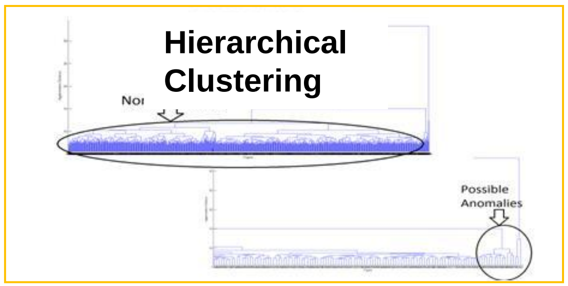
$$NCD(x,y) = \frac{C(xy) - min\{C(x),C(y)\}}{max\{C(x),C(y)\}}$$



NCD Metric

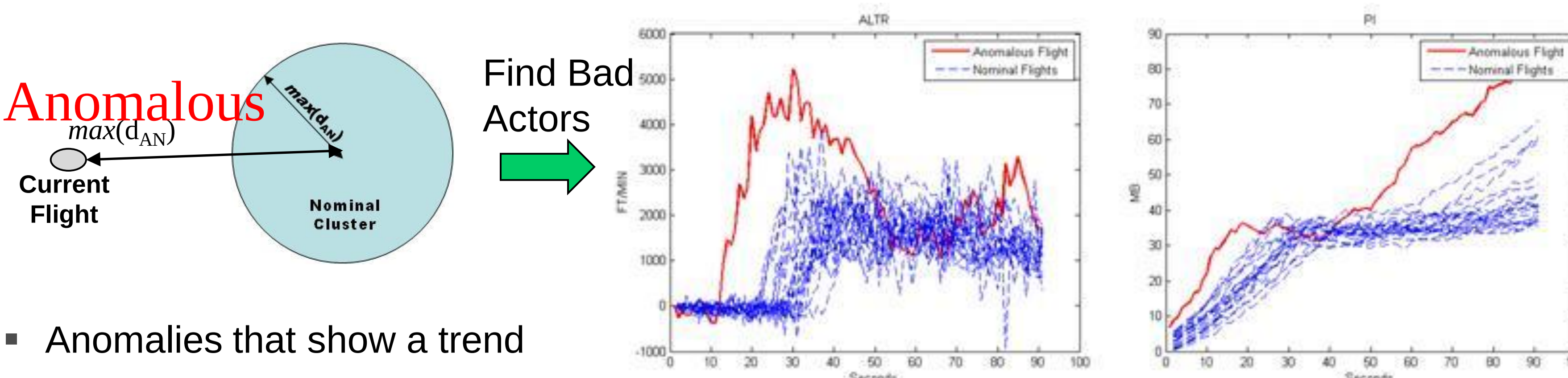


Euclidean Metric



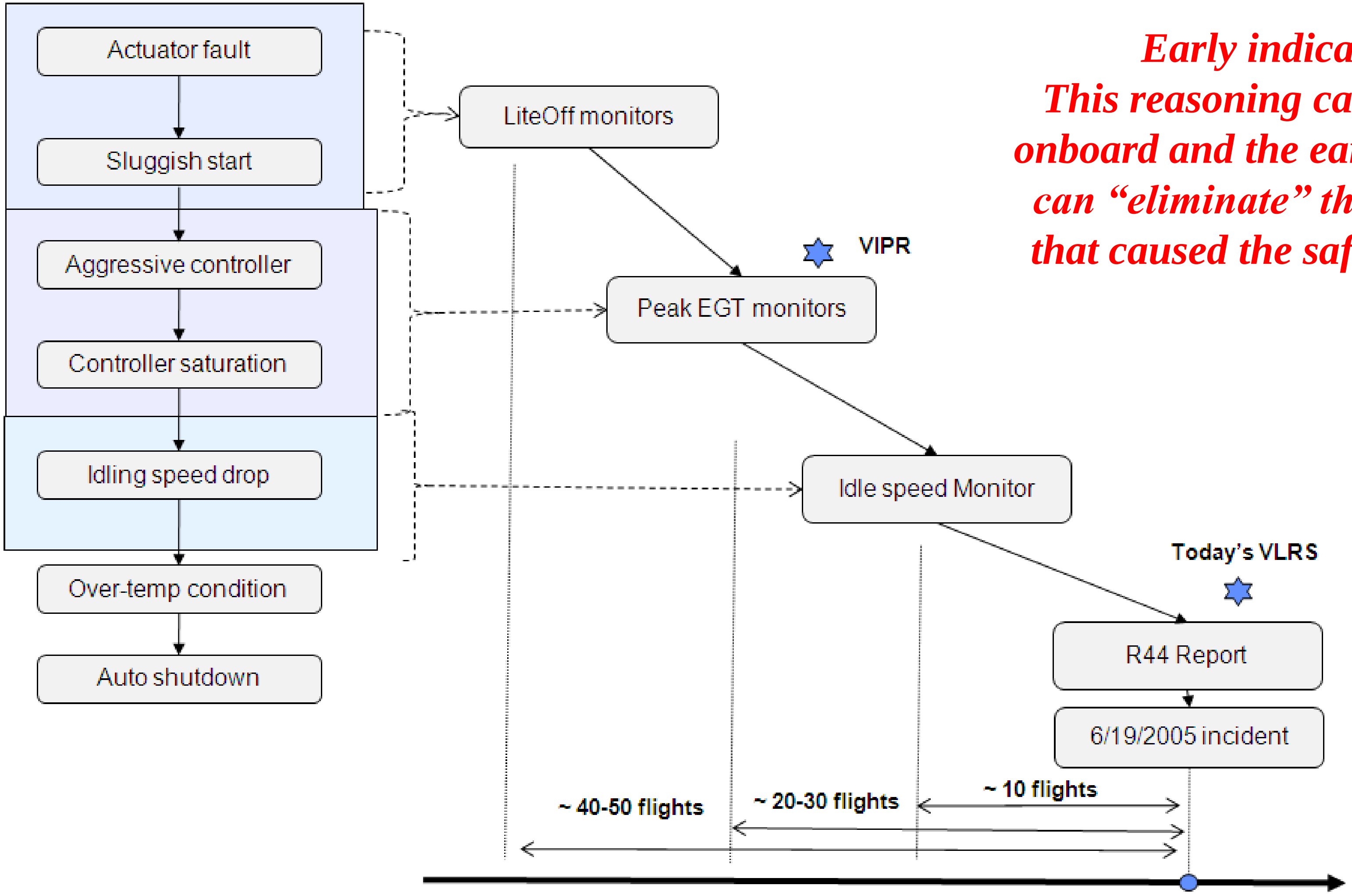
➢ Online Analysis

- On line detection for ACMF Function: Compare Individual Flight Data to Nominal



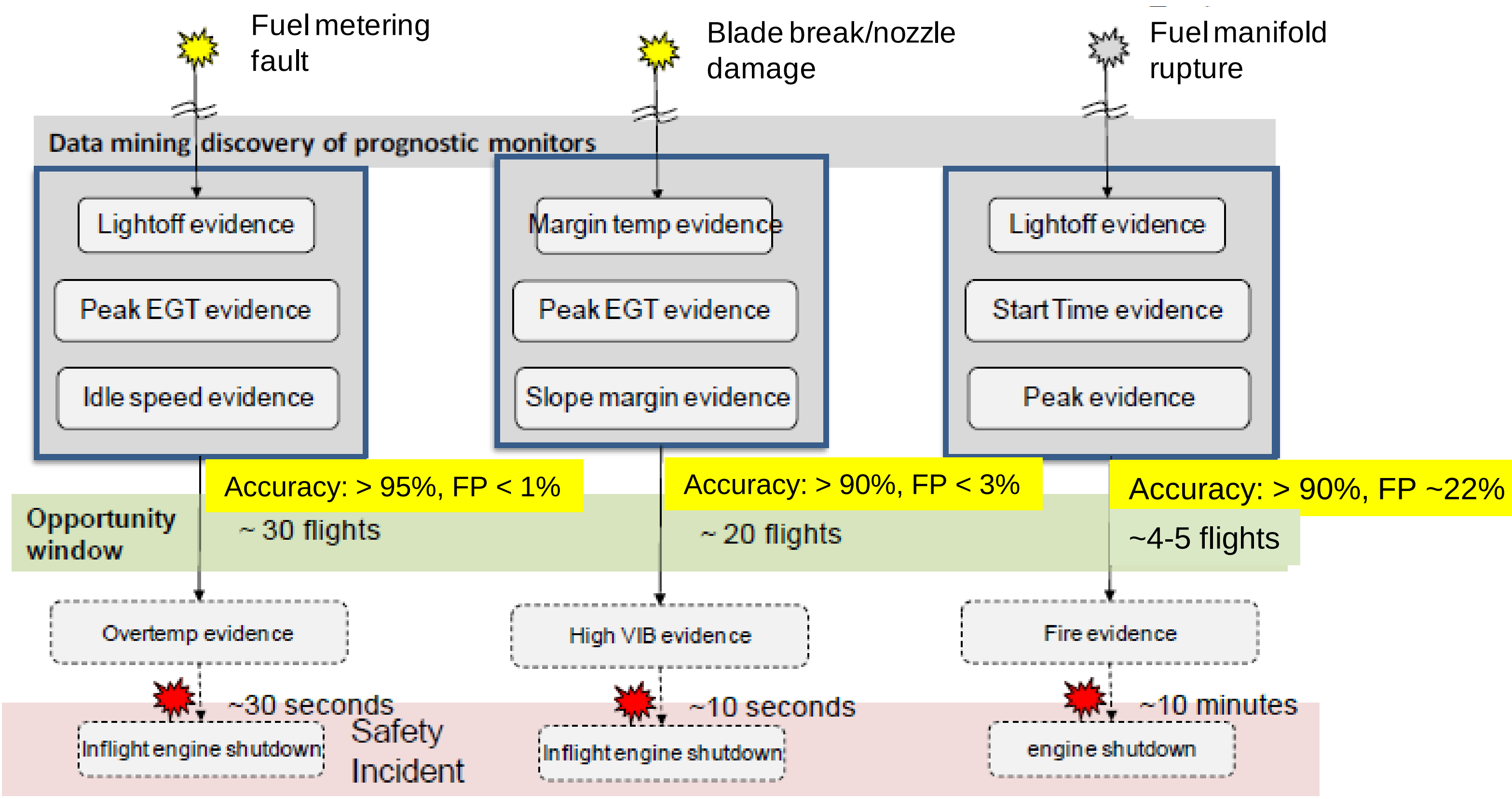
- Anomalies that show a trend linked to faults/safety incidents can be incorporated into the VIPR reference model.

