

Embedded Reasoning Supporting Aerospace IVHM

Michael H. Schoeller¹ and Michael J. Roemer²
Impact Technologies, LLC, Rochester, NY 14623

and

Matthew S. Leonard³ and Mark Derriso⁴
AFRL/VASA, Wright-Patterson AFB, OH 45433

The work presented in this paper summarizes an effort to advance the current state of vehicle health reasoning practice as a fundamental element of an Integrated Vehicle Health Management (IVHM) approach. This effort seeks to integrate detection, diagnostic, and prognostic capabilities with a hierarchical diagnostic reasoning architecture into a single synergistic, embeddable commercial off-the-shelf (COTS) unit. Three avenues have been pursued to advance the capabilities of health reasoning in support of IVHM. First, the scope of the systems/subsystems being monitored by the underlying Health and Usage Monitoring System (HUMS) has been expanded. The HUMS is comprised of subsystem specific prognostic and health management (PHM) modules developed to process raw measured data and supply condition indicators (CI) targeting several flight critical areas from advanced aerospace applications: such as electro-mechanical actuators (EMA), engine performance, power train vibration, and structural impact detection and isolation. Second, a hierarchical, model-based reasoning approach has been developed to isolate and classify latent failure mode manifestations existing in the condition indicators output by the diagnostic and prognostic algorithms and utilize this information to assess the functional availability of the vehicle and its constituent subsystems. At the lowest level, the embedded reasoning seeks to classify latent failure mode indications from raw sensor data or condition indicators considered as evidence sources and isolate the root cause mechanism along with the accompanying severity. In addition, a confidence assessment is determined by considering the temporal element and how many times a particular piece of indicting evidence has repeated in the previous N evaluation iterations. The mid-level of the reasoning architecture is employed to determine the overall functional availability of the constituent subsystems, i.e. what are the implications of the detected failure modes on the capability of the subsystem? At the highest level of reasoning provided, the functional availability assessments, or condition indicators, from all underlying subsystems are utilized in conjunction with any applicable remaining useful life assessments to facilitate informed decisions about whether or not the requirements of scheduled flight operations can be fulfilled. Additionally, the temporal element is again exploited to determine the existence of a failure propagation path and isolate the most likely root cause failure mode. The final avenue is a demonstration on a COTS platform, built upon the PC-104 data bus architecture that provides a solid data acquisition and processing foundation. The work presented here highlights an end-to-end approach for addressing mission-critical subsystems across an advanced aerospace platform, from data acquisition to communication of intelligently reasoned root-cause failure mode and functional availability assessments made at the subsystem, system, and vehicle levels. Integration of this technology is believed to greatly support and enhance the efficacy of an IVHM system in managing the operation and maintenance of the host platform, facilitating proactive management, reducing operating costs, and improving operational availability.

¹ Project Manager, 200 Canal View Blvd, and AIAA Member Grade for first author.

² Director of Engineering, 200 Canal View Blvd, and AIAA Member Grade for second author.

³ Insert Job Title, Department Name, Address/Mail Stop, and AIAA Member Grade for third author.

⁴ Insert Job Title, Department Name, Address/Mail Stop, and AIAA Member Grade for third author.

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 2007		2. REPORT TYPE		3. DATES COVERED 00-00-2007 to 00-00-2007	
4. TITLE AND SUBTITLE Embedded Reasoning Supporting Aerospace IVHM				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Impact Technologies LLC,200 Canal View Blvd,Rochester,NY,14618				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 19	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

I. Introduction

Integrated Vehicle Health Management (IVHM) is the next evolutionary step in condition based asset management, endeavoring to build on the safety and readiness benefits obtained from legacy Health and Usage Monitoring Systems (HUMS) while also enhancing the scope of the systems covered beyond the traditional drive train monitoring. Over the years the evolution of data acquisition, processing, and storage capabilities has enabled continuous advancements in the practice of Prognostics and Health Management (PHM) and condition-based asset management approaches. For air vehicles, this has resulted in increased autonomous operation in flight as well as on the ground; reduced ground maintenance and repairs, facilitated by regular system health checks in flight; and improved safety and reliability via the continuously improving efficacy of the systems monitoring health¹. IVHM is an integration and coordination of both the embedded aspects of health and usage monitoring and advanced reasoning onboard with the ground based supporting infrastructure all focused on mitigating the impact of detected component failures on mission readiness.

As the name implies, the functional foundation of IVHM in its true sense is highly integrated into the vehicle, its constituent subsystems, and the supporting ground based infrastructure. In broad terms an IVHM system can be thought of as an enabling, decision support tool which provides autonomous, timely, and accurate assessments of a vehicle's health and functional availability to operations personnel. The greater vision of the system seeks to include utilizing this information on the ground to facilitate decreasing operational costs and increasing operational readiness in order to provide significant life cycle benefits to aerospace programs. As such, a pragmatic view of an IVHM approach would be realized through two distinct areas of functionality: The first, an embedded aspect, facilitating the goal of on-board vehicle health monitoring as summarized by Ofsthun²; "to have the vehicle dynamically identify any degradation in functional performance that may affect safety or successful field operations as well as to identify the specific subsystems that require maintenance to restore full operational capability." The knowledge gained from the embedded IVHM functionality must then be forwarded along to a ground-based aspect of the greater system to be used by operations personnel. The ground based system provides the interface between the onboard monitoring and reasoning functionality and operations, maintenance, and logistics personnel and infrastructure, facilitating informed, condition based asset management and support. The scope of this paper shall remain focused on the embedded implementation of the IVHM architecture in aerospace applications.