Impact on Safety



Early indication
This reasoning can be done onboard and the early indication can "eliminate" the root cause that caused the safety incident.

Safety Incidence Avoidance



- Demonstrated VIPR capabilities wrt diagnostic, and prognostic reasoning
- Demonstrate VIPR capability for safety incidence avoidance by incorporating monitors discovered through data mining.

VIPR detect impending in-flight engine shutdown

Closing Remarks

- Vehicle level reasoner is aimed at:
  - Improving aircraft safety due to enhanced monitoring and reasoning about the aircraft' s health state
  - Operational cost savings by enabling Condition Based Maintenance (CBM)
- In this talk, we outlined the next gen VLRS – namely VIPR
  - Trade space: user requirements and safety drivers, delta-increments from baseline to realize the advanced functions of VLRS
  - Reasoning steps: defined the steps for evidence aggregation, fault hypothesis management, using an abductive reasoning framework
  - Role of Data mining: defined algorithmic approach to update the capture new information

Wind Farm Health Management

Onder Uluyol

Acknowledgment: “This material is based upon work supported by NASA Award Number NNL09AD44T and the Department of Energy under Award Number DE-EE0001368.”

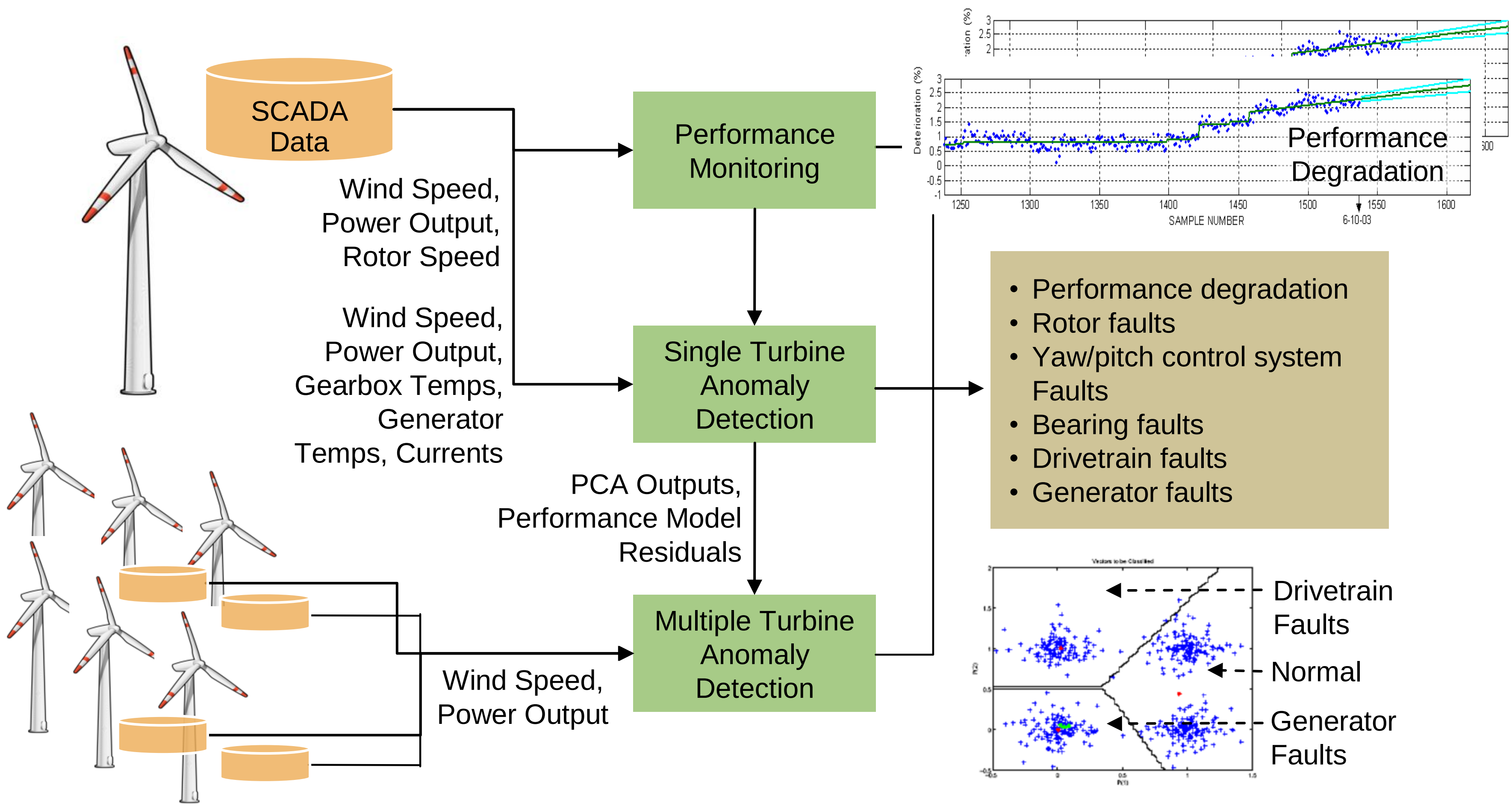
Honeywell

Need for Data Analysis and Monitoring

- Wind turbines operate continuously in severe environments; in remote locations; need frequent scheduled maintenance
- High cost of un-detected failure and repair, and lost production time
- Tremendous growth in the wind industry – large growth in number of older wind turbines
- Performance issues with aging
  - Availability can decrease 1% per year after year 5
  - O&M costs rise with age
  - Performance degradation reduces capacity factor
- Monitoring and data analysis
  - Enables condition based maintenance and performance tune-up
  - Catches failures before reaching catastrophic, or secondary damage stage
  - Extends asset life
  - Keeping assets working at initial capacity factors
  - Increasing availability by reducing routine maintenance, and predicting failures for optimum repair planning

Condition Based Rather than Hours Based Maintenance Reduces O&M costs

SCADA data based Performance Monitoring



Under-analyzed SCADA data is valuable in performance and fault monitoring

Large wind farm data set

- One example used in the project – SCADA data procured from a large wind farm operator
- This data was obtained using a OPC connection to the existing turbines, and pushing the data into a historian
- Wind turbine SCADA data in the OPC historian is from three different wind parks, with different wind turbine manufacturers, differing numbers of parameters & varying naming conventions across the parks
- Organization of these points is flat – several thousand points in a single farm, without hierarchy, e.g. using a naming convention:

X002_II_P2_T083_T_GEAR_BEAR	T083 Temperature Gear Bearing

- The problem of mapping from a flat hierarchy to a standard set of meta data is common to other domains (e.g. building control systems)

Example Meta Data Generated

Parent-Child relationships derived from [Tag Name]

ParkID	TurbineID	PointType	Tag Name
X002_II_P2	T078	Active Power Warning	X002_II_P2_T078_P_ACT_WARNING
		aintenance Time End of Last Month	X002_II_P2_T078_MT_ULM
		aintenance Time Total Since Commisioning	X002_II_P2_T078_MTT
		Blade 1 Actual	X002_II_P2_T078_BL1_ACT
		eather Out Time Total Since Commissioning	X002_II_P2_T078_WOTT
		Frequency	X002_II_P2_T078_FREQ
		Hydraulic Pressure	X002_II_P2_T078_HYD_PRES
		I phase A	X002_II_P2_T078_I_A
		I phase B	X002_II_P2_T078_I_B
		I phase C	X002_II_P2_T078_I_C
		Operating (include in calculations and warnings)	X002_II_P2_T078_OPERATING
		Power Actual	X002_II_P2_T078_P_ACT
		Power Factor	X002_II_P2_T078_PF
		Resetable Fault Condition	X002_II_P2_T078_FAULTED
		State Fault	X002_II_P2_T078_STATE_FAULT
		Temperature Ambient	X002_II_P2_T078_T_AMB
		Temperature Bearing A	X002_II_P2_T078_T_BEAR_A
		Temperature Bearing A Warning	X002_II_P2_T078_T_BEAR_A_WARNING
		Temperature Bearing B	X002_II_P2_T078_T_BEAR_B
		Temperature Bearing B Warning	X002_II_P2_T078_T_BEAR_B_WARNING
		Temperature Bearing Shaft	X002_II_P2_T078_T_BEAR_SHAFT
		Temperature Bearing Shaft Warning	X002_II_P2_T078_T_BEAR_SHAFT_WARNING
		Temperature Gear	X002_II_P2_T078_T_GEAR
		Temperature Gear Bearing	X002_II_P2_T078_T_GEAR_BEAR
		Temperature Gear Bearing Warning	X002_II_P2_T078_T_GEAR_BEAR_WARNING
		Temperature Gear Warning	X002_II_P2_T078_T_GEAR_WARNING
		Temperature Gen 1	X002_II_P2_T078_T_GEN_1
		Temperature Gen 1 Warning	X002_II_P2_T078_T_GEN_1_WARNING
		Temperature Gen 2	X002_II_P2_T078_T_GEN_2
		Temperature Gen Cooling	X002_II_P2_T078_T_GEN_COOL
		Torque actual	X002_II_P2_T078_M_ACT
		Tower acceleration	X002_II_P2_T078_A_TOWER
		Tower Acceleration Warning	X002_II_P2_T078_A_TOWER_WARNING
		Voltage A N	X002_II_P2_T078_U_A_N
		Voltage B N	X002_II_P2_T078_U_B_N
		Voltage C N	X002_II_P2_T078_U_C_N
		Weather Out Time End of Last Month	X002_II_P2_T078_WOT_ULM
		Windspeed	X002_II_P2_T078_V_WIN
		Windspeed Warning	X002_II_P2_T078_V_WIN_WARNING