The development of an IVHM system to monitor advanced aerospace platforms is predicated upon expert knowledge of the vehicle and the subsystems called upon for its operation. Engineers tasked with development of an IVHM system require a detailed understanding of the critical subsystems, their dominant failure mechanisms, and the characteristics of failure interaction and propagation. Pareto analysis of these subsystems, in conjunction with relevant Failure Modes and Effects Analysis (FMECA) provide a solid starting point. Overall system safety and operationally critical subsystems are given a high priority when considering which subsystems to target. FMECAs of these subsystems provide insight into the physical effects of dominant failure modes and into how to isolate these manifestations as evidence to indict the particular fault. The developer weighs this information along with the 'real-world' constraints of available data acquisition and processing capabilities to formulate the cost/benefit design choices and model the embedded aspect of the IVHM system. Developing a model of the IVHM system prior to installation provides two distinct benefits: First, the model provides the functional blueprint to develop the Health and Usage Monitoring System (HUMS). The model is then exercised to ensure a satisfactory level of health monitoring coverage is provided by the system prior to its development in software and implementation in hardware. Second, the model provides the functional architecture which is exploited by the embedded reasoner to isolate root cause failure modes and remaining functionality available in the presence of a detected failure. Figure 1 provides an overview of the process described. The feedback loop outlines an updating process wherein certain parameters within the PHM model can be modified based on user feedback to improve the results obtained from the reasoner.

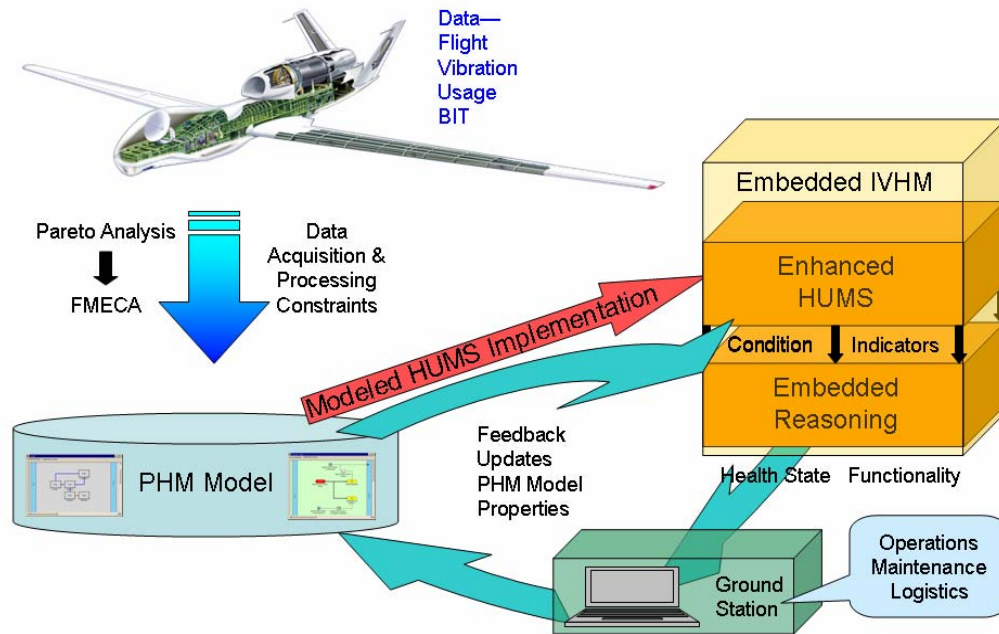


Figure 1. IVHM Development Process

II. PHM MODEL

This paper begins by addressing model development of the PHM functionality required to provide the desired level of health monitoring coverage to the targeted vehicle. The model of the underlying PHM system provides the foundation for both functional areas of the Embedded IVHM system presented in Figure 1, the Enhanced HUMS and Embedded Reasoning: First, guiding development and optimization of the enhanced HUMS for the vehicle, including LRU coverage, sensor selection, and algorithm development. Second, the information characterizing the efficacy of various evidence sources in indicting their respective failure modes and functional hierarchy information required by the subsequent reasoning processes also resides in the PHM system's model.

Significant working knowledge of the vehicle and its subsystems is necessary to effectively design the PHM system for the platform. The suggested approach would employ a Pareto analysis of the dominant subsystem failures experienced by a vehicle, followed by a Failure Modes Effects and Criticality Analysis, FMECA, of the subsystems dominating the Pareto. At this point safety and mission criticality judgments are combined with reachability (the capability to monitor, measure, and assess) considerations to identify which failure mechanisms the final PHM functionality will target. Developers can then tailor subsystem specific approaches when developing the PHM modules employed in the HUMS, gaining maximum benefit from use of a minimum set of sensors. Modeling the attributes of a PHM approach, including sensors, evidence sources, failure modes, and functional dependencies, allows developers to ensure adequate coverage of the underlying subsystem's failure modes and gain maximum efficacy of the onboard aspect of the IVHM system.

A forthcoming section outlines the process employed by the reasoning engine in ranking failure modes based on the information contained in the PHM system model. The model-based aspect of the reasoning process comes from the inherent understanding IVHM developers must have of the relationships between the targeted failure modes of interest and the evidence sources which are called upon to indict them. In addition to this information, knowledge is required of the functional dependencies which exist between line replaceable components, LRCs, and subsystems, as well as between subsystems and the vehicle. To this end, a model is required of the PHM system developed to monitor the health state of the underlying system. The functional awareness is less important for the lower level reasoning, but is critical to reasoning at the higher levels.

The PHM Design™ modeling environment has been utilized for model development. This design tool offers developers two parallel environments to graphically model the functionality of a health management system: the *Functional Model* and the *Health Management Design*. The Functional Model palette provides an environment where the functional relationships between components, assemblies, and subsystems comprising a system can be

organized and defined. The Health Management Design palette offers a medium to visualize and store failure modes, effects, health management techniques, sensors, and the relationships between these elements.

Developers begin with the top functional level of the vehicle. This level consists of the main functional areas comprising the vehicle for which PHM coverage is desired. The Functional Model palette is populated with blocks or Areas which represent systems, subsystems, assemblies or components. A cascading development process is employed to break down the high-level systems until sufficient fidelity is attained such that all remaining functionality is representative of either a component or a LRC. The blocks are linked by color-coded lines representing functional interaction or dependencies; green lines represent fluid/flow connections, red lines represent mechanical connections and blue lines signify the electrical connections. Figure 2 illustrates the coarse, high-level

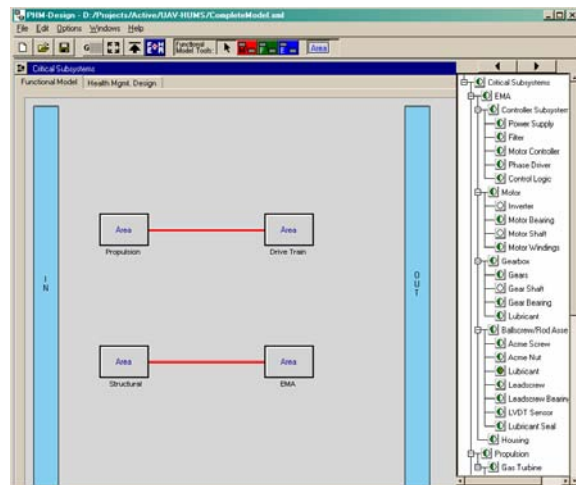


Figure 2. Overall System PHM Functional Model

system model containing the functionality represented in this design program. Notice that a block (Area) has been defined in the model corresponding to each of the targeted subsystems, propulsion, flight control actuators, drive train and structural monitoring. Each Area represents hierarchical subsystems, assemblies, and components of a physical system in the functional model. A red line between the Propulsion block and the Drive Train represents a mechanical link between the engine and the drive train.

A closer inspection of the 'EMA' Area, Figure 3, reveals five constituent subsystems making up the Functional Model of the electro-mechanical actuator subsystem. Similar high-level subsystem diagrams are available for each of the subsystems shown in Figure 2. The cascading development process breaks down the functionality employed in the subsystems until sufficient fidelity is reached. At that point, any of the following conditions may exist: the remaining components can no longer be divided, only line-replaceable-

units exist, or it is no longer practical to monitor the operation of the subsystem at any lower levels. The tree structure in the pane on the right of Figure 2 reinforces the hierarchical structure of the Areas defined in the model and also facilitates easy navigation among the different functional levels.

In conjunction with the Functional Model development environment, PHM Design™ also offers developers the Health Management Design palette on which to develop a graphical representation of the desired PHM system. Information typically assembled for a FMECA is combined with information regarding the PHM system developer's capability to detect emerging faults, i.e. sensors and detection and diagnostic algorithms. Fault/Failure Modes (red ovals) for the respective subsystems and components are mapped along with Symptoms/Effects (yellow ovals), available sensors (eyeball), health assessment algorithms ('graph' icon), BIT/discrete health indications (traffic lights), and maintenance actions (multi-colored rectangles) to facilitate development of a comprehensive health management system. Failure modes are introduced for all of the targeted subsystems and replaceable units. Figure 4 provides an example implementation of some information contained in an actuator FMECA. Failure modes (red ovals) generally manifest themselves in some manner of Symptom/Effect (yellow oval) shown in the model as a directional connectivity. The symptom in turn is monitored by a diagnostic method (BIT or health assessment algorithm) which represented by the connection between these elements. Finally, the monitoring diagnostic relies on input information and as such is connected to a sensor (eyeball).

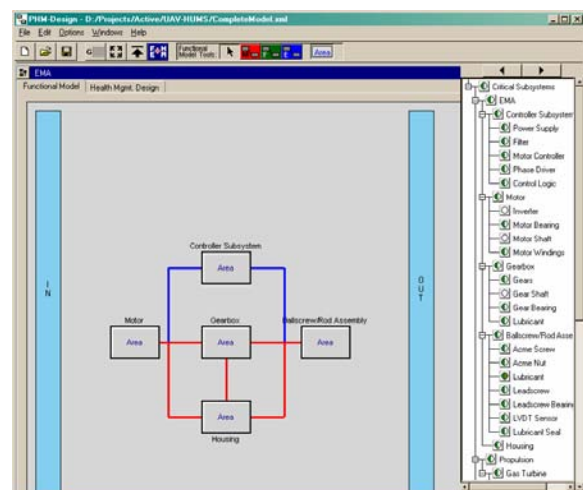


Figure 3. Actuator PHM Functional Model

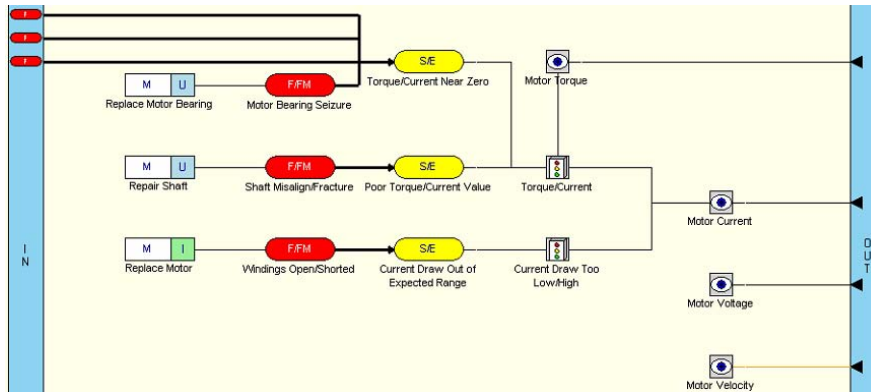


Figure 4. Health Management Design Example of an EMA Motor

In the diagram, the vertical “IN” and “OUT” mux bars represent connections coming from and going to other parts of the system. For example, one of the failure modes coming into the EMA motor is from the acme configuration element of the EMA. If the nut seizes to the screw it would produce the same symptom/effect as a motor bearing seizure.