PointType is derived from the [Description], but is not identical to the [Description]

Other Data Sets

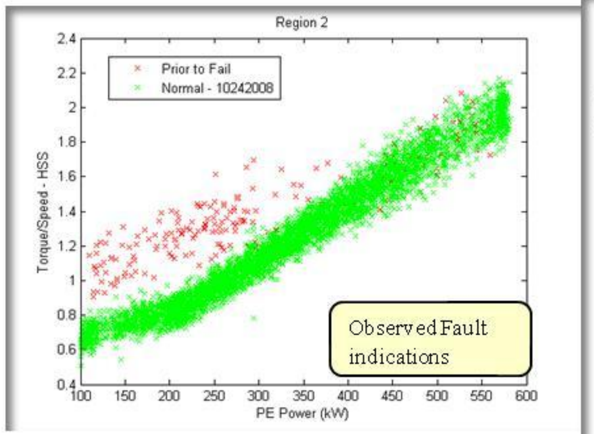
- Data Set II
  - a mid-power wind turbine
  - supplies power to a university campus
  - recently came out of 5-year warranty
  - SCADA data is available in 10 minute and hourly intervals for 2006-2010.

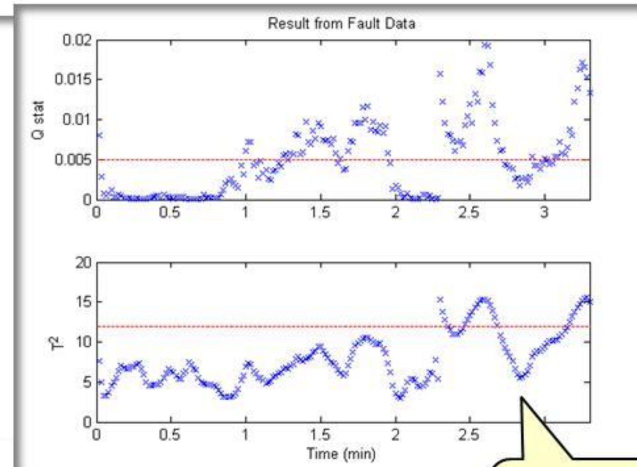
- Data Set III
  - collected from a small, reconditioned wind turbine
  - provides power to the operator’s office building in an urban setting
  - data is available at 1-min sampling rate.

- Data Set IV
  - a mid-power wind turbine
  - Installed to test new control schemes
    - CART-2 : 2 blade turbine. Data collected at 100 Hz
    - CART-3 : 3 blade turbine. Data collected at 400 Hz
  - 88 measurements stored in 10 minutes block

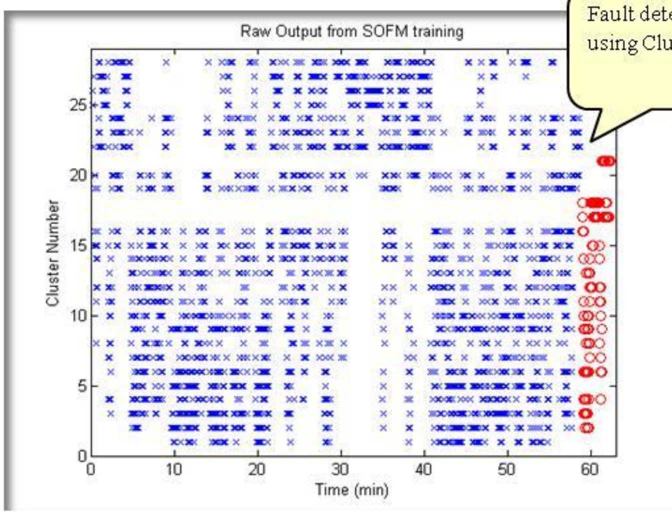
Single Turbine Anomaly Detection

NREL Control Advanced Research Turbine - 2



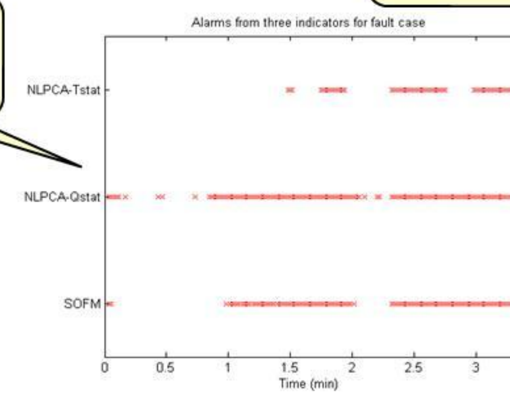


Fault detection using PCA



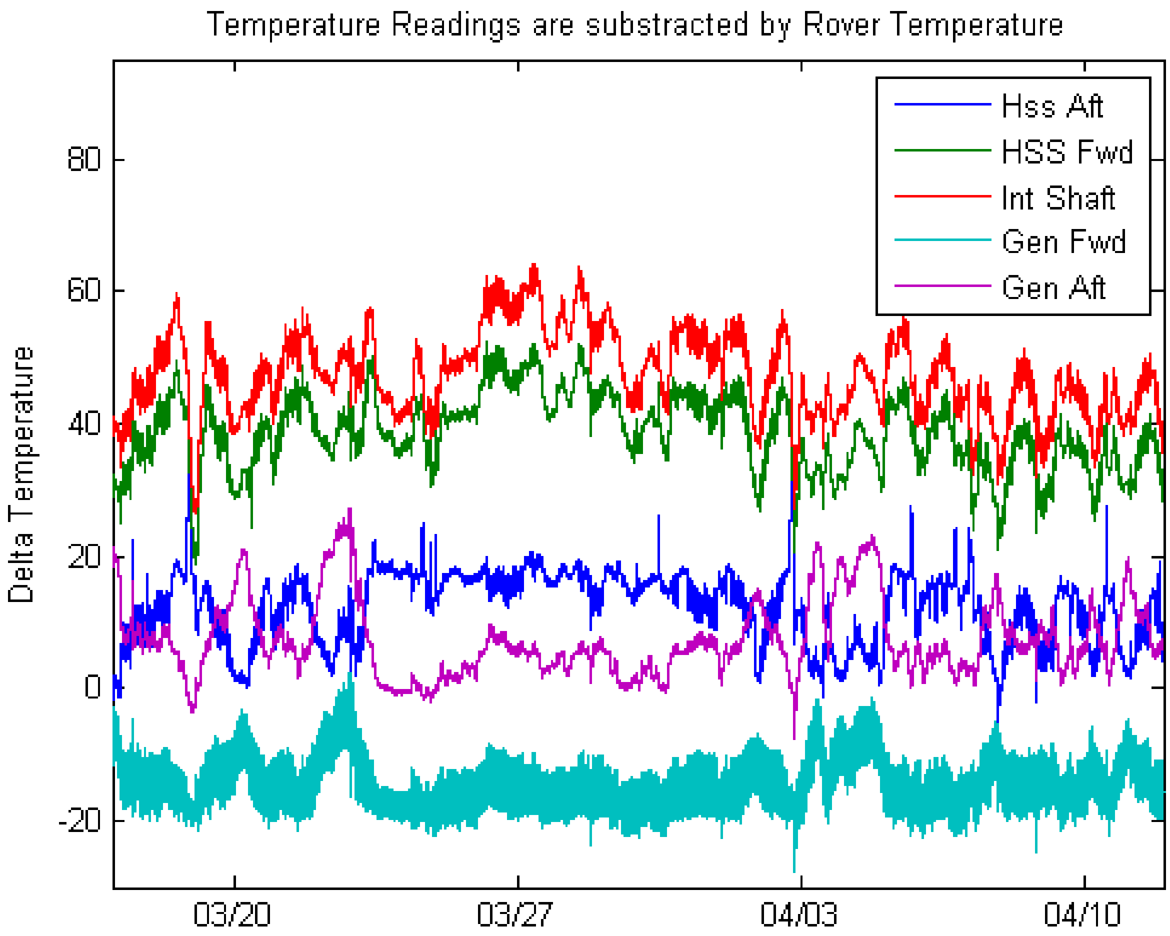
Fault detection using Clustering

Illustration of alarm from 3 indicators



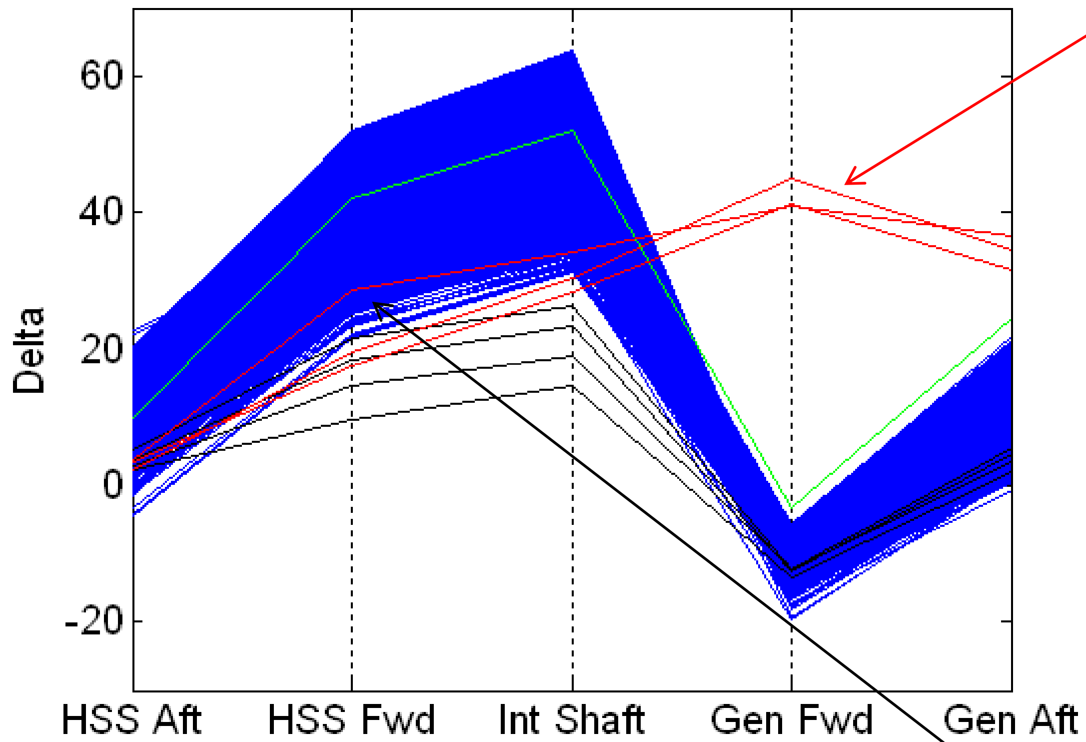
Successful application of anomaly detection algorithms to SCADA data