PHM Design™ offers an efficient means of graphically representing the targeted systems and their respective PHM

capabilities. The process of first building up the functional relationship and then augmenting this with the PHM capabilities available at each successive level of model fidelity provides a hierarchical approach on which to develop the reasoning logic. This graphical representation serves as an intuitive means of formulating and visualizing the framework which must be employed in the Embedded Reasoner System to determine root-cause failure modes at the vehicle level.

III. Enhanced Health and Usage Monitoring System

The overall function of IVHM is that of a decision support tool providing invaluable real-time health state and functional availability information in order to facilitate optimal asset management. To achieve this goal the IVHM system requires its own source of information regarding the health state of safety and mission critical systems on which to base its assessments; this responsibility is incumbent upon the enhanced Health and Usage Monitoring System (HUMS). The HUMS is comprised of multiple subsystem specific PHM modules responsible for processing measured data to obtain *Condition Indicators* (CI) which capture and reflect the current health state of the underlying line replaceable units (LRU) comprising the various functional levels. The HUMS achieves this with the support of an embeddable COTS data acquisition and processing platform capable of supporting high data throughput rates necessary to support advanced signal processing techniques employed by the subsystem specific diagnostic modules. The enhanced HUMS discussed here provides the foundation of health information, captured in the diagnostic features or condition indicators, necessary to support IVHM for the vehicle. Acting on the stream of data coming real-time from sensors and other data sources employed, the HUMS brings the health monitoring coverage, designed via the process laid out above, to fruition on the vehicle.

The enhanced HUMS is comprised of subsystem specific modules. The system discussed here incorporates modules targeting structural impacts, drive train, engine performance and vibration, and electromechanical actuators used to control various flight and operational duties. The modules process, either raw analog data acquired from sensors, or digital data obtained from other sources. Output results delivered from the on-board processing provide CI to the subsequent reasoning process. These condition indicators correlate directly to evidence sources upon which to classify targeted operational anomalies within the vehicle’s components and subsystems. The approaches employed in the diagnostic modules can be categorized into three general types based on the type of algorithm supplying them: Built-In Test (BIT), Diagnostic, and Prognostic evidence.

A BIT can be thought of as an intelligent indicator light, indicating the presence or absence of an anomaly in a given system or component. Although a BIT is intelligently designed, it can be influenced by system noise leading to false alarms and intermittent indications. A Diagnostic type algorithm is a more informative type of evidence source, often referred to as gray-scale evidence, due to its ability to provide varying levels of anomaly indication. A Diagnostic can provide confidence that a fault exists, severity of the fault, or a condition index (health value) of a component or system. Still, Diagnostics often do not provide a clear correlation with the root cause failure mode that has occurred, especially if the failure has occurred at an upstream system or component. In those cases, higher level reasoning is required. At the more advanced level, Prognostics have the ability to estimate when a fault will become a functional failure or reach a critical severity level. Prognostics offer the advantage of providing advance warning, by predicting remaining useful life (RUL) or time to failure (TTF), before the situation becomes critical, so that serious consequences can be averted or maintenance can be scheduled appropriately.

The following sections outline the approaches adopted by the various modules to provide timely condition indicators categorizing the health state of key mission critical subsystems.

A. Structural Impact Monitoring

One of the areas of interest in implementing comprehensive vehicle health monitoring functionality includes the detection and localization of impact events on key structural and flight control surfaces. These impacts may cause damage that ultimately affects the vehicle's ability to complete the required mission. In the arena of unmanned flight, without a pilot on board to make critical decisions, autonomous assessment of damage becomes more important to operational success. A technique has been developed and implemented that can assess the location and damage level resulting from impacts on structural components that occur during flight³.

The embeddable structural health monitoring software module accepts raw vibration signals from strategically placed accelerometers on a structural panel. The continuous monitoring procedure looks for sudden increases in vibration levels. If an impact is suspected to have occurred, further processing takes place to determine the location and estimate the severity of the impact. The analysis is based on using the relative time of arrival of the wave energy propagating away from the point of impact to the sensor locations. Applying a combination of advanced signal processing techniques and lookup tables enhances the accuracy.

A potential extension of this technology is to perform fatigue analysis given the assessed structural damage level. This would provide a means to calculate the amount of time the craft can remain airborne before a damage induced crack grows to a critical, flight-ending size. The approach is based on crack growth formulations and fatigue calculations; implementation requires knowledge of static and cyclic loading profiles on the structural member under consideration.

B. Drive Train – Bearings

Rolling element bearings form an integral part of many machinery applications. Failure of a bearing can be critical, especially in an airborne vehicle. Faults in bearings within the drive train of a vehicle can be detected and diagnosed based on changes in the measured vibration near these critical components. The enhanced HUMS employs an embeddable software module that performs bearing diagnostics using data from one or more COTS vibration sensors.

The bearing module performs both conventional vibration analysis as well as specially developed procedures. In the time domain, statistical vibration features are calculated that can be trended to indicate the presence of a drive train fault. These include RMS, kurtosis, skewness, and other commonly used statistical measures. In addition to time domain feature extraction, the vibration signal is transformed to the frequency domain where further analysis takes place. In the frequency domain, the diagnostic module examines the energy at particular frequencies to extract features that can be used to isolate faults to a particular bearing in the drive train. The primary bearing frequencies of interest correspond to the bearing's characteristic fault frequencies, namely the Ball Spin Frequency (BSF), Ball Defect Frequency (BDF), Ball Pass Frequency Inner Race (BPFI), Ball Pass Frequency Outer Race (BPFO), and the Fundamental Train Frequency (FTF).⁴ In addition to this conventional analysis, the raw vibration signal is also processed using an ImpactEnergy™ routine. ImpactEnergy™ is a customizable, COTS sensor compatible, amplitude demodulation-based procedure for detecting high frequency excitations induced by periodic impulses from bearing defects. This signal processing method allows the spectrum magnitudes at key frequencies to be more easily extracted from noisy signals. The features computed from the ImpactEnergy™ signal serve as early warning indicators for impending failure, and often provide advance warning before conventional vibration features would respond.⁴ In general, shifts in the conventional and ImpactEnergy™ features when trended over time can be indications of degrading bearing health. These bearing diagnostic techniques have been successfully applied to a number of machinery test rig applications and in engine test cells⁵.

Prognostics based on statistical life distributions for specific bearing geometry can provide an estimate of the remaining time a bearing can be expected to perform its required function. The expected life is a function of load and speed, but within the software this prediction can be tempered by evidence of the existence of a fault and incorporating bearing fatigue calculations.

C. Drive Train – Gearboxes

Gears are commonly found in many mechanical systems and are a critical element of power transmission. A failure in the drive train gearing can result in a serious compromise in operation, potentially leading to catastrophic failure. The drive train module applies vibration based techniques to provide coverage of critical gear and shaft failure modes. Some of the potential faults that can be detected include gear tooth cracking and fracture, and tooth surface pitting.

Vibration analysis techniques for identifying gear faults are founded on basic engineering principles. Gear faults generate events in the vibration signature that are synchronous with the rotation speed of the gear. Gear fault detection is best performed utilizing Time Synchronous Averaging (TSA) to reduce the effects of random noise and emphasize vibrations caused by the gearset under consideration. Although the extraction of gear fault detection and tracking features from a vibration signal is well understood and documented in literature, this procedure has been enhanced and customized for this application. The process is performed on vibration data collected from a sensor located near a critical gearset, from which trendable features are computed. A technique has developed to perform the analysis without requiring a synchronized tachometer signal, which saves computing resources.⁶

Prognostics based on crack growth models for specific gear geometry can provide an estimate of the remaining time before a defect reaches a critical size. This approach depends on extensive finite element modeling performed in advance. These results are cross referenced based on the estimated crack size currently present in the gear.

D. Drive Train – Shafts

In addition to allowing the computation of gear diagnostic features, the TSA signal processing is leveraged to provide information about the condition of shafts in the drive train. Vibration features are extracted from the signal that indicate shaft faults such as cracking and unbalance. This portion of the analysis looks for increases in the magnitudes of the TSA vibration spectrum at shaft rotating frequency and higher orders.

E. Engine Performance

Advanced vehicles often have many monitored engine parameters utilized in the electronic control of the engine that may be made available for subsequent analysis by a health monitoring system. Engine performance monitoring is based on exploiting the availability of these monitored gas path parameters, including spool speeds, temperatures, pressures, and flow rates at key points in the engine. Degradations in the engine's performance will commonly manifest themselves as gradual shifts away from expected values corresponding to the current operating state of the engine, as illustrated in **Error! Reference source not found.** and Figure 6. The embedded engine performance monitoring module implements a model based approach based upon detecting and classifying these shifts as they occur. This technology assesses signal health and system performance by employing a combination of signal processing, statistics, and data-driven modeling techniques. The process is illustrated in Figure 7. Measured engine parameter data are input to the model, which provides expected values as output. These expected values are then evaluated relative to the original measured data to determine the residual value. These residual values used together

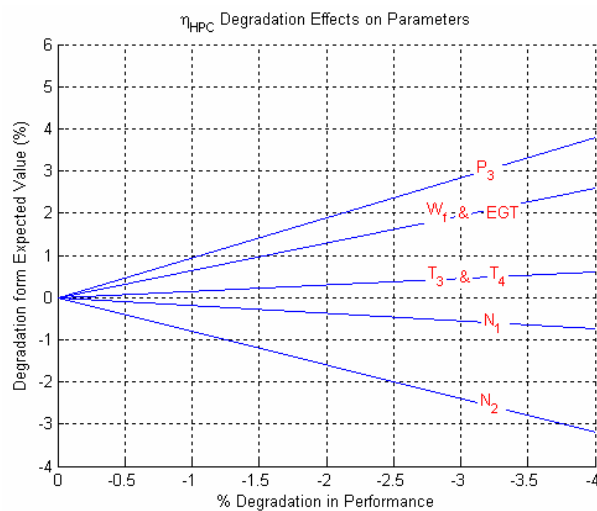


Figure 5. Effects of High Pressure Compressor Degradation on Engine Parameters⁷

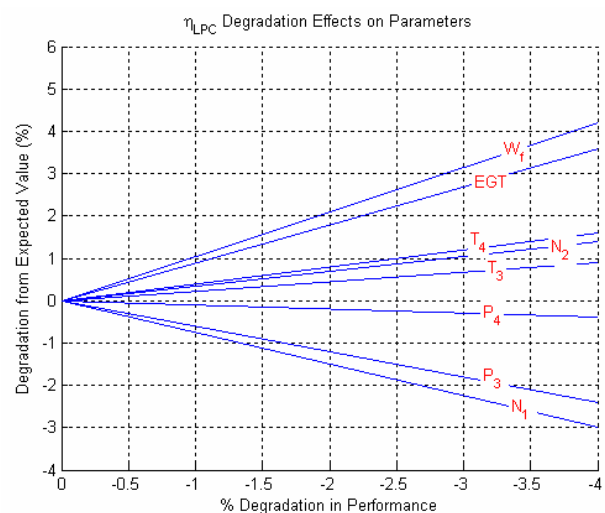


Figure 6. Effects of Low Pressure Compressor Degradation on Engine Parameters⁷

form a pattern which can then be compared to the characteristic degradations like those depicted in **Error! Reference source not found.** and Figure 6. Subsequent reasoning can then be applied to isolate the source of the performance degradation to a specific section of the engine. The techniques described here have been implemented successfully in gas turbine engine test cells⁸.