Normalization of Temperature - Difference



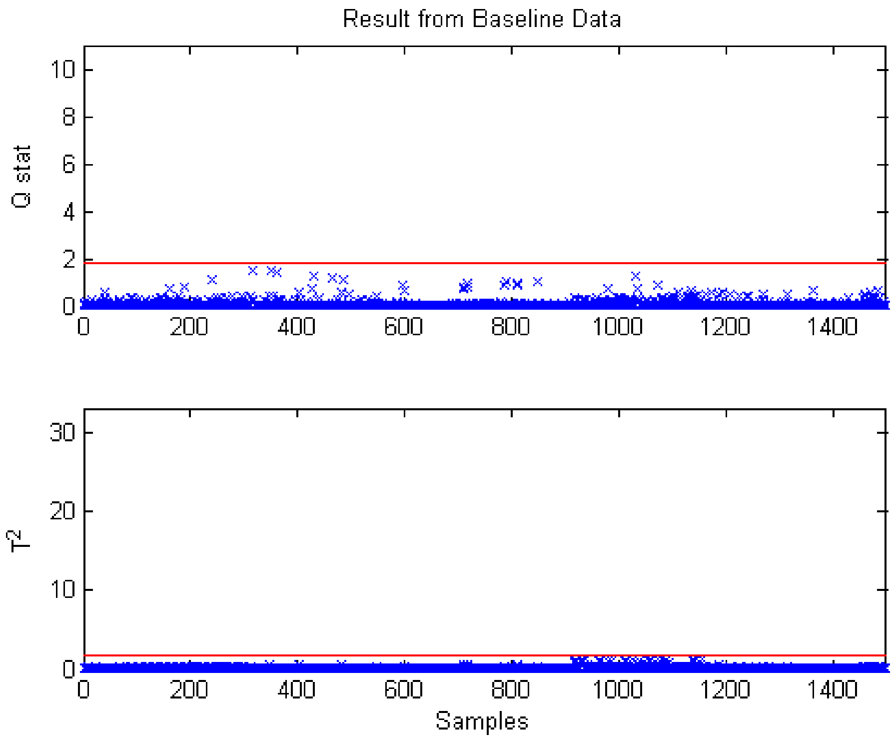
Parallel Coordinate Plot

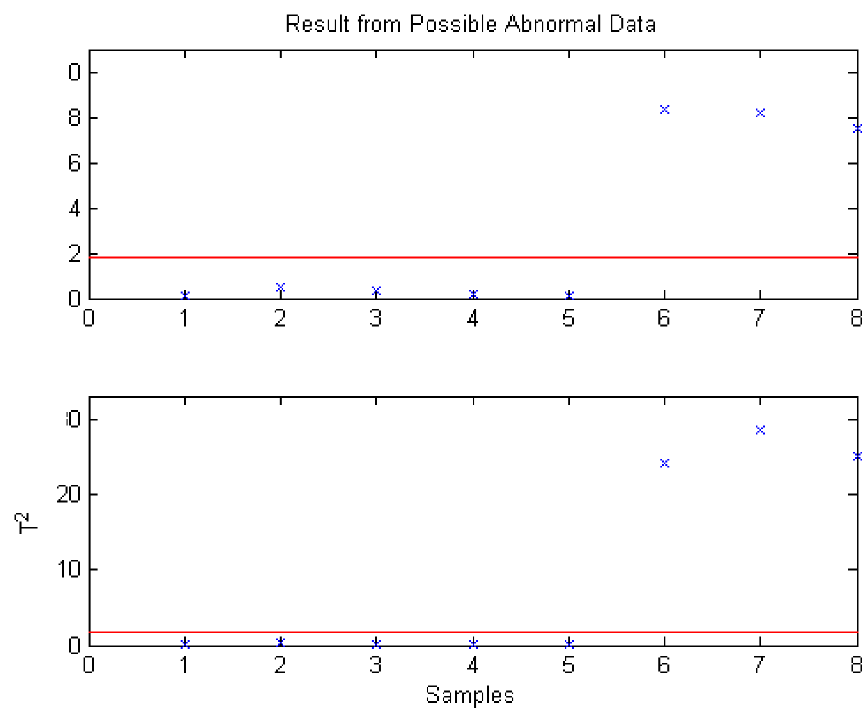
possible abnormal ones
- 07-Jun-2011 18:46:47
- 07-Jun-2011 19:56:48
- 07-Jun-2011 20:17:04



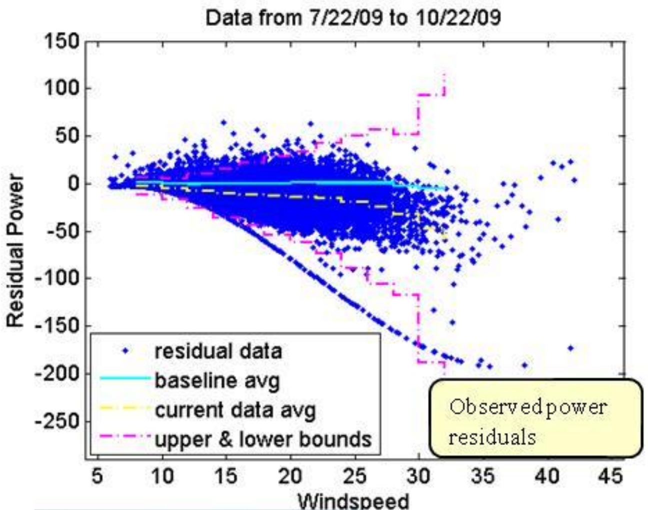
possible abnormal ones
- 12-Apr-2011 21:05:31
- 12-Apr-2011 21:15:35
- 12-Apr-2011 21:25:40
- 12-Apr-2011 21:35:48

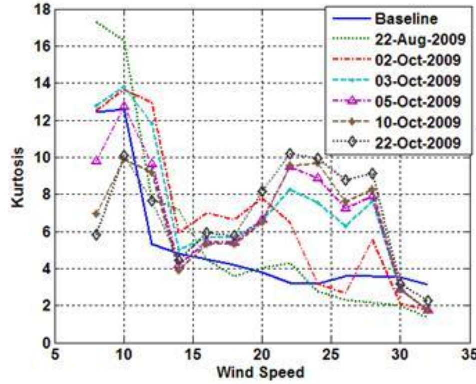
Test Set – baseline and abnormal

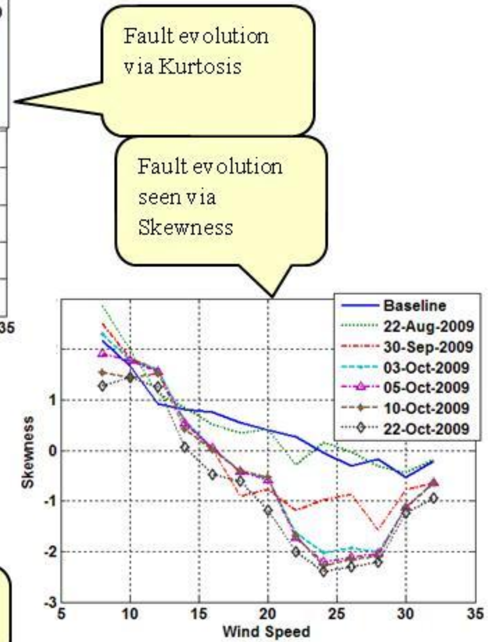


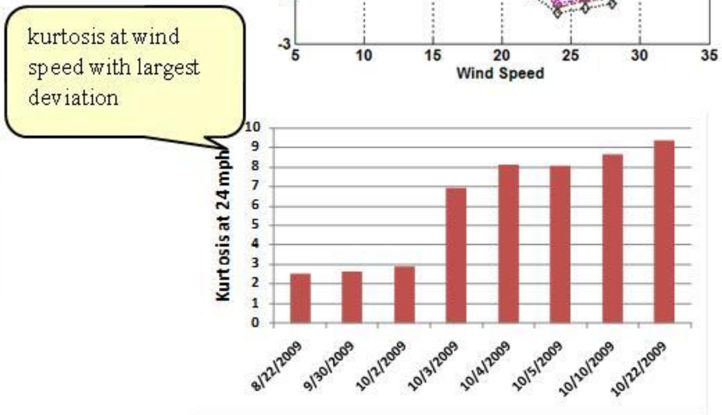


Power Curve Analytic







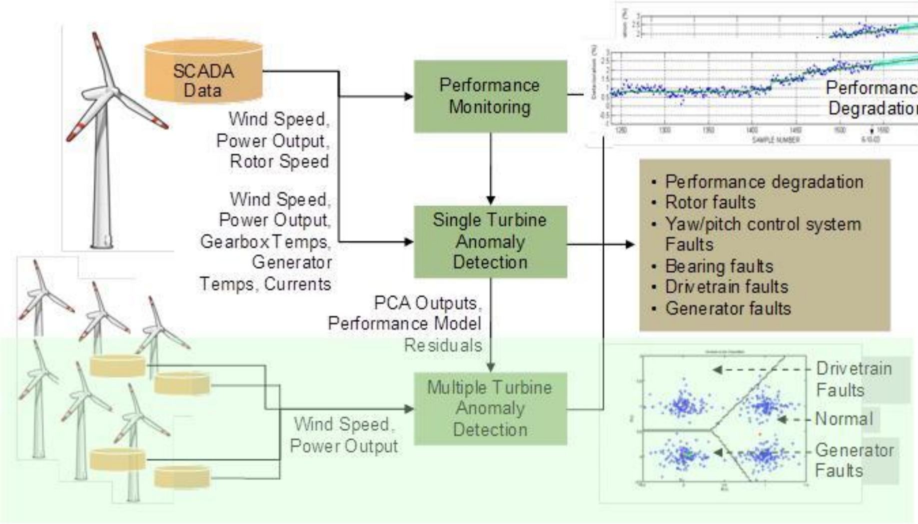


U of MN Morris Vestas wind turbine

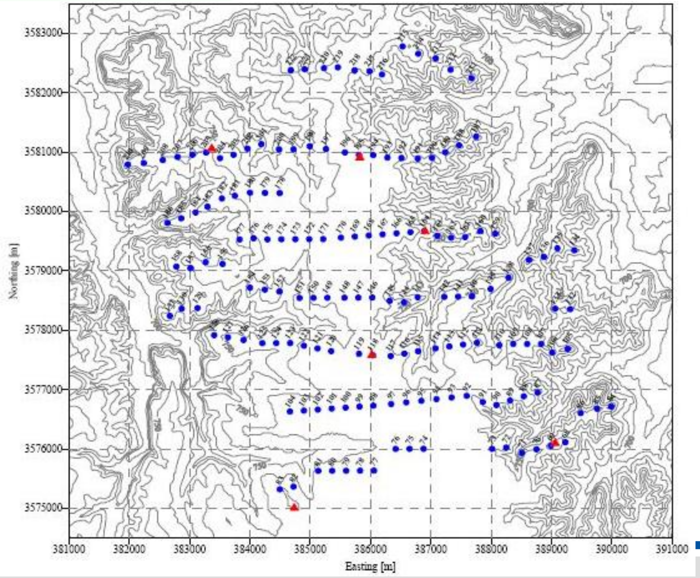
NEG Micon wind turbine at Great River Energy

Simple analytic detected anomaly >20 days in advance of semi-annual maintenance

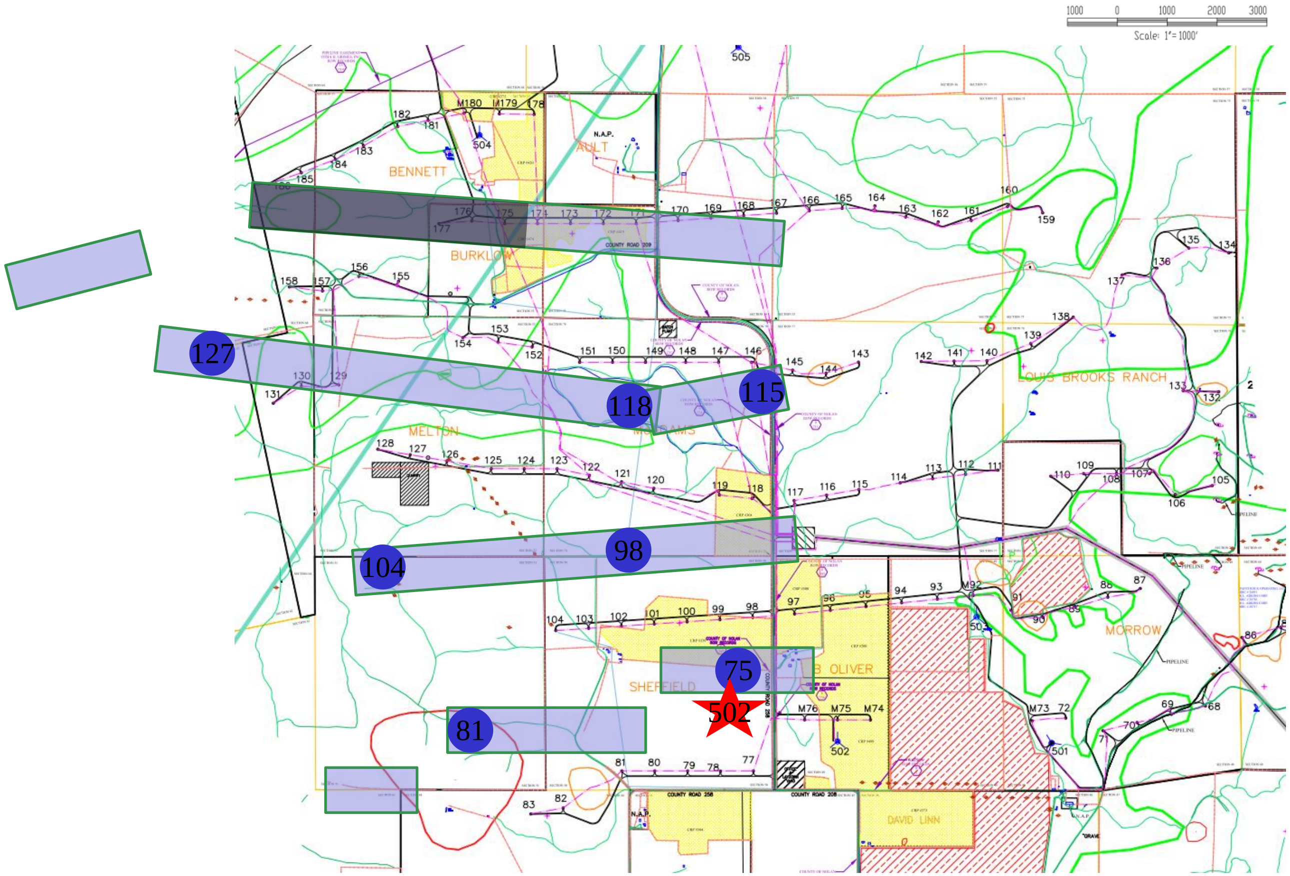
Multi-turbine Analytic



- Obtained archived data from 3 large wind parks
  - the wind parks were equipped with Honeywell–Matrikon data connectivity and historian solutions
  - Very large set of data set with ~16000 tags, 1 year worth of data and 300+ turbines
- GE, Mitsubishi and Micon wind turbines
- Organized data by mapping the tags from a flat layer to a multi-layer meta-data structure
- Matlab OPC Toolbox to connect to the OPC historian

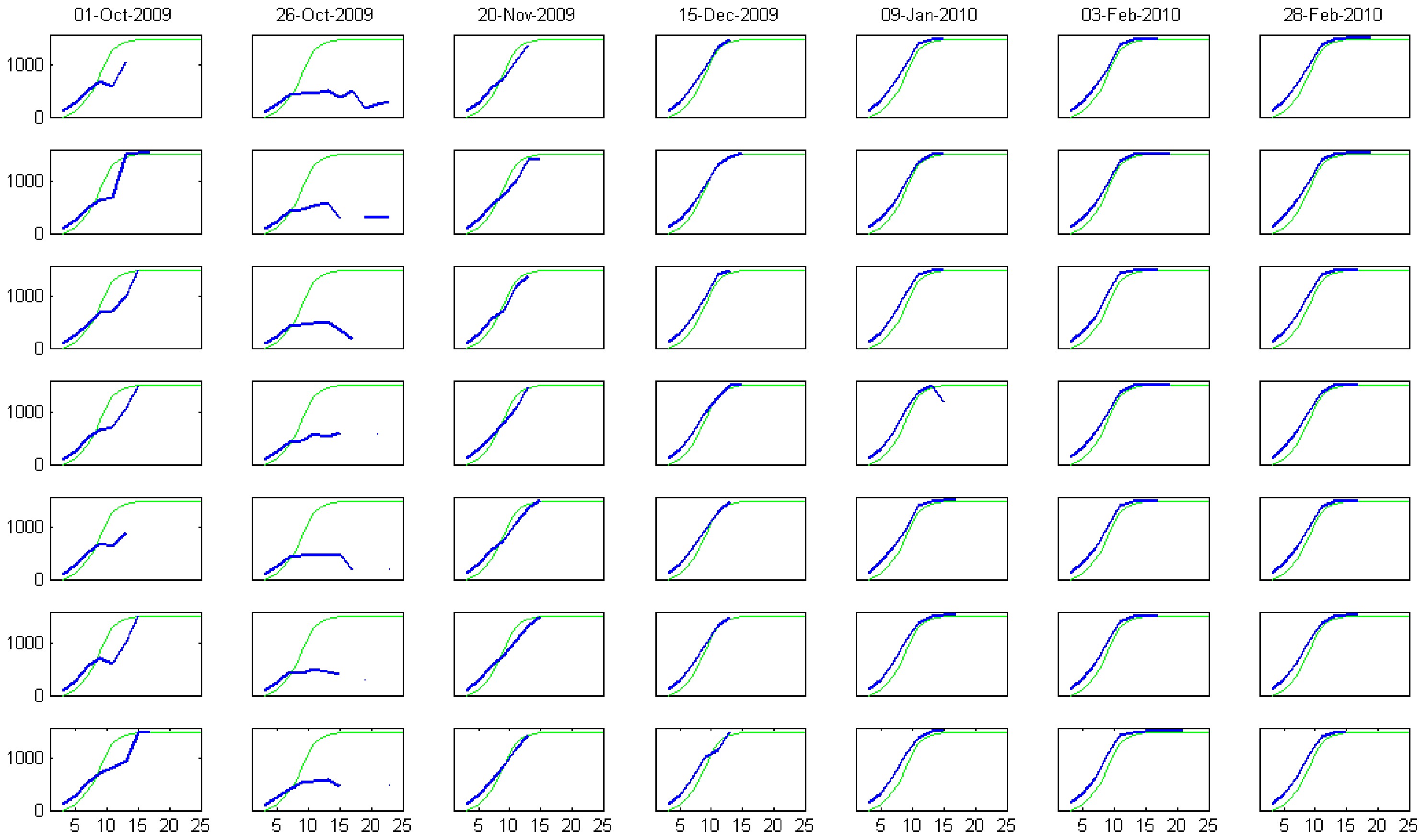


Turbines in the 502 group



- 49 turbines are associated with MET-502
- No windspeed data from T149-T154
- Select seven turbines to establish a baseline: T75, T81, T98, T104, T115, T118, T127.
- The selected turbines are geographically well distributed and have more consistent data.