Consideration of mechanical vibrations in the engine's moving parts augments the engine performance evaluation to form a more complete propulsion monitoring system. Techniques described elsewhere in this paper can be applied to drive train components specific to the engine such as rolling element bearings. In addition, engine orders (1x, 2x, and higher harmonics of running speed) can be tracked to provide indications of a variety of faults

Prognostics based on fatigue accumulation models for specific components and engine configurations can provide an estimate of the remaining life before maintenance or overhaul is required. These models are a function of specific usage, mission profiles, speeds and loads, and can be adjusted by diagnostic indicators of the existence of faults. A much simpler approach can be implemented which tracks Total Accumulated Cycles (TAC) to provide a rudimentary approach to assessing engine life consumption. This approach, applied to gas turbines, monitors very generalized engine speed movements to gauge the severity of thermal transients and assess usage accumulation.

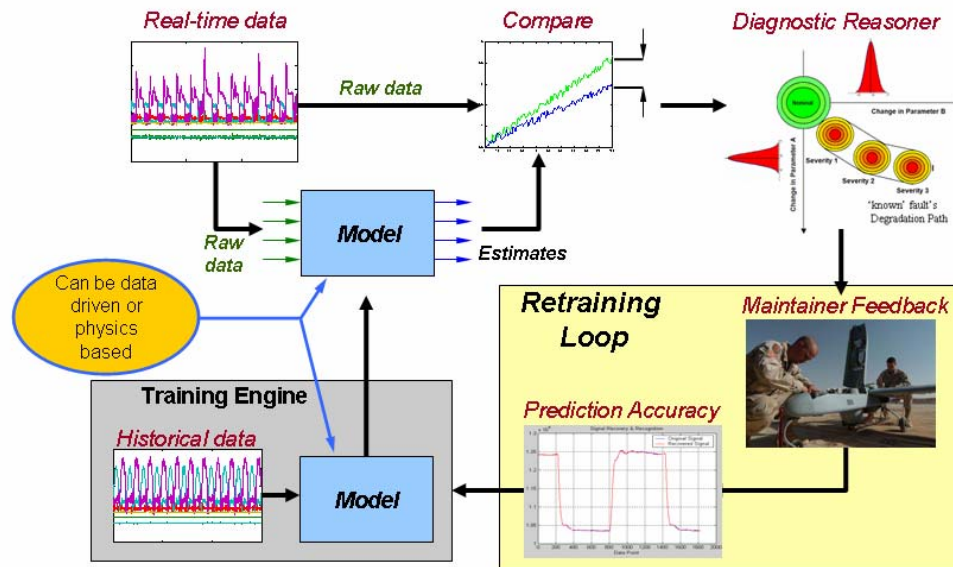


Figure 7. Model Based Vehicle-level Anomaly Detection, Diagnostics, and Trending

F. Actuators

Linear actuators are used extensively in aircraft for everything from manipulation of flight control surfaces to weapon release systems. One commonly applied type of actuator is the electro-mechanical actuator (EMA), which uses an electric motor and gearset to produce linear or rotary motion. EMAs have some advantages over equivalent hydraulic actuators; less weight, size, and complexity, characteristics that allow them to gain acceptance for aerospace applications.

The EMA module implemented in the HUMS employs two technically independent approaches to yield an overall assessment of EMA health. A model based approach utilizing only command and response information from the EMA's controller employs a core group of diagnostic features, including local gear stiffness, frictional damping, and torque constant to determine EMA system health. In this implementation the model based approach illustrated in Figure 7 utilizes command signals as input and exercises a model of the actuator system varying the values for the three diagnostic features until a suitable match is found, measured by a minimization of the residual between the modeled response and the measured response. The set of the three diagnostic features then serve as the axes of a three dimensional failure space wherein different failure modes propagate through the failure space on different, unique paths. A second technique utilizes vibration data collected from an accelerometer to monitor the characteristics of the rolling elements in the actuator. This approach utilizes traditional vibration based features to determine overall actuator health. Results from the two approaches are then evaluated by the low level reasoning to assess EMA health.

IV. VEHICLE AND SUB-SYSTEM LEVEL REASONING

One of the key aspects distinguishing an IVHM approach from a basic HUMS is an enhanced reasoning capability. Reasoning, within the IVHM system, processes Condition Indicators (CI) obtained from the subsystem specific HUM modules to determine the current *health state* of the vehicle and its constituent subsystems. This goes beyond the basic diagnoses of observed condition indicators by attempting to quantify the remaining functionality, which has been reduced by the indicted failure mode(s). The reasoning engine employs a hierarchical architecture to not only identify failure modes in line replaceable components (LRC), but also determine the functional impact in terms of remaining functional / operational availability at the subsystem and vehicle levels. When analyses result in an ambiguous outcome the reasoner attempts to isolate the root cause or at least reduce the ambiguity group. The reasoner also attempts to provide estimates of the severity of the underlying failure and the remaining useful life of the LRC where prognostic capabilities are available. The reasoned results output by the IVHM system provide real-time health and remaining functionality information useful to operations personnel for decision support.

At the lowest level, diagnostic reasoning seeks to classify latent failure mode indications from raw sensor data or diagnostic feature data processed by the subsystem specific modules within the HUMS. The mid-level of the reasoning architecture is employed to determine the overall functional availability of the constituent subsystems, i.e. what are the implications of the detected failure modes on the functional availability of the subsystem? The vehicle or system-level reasoning, the highest level of on-board reasoning, share this task of determining and quantifying functional availability, but from a vehicle or system-wide perspective. At this highest level, the functional availability assessments, or condition indicators, from all underlying subsystems are utilized to determine the capability of the vehicle to continue operations as expected or needed.

A. Low-level Reasoning

Typically, PHM approaches implemented in intelligent health and usage monitoring modules are designed to target from one to numerous failure modes within a specific LRC, assembly, or subsystem, often utilizing a common set of diagnostic features. Low-level reasoning seeks to sort out the evidence provided by these diagnostic features and identify the failure mode or modes most likely present. Two approaches are employed in achieving this objective: First, an abductive rule-based reasoning approach where qualitative and quantitative knowledge of the manner in which the failure modes manifest themselves in the diagnostic features is applied to correlate the observed features to the most likely diagnosis. This knowledge can be obtained either during the design process, from similar systems already fielded, or from a FMECA. The second method applies an alternative, model-based reasoning approach building upon the effectiveness with which the knowledge applied in the first method indicts the correct failure mode. The model-based approach applies parameters, further explained below, which characterize the efficacy of evidence sources, over time, to accurately indict a given failure mode, to rank the likelihood of a given failure being present given the current observation of the diagnostic features. This second reasoning scheme is only employed when ambiguous or conflicting results are obtained from the first approach.