Power average versus nominal power curve

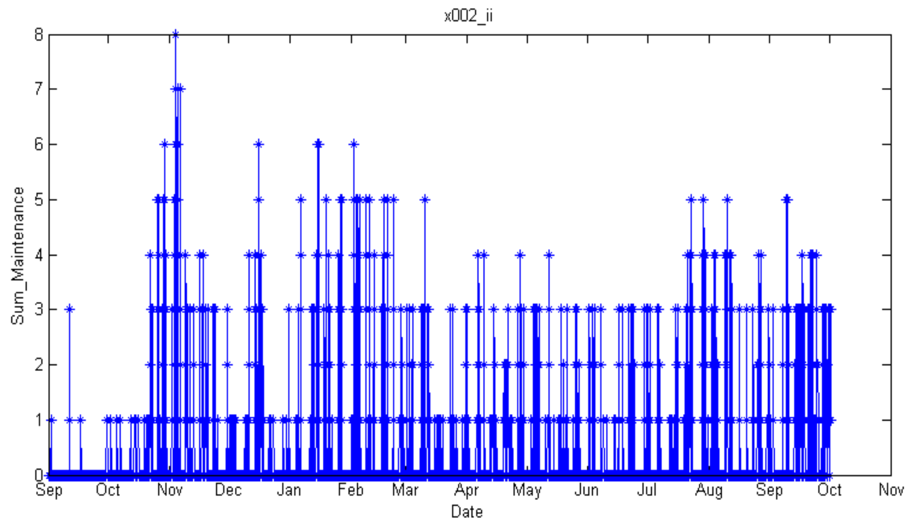


Anomalous WT

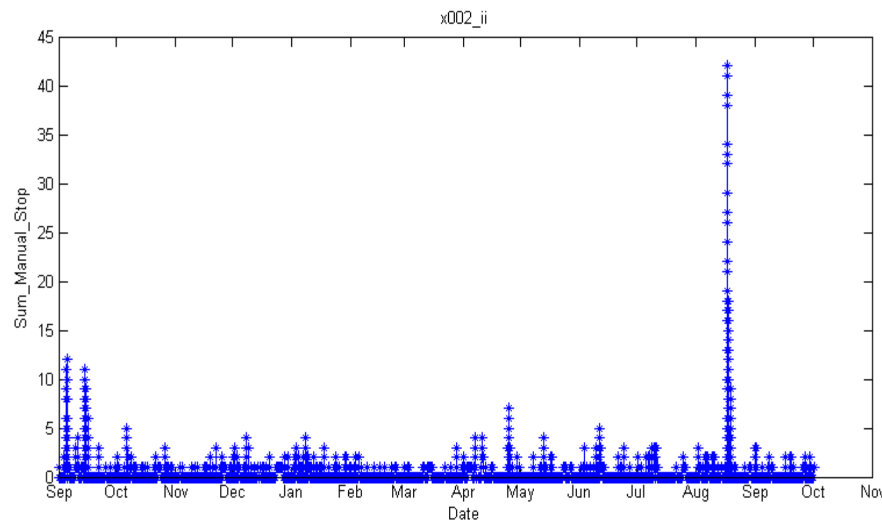
- Data from a Wt_i may look anomalous in a group of WTs for a number of reasons. In some cases, the cause of anomaly can be detected using simple statistics, in other cases using associative models to capture dynamic dependencies is needed.

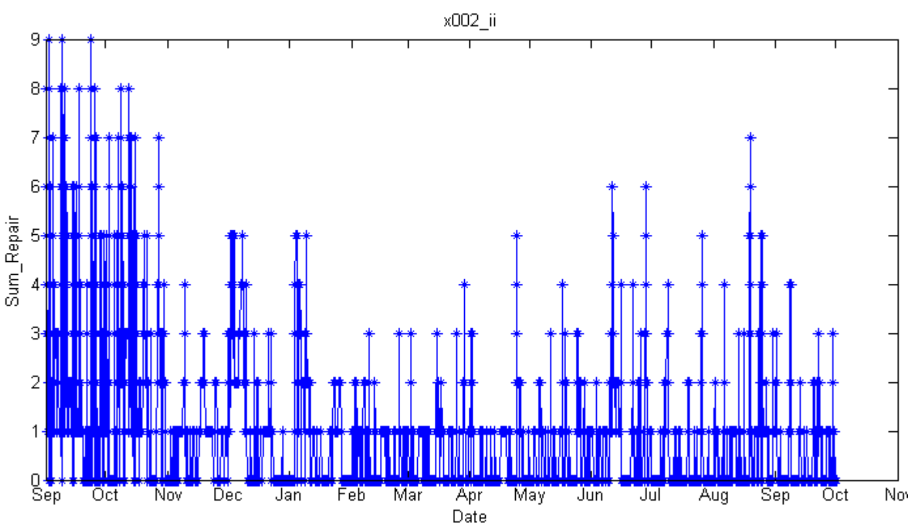
Case No	Wt_i	Remaining WTs	Action
1	Down	Normal	Use simple stats
2	Curtailed	Normal	Use simple stats
3	Normal	Down	Use simple stats
4	Normal	Curtailed	Use simple stats
5	Location effect	Location effect	Capture in associative model
6	Park-wide control effect	Park-wide control effect	Capture in associative model
7	Performance degradation due to fault	Normal	Detect using associative models

Parameters to Filter Data for Baseline



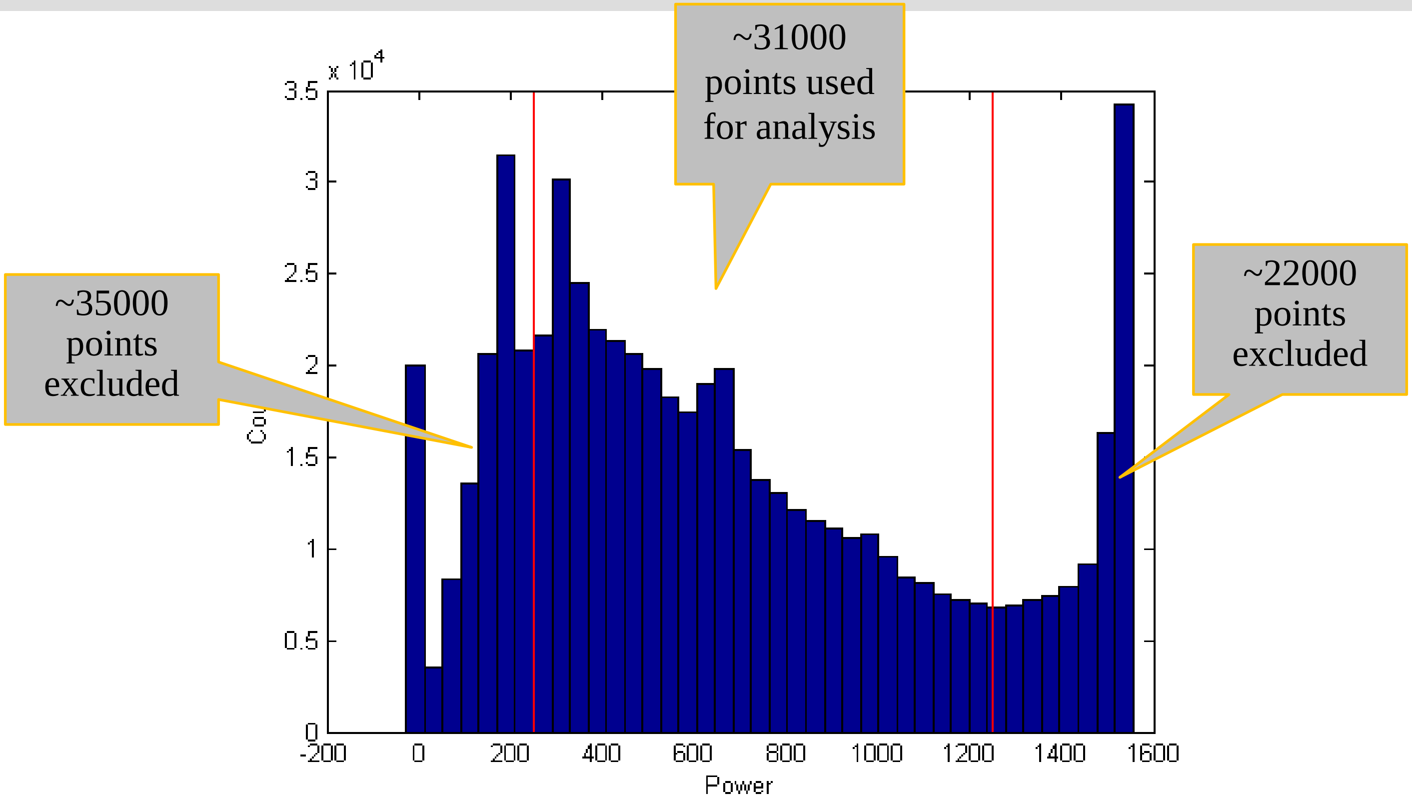
- Tag: “Sum_Maintenance” at Park Level
- Tag: “Sum_Repair” at Park Level
- Tag: “Sum_Manual_Stop” at Park Level





At any given time, up to 10 turbines (except the outlier in Sep10) are stopped manually and maintenance or repair performed.

Data filtering – Power profile



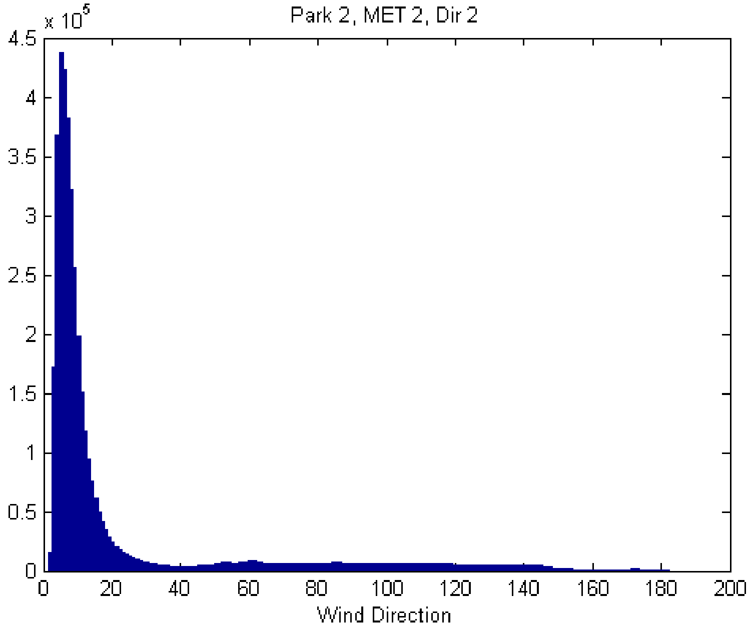
Exclude points that lay outside of 250 kW and 1250 kW. In these startup and max high power regions, the operation of WT is highly non-linear.

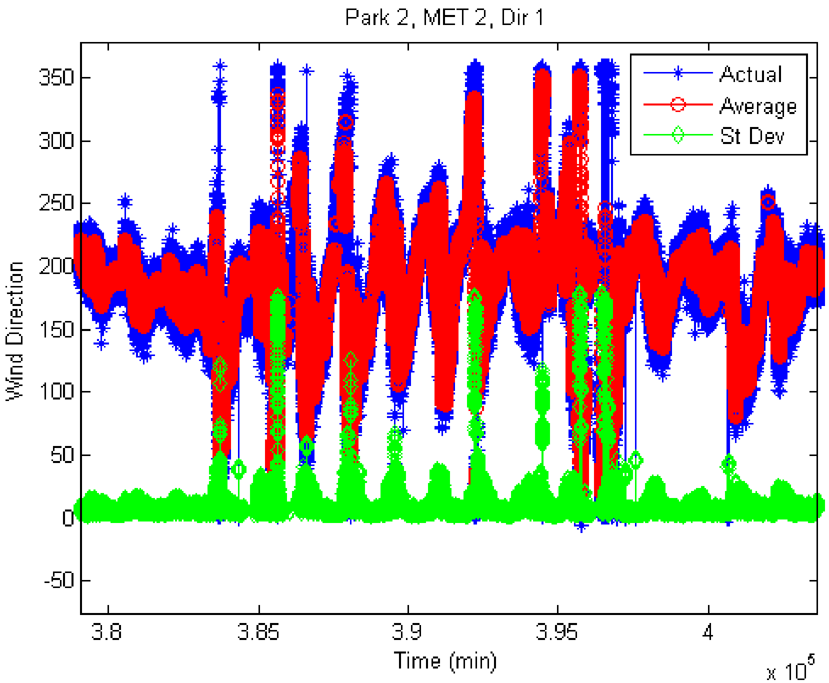
Honeywell

Honeywell.com

MET 2 – Wind Direction 1 & 2

- Seek periods of stable wind direction
- Compute wind direction ave and std
  - Scan 3 min around each sample
  - Require at least 20 samples (usually there is about 30 points)



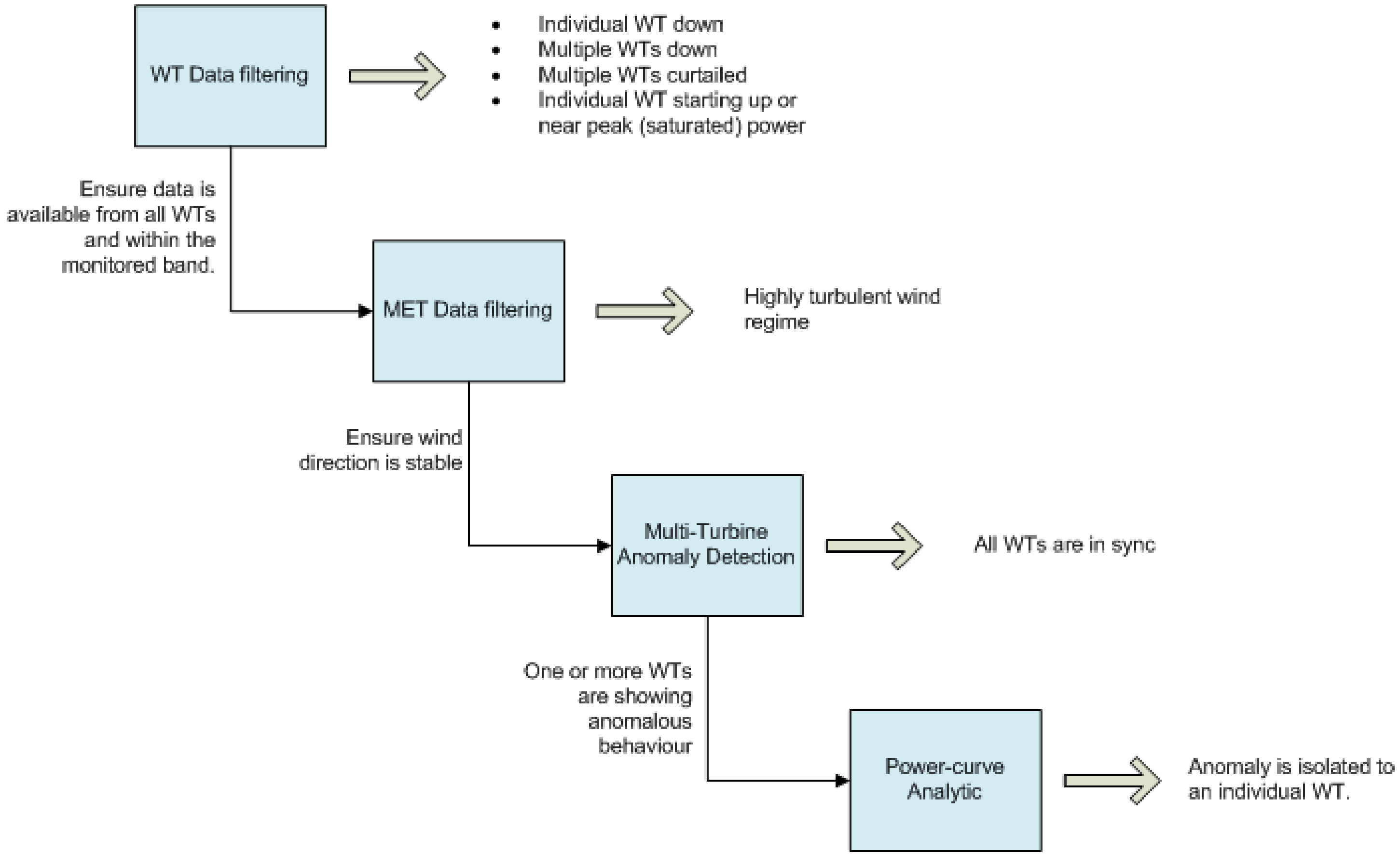


- Variation in wind direction is generally limited to less than std=20 deg within 3 min.
- Cut-off at std=10, filters 55% of data in Direction 1 and 65% of data in Direction 2.