1. Rule-based Reasoning

Abductive, rule-based reasoning calls upon the diagnostic features for the low-level reasoning functionality. Reasoning at this level is called upon to interpret diagnostic features; map the feature values to known failure modes; and report their corresponding severity and confidence where possible. In general terms, abductive reasoning maps from a set of observations to a hypothesis which best explains or accounts for the observations⁹. Numerous approaches can be employed to accomplish rule based reasoning. Some of the more common approaches include application of hard threshold values to features to determine excessively high or low levels. Artificial intelligence approaches are also well suited to the reasoning task at this level. Neural networks and fuzzy inference systems determine diagnoses based on the set of input feature data presented to the classification system. Neural networks generally require a training step to map input data to failure modes, severity, and confidence. Fuzzy Inference Systems (FIS) require development of an expert system rulebase to interpret the soft input membership values. The concept of soft membership values, employed in FIS, allows for shades of gray to interpret the feature values relative to black and white absolute of a threshold-based classification technique. Another low-level reasoning approach, referred to as FORM (First Order Regression Method), classifies failure modes based on an N-dimensional feature space. Here, each of the diagnostic features output from a specific diagnostic module represent an axes of this N-dimensional feature space. Prior knowledge is required of the characteristic degradation paths followed by targeted failure modes as they mature and propagate through the feature space. The current set of diagnostic features represents a location in this feature space. A diagnostic assessment can be made by conducting a proximal analysis of the current observation relative to the known failure propagation paths.

Each of the approaches highlighted above yields results pertaining to a specific set of known failure modes. However, the potential remains for an ambiguous assessment from the various reasoned results. In this eventuality, an additional step is required to reduce the size of the ambiguity group and rank the remaining entries from most to least likely.

2. Model-Based Reasoning

The reasoning engine employs a second approach in the event that an ambiguity group results from the initial low-level reasoning process. Model based reasoning derives its name from its dependence upon a model of the PHM system developed for the monitored system(s), such as the models discussed above. The PHM model is loaded onto the hardware during IVHM system initialization and all model querying events take place with that embedded copy of the model. The premise of the approach is to assign each plausible failure mode a score, or rank, by considering the historical efficacy of a piece of evidence in indicting the particular failure. Evidence may originate from sources such as; Built-In-Test (BIT), diagnostic algorithms, prognostic algorithms, usage information, etc., monitoring the particular failure.

A proprietary model-based reasoning tool called ReasonPro™ has been utilized at the core of the reasoning engine. At the low level of the architecture, this reasoning technology is employed to rank the possible failure modes from an ambiguity group by analyzing various evidence sources. The premise of ReasonPro™ is to assign each ambiguous failure mode a score, or rank, by considering the historical efficacy of all the active evidence sources, Built-In-Test (BIT), diagnostic algorithms, prognostic algorithms, usage information, etc., monitoring the particular failure. The efficacy or performance over time of the evidence sources is summarized in two parameters, *Real Fault Probabilities* and *False Alarm Probabilities*. The probability of a fault existing given that an evidence source has detected it corresponds to the Real Fault Probability (RFP); the ideal monitor (a.k.a diagnostic feature, diagnostic indicator, health indicator, evidence source) has an RFP of one. The False Alarm Probability (FAP) is related to the likelihood of an evidence source activating indicating the fault, given the absence of a fault. The ideal monitor has an FAP of zero. As an example, in Figure 8, we know that a diagnostic alarm, an inspection, and a sensor value reaching a threshold are all indicators of failure mode one. In this case, the diagnostic and inspection yield a positive result and are thus referred to as Positive Evidence, but the sensor value hasn't reached the threshold value and is therefore termed Negative Evidence. Each of these evidence sources has an associated RFP and FAP which get applied, see Eq. (1), to determine the failure mode's score relative to indictment.

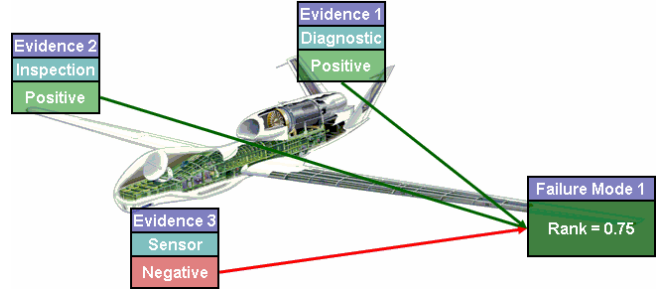


Figure 8. Model Based Reasoning Approach

ReasonPro™ also capitalizes on the notion that the absence of anticipated information in the presence of a fault (negative evidence) can also lend insight into the true health state. The core ranking equation, shown in Eq. (1), mitigates the indictment of a failure mode by the potential of the monitor going active without the presence of a fault and the inactive (negative) evidence weighted by how likely the monitor is to detect a true fault (Real Fault Probability). These values of False Alarm Probability and Real Fault Probability between each evidence and failure mode are updated based on historical information and can be updated as more data are collected.

$$RANK_{FM_{(1)}} = \frac{\sum_{j=1}^{PosE} (1 - FAP_j) - \sum_{k=1}^{NegE} RFP_k}{\text{No. of Evidence Sources}} \quad (1)$$