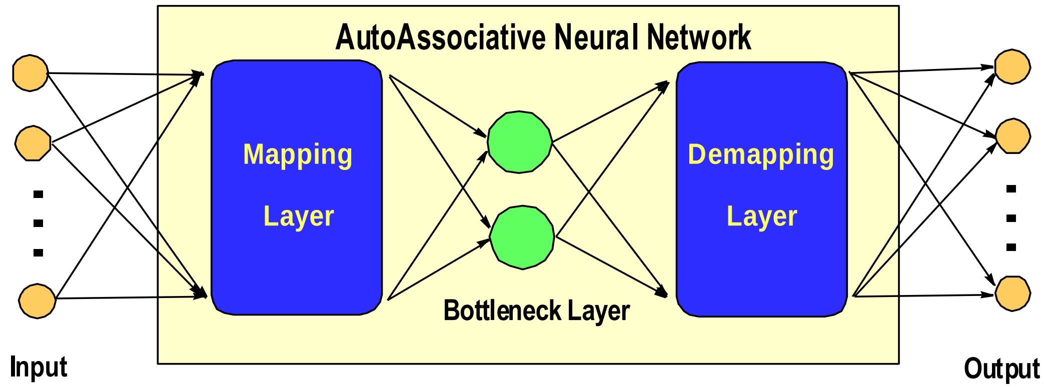
Honeywell

Honeywell.com

Hierarchical Monitoring of Wind Turbines



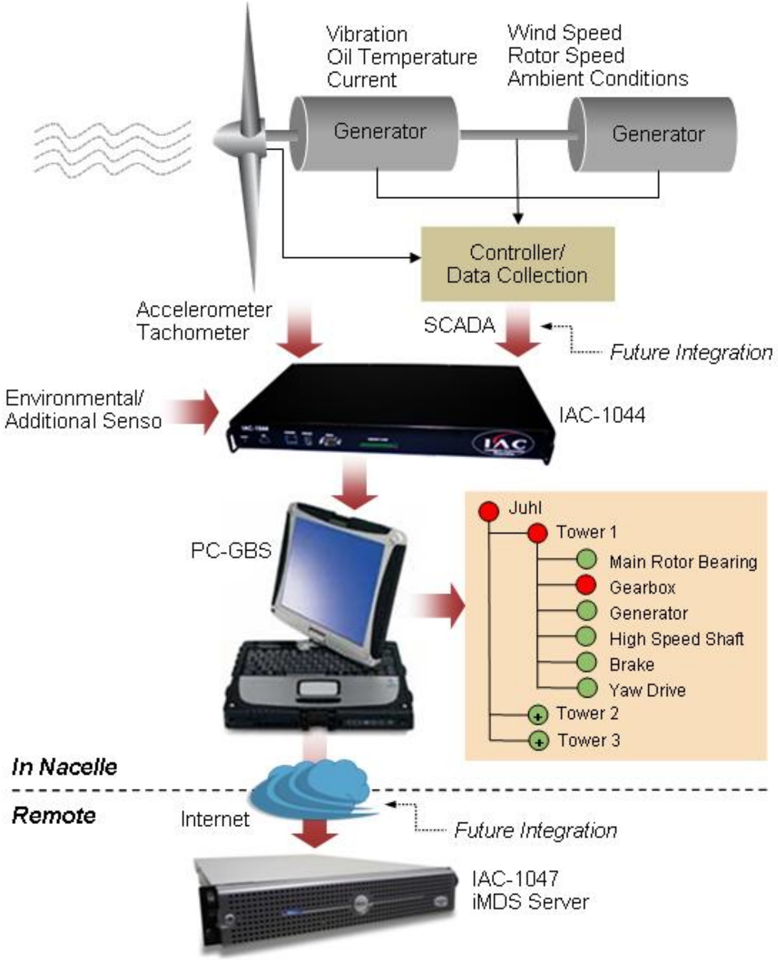
Associative Model Approach



- Approach
  - Employ multivariate analysis for analytical redundancy to capture non-linear correlations among wind turbines in a park
  - a multi-layered neural network architecture
    - Include a small bottleneck layer to ensure good generalization and prevent the network from forming a look-up table.
- Goal
  - If no fault present, reproduce the input data at the output as closely as possible
  - If there is fault, isolate the faulty wind turbine and estimate the power loss

Generic Wind Turbine CBM System

- Algorithms alone will not determine the success of CBM
  - System design and usability are key
  - Significant factor in the success of Honeywell's HUMS deployment
- Honeywell's HUMS software defines configurations that are setup once and duplicated
  - Setup tool capability allows diverse aircraft to be configured without source code changes
  - Flexibility enables rapid configuration and tuning of HUMS algorithms
- Beginning the path toward wind turbine CBM configurability
  - Gather equipment specifications, SCADA data configuration across multiple wind turbine models
  - Use this information to define an information model for wind applications using a structure similar to Honeywell's HUMS data model



Conclusions

- Objectives:
  - Safety
  - Economics:
    - Aftermarket services
    - Availability
    - Improve customer experience
- It has to be designed in-collaborative
  - Member systems need work together
  - Minimize resource utilization and maximize availability
  - Have small computation and communication footprint
  - Distributable
- System level conclusions shall support the objectives

System-wide health monitoring is "designed in" not "added on"

(Ref: www.public.navy.mil/navsafecen #426Teamwork)

Acknowledgements

- This material is based upon work supported by NASA Award Number NNL09AD44T
- NASA
  - Eric Cooper (COTR)
  - Robert Mah
  - Ashok Srivastava
- Vanderbilt University
  - Gautam Biswas
  - Xenofon D. Koutsoukos
  - Daniel Mack
- Honeywell
  - Dinkar Mylaraswamy, George Hadden
  - Craig Schimmel, Lee Graba, Dennis Cornhill, Wayne Schultz
- We are grateful for cooperation from
  - Great River Energy, Minnesota
  - Broadwind Energy Services, S. Dakota
  - University of Minnesota, Morris
  - Honeywell Matrikon, Richmond, BC
- Honeywell
  - **Golden Valley**: Girija Parthasarathy, Wendy Foslien, Kyusung Kim
  - **Poway**: Mark DiCiero, Julia Vega, Sonya Petty
- NREL
  - Shawn Sheng, Paul Fleming
- Juhl Wind
  - Corey Babcock, Dan Juhl, Tyler Juhl

We thank the US Department of Energy for the support for this work under Award Number DE-EE0001368.

Questions

Acknowledgment: "This material is based upon work supported by NASA Award Number NNL09AD44T and the Department of Energy under Award Number DE-EE0001368."

Honeywell

References

- Read the standards for your industry
  - Aerospace
    - Aeronautical Radio, Inc. (ARINC) www.arinc.com/amc
    - ARINC Report 604-1: Guidance for Design and Use of Built-In Test Equipment
    - ARINC Specification 624-1: Design Guidance for Onboard Maintenance System
  - Computer Networking / Communications
    - Open Systems Architecture for Condition-Based Maintenance (OSA-CBM)
    - www.mimosa.org/?q=resources/specs/osa-cbm-v330
  - Automotive
    - www.epa.gov/otaq/regs/im/obd/
    - www.obdii.com
    - OBD-I “On-Board Diagnostics, version 1” 1985
    - OBD-II “On-Board Diagnostics, version 2”1994
  - Alternative Energy (wind, solar, hydro, etc.)
  - Building Automation
  - Factory Production Lines
  - Medical Devices
  - Rail Transportation

- “System Health Management: with Aerospace Applications” published by John Wiley & Sons
- D. Mylaraswamy, “Addressing Aviation Safety using Vehicle Level Reasoning,” 1st Indo-US Workshop on IVHM and Aviation Safety (WIAS), NAL Bangalore, 2012.
- Srivastava, D. Mylaraswamy, R.W. Mah, E.G. Cooper, “Vehicle-level Reasoning Systems” in *IVHM Perspectives on an Emerging Field*, Ed. I.K. Jennions. Publisher: SAE International. 2011.
- Ron J Patton et al., Issues of Fault Diagnosis for Dynamic Systems, Chapter 9

”

References …

- G. D. Hadden, D. Mylaraswamy, Craig Schimmel, Gautam Biswas, Xenofon Koutsoukos, and Daniel Mack, “Vehicle Integrated Prognostic Reasoner (VIPR) 2010 Annual Final Report,” NASA/CR–2011-217147
- D. Mylaraswamy, “Vehicle Level Reasoning and Data Mining,” CIDU, 2010.
- D. Mylaraswamy, D. Hamilton, and G. Hadden, “Information Protocols: NASA VIPR Program,” Technical Report, submitted by Honeywell to NASA (CDRL Sequence Number: 4.1.03), December 2009.
- D L.C. Mack, G Biswas, X D. Koutsoukos and D Mylaraswamy, “Using Tree Augmented Naive Bayesian Classifiers to Improve Engine Fault Models," presented at the 8th Bayesian Modeling Applications Workshop in Barcelona, Spain on July 14th, 2011.
- D L.C. Mack, G Biswas, X D. Koutsoukos, D Mylaraswamy, and G Hadden, “Deriving Bayesian Classifiers from Flight Data to Enhance Aircraft Diagnosis Models,” to be presented at the Annual Conference of the Prognostics and Health Management Society, 2011.
- D. Mylaraswamy, “ Vehicle Level Reasoning and Data Mining,” CIDU, 2010.
- T. Felke, G. Hadden, D. Miller, and D. Mylaraswamy, “Architectures For Integrated Vehicle Health Management,” AIAA-2010-3433, 2010.
- G. D. Hadden, D. Mylaraswamy, C. Schimmel, G. Biswas, X. Koutsoukos, and D. Mack, “Vehicle Integrated Prognostic Reasoner (VIPR) 2010 Annual Final Report,” NASA/CR–2011-217147
- G. Biswas, X. Koutsoukos, D. Mylaraswamy, and G.D. Hadden, “Benchmarking the Vehicle Integrated Prognostic Reasoner,” Annual Conference of the Prognostics and Health Management Society 2010.
- D. Mylaraswamy, et al., “Vehicle Integrated Prognostics Reasoning,” NASA Aviation Safety Annual Meeting, St Louis, 2011.
- D.L.C. Mack, G. Biswas, X. D. Koutsoukos and D. Mylaraswamy, “Using Tree Augmented Naive Bayesian Classifiers to Improve Engine Fault Models,” presented at the 8th Bayesian Modeling Applications Workshop in Barcelona, Spain on July 14th, 2011.
- Srivastava, D. Mylaraswamy, R.W. Mah, E.G. Cooper, “Vehicle-level Reasoning Systems” in *IVHM Perspectives on an Emerging Field*, Ed. I.K. Jennions. Publisher: SAE International. 2011.
- D.L.C. Mack, G. Biswas, X. D. Koutsoukos and D. Mylaraswamy, and G. D. Hadden, “Deriving Bayesian Classifiers from Flight Data to Enhance Aircraft Diagnosis Models,” presented at the Annual Conference of the Prognostics and Health Management Society, 2011.
- R. Bharadwaj, et al., “Vehicle Level Prognostic Reasoning System,” talk at SAE 2012 Aerospace Electronics and Avionics System Conference, Oct 30--Nov 1, 2012. Phoenix, AZ
- R. M. Bharadwaj, K. Kim, C.S. Kulkarni, G. Biswas, “Model-Based Avionics Systems FaultSimulation and Detection,” American Institute of Aeronautics and Astronautics, AIAA Infotech Aerospace 2010, April 2010, Atlanta, GA. AIAA-2010-3328.
- D. Mylaraswamy, et al., “Vehicle Integrated Prognostics Reasoning,” NASA Aviation Safety Annual Meeting, St Louis, 2011.