3. Evidence Preprocessing

The primary objective held close by IVHM system developers is to make every effort to minimize the possibility of the system issuing false alarms, positive indications of a nonexistent fault. This directive is driven by an industry-wide desire to reduce support costs associated with system operation. In addition to reduced mission readiness, the net result of issuing false alarms is a maintenance and logistics infrastructure that must sustain substantial overhead in order to support replacement of equipment that may not be faulty. To this end, a preprocessing step has been

added to the reasoning architecture which modifies the incoming CI values prior to evaluation by the reasoner. The preprocessing step has two fundamental assumptions at its core. First, components don't get healthy by themselves. A faulted LRU will remain faulted until some maintenance or repair action is taken. There are perceived exceptions to this assumption, as in the case of intermittent failures. Many intermittent failure modes are likely caused by a temporary condition which exists external to the scope of the diagnostic approaches developed to monitor the suspect LRU. These external influences, if prevalent enough to warrant the cost, are a prime target for improvement of the employed PHM approach. The concept of intermittence leads us to the second assumption: Diagnostic evidence sources have a noise floor associated with normal operation which must be accounted for by the reasoning approach. In the case of gray scale evidence, attention is required only when the values progress beyond some pre-determined level of expectation. As such, prior knowledge is required to determine the appropriate level of significance and consideration must be given to an inherent amount of variability as a particular CI transcends this level. Considering BIT evidence sources, their underlying test is also often built upon some measured value which will have an associated level of measurements noise, again manifesting in variability as the measured value crosses the initiating level.

An approach has been implemented which windows the evidence values over a short duration of time prior to processing. A weighted average is then calculated from the data as shown in Eq. (2). The weighting process serves to introduce a linear *forgetting factor* such that the latest values are given a greater significance than earlier values. Figure 9 has been included to illustrate the forgetting process. The

contributions made by the initial values are pale in significance relative to the most recent values. The weighting values linearly decrease from one to $1/(\text{length of the window})$ in decrements of $1/(\text{length of the window})$. Utilizing a weighted average in alternative to the original CI values as they appear from the HUMS modules serves to dampen, smooth, any inherent variability and help to insure that values evaluated by the reasoner are truly significant and not just a peak value from a measurement oscillation in the noise floor.

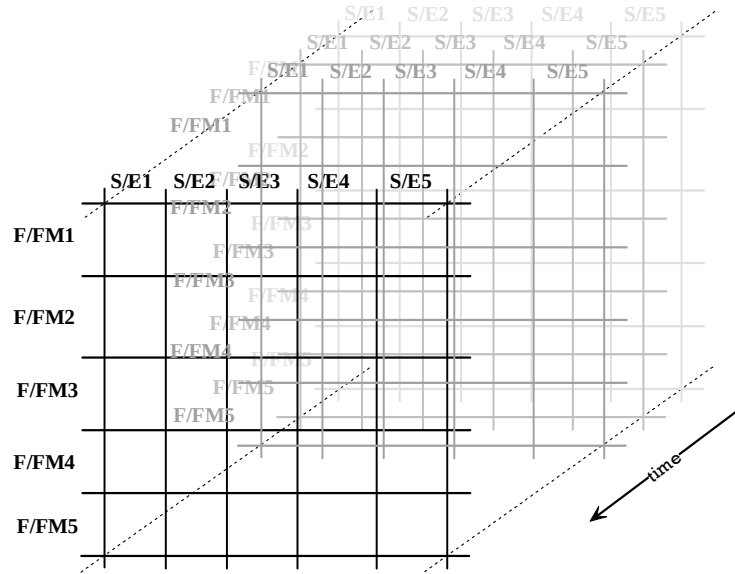


Figure 9. Forgetting Factor Applied to Evidence Sources

$$CI_w = \frac{\sum_{i=1}^N W_i \times CI_i}{\sum_{i=1}^N W_i} \quad (2)$$

where :

N = length of the window

$W = [1, 1 - 1/N, 1 - 2/N, \dots, 1/N]$

4. Severity Assessments

Detection of an emerging failure mechanism from within the condition indicators is only a portion of the information required to optimally manage vehicle health. Quantifying the severity level of the indicted fault is of equal importance. In lieu of an accurate prognostic estimate of remaining useful life, severity provides the most meaningful means of determining how big of a concern an indicted failure mode needs to be. A severe fault likely warrants immediate action, whereas an incipient fault may require little more than being placed on a 'watch' list.

In general, condition indicators have varying capabilities with regard to assessment of severity levels. BIT evidence offers little in support of determining severity due to their Boolean nature. Gray scale CIs provide the best chance of directly correlating fault severity to their inherent range of output values. In addition, these varying capabilities, or sensitivities, are not always well defined with solid discernable states. An ideal CI would exhibit a well behaved, linear correlation between amplitude and severity level, once out of the noise floor. Realistic indicators have varying degrees of correlation over their range of output. A methodology is required which facilitates the fusion of various CI values with differing correlation characteristics to level of severity.

A Fuzzy Inference System (FIS) provides an intuitive methodology to handle this shortcoming and combine and fuse evidence from numerous sources into one meaningful assessment. In effect, the methodology provides a flexible means of mapping the input space (diagnostic features) to the output space (known failure mode severity). Fuzzy inference systems utilize ‘membership functions’ to facilitate this mapping process. A Membership Function (MF) is a curve that defines how each point in the input space—the range of values of the diagnostic feature—is mapped to a membership value (or degree of membership) between 0 and 1. The input space is sometimes referred to as the universe of discourse. Through thoughtful consideration of the characteristics of the evidence source in its ability to implicate a severity, membership functions can be constructed which accurately capture and reflect this knowledge. The membership values are then combined in the mapping process using logic operators like AND, OR, and NOT to determine the implications of the inputs on the output membership functions. The final step determines an output ‘hard value’ reflecting the severity rating of the indicted failure mode.

The fuzzy inference methodology utilizes the pre-processed diagnostic features as the “hard input” values. The fuzzy logic system then “fuzzifies” the hard input by evaluating the degree of membership each hard input has in each of the fuzzy sets defined by the membership function, Low, Medium, and High in the example (See Figure 10). Fuzzy inputs are then evaluated using the rules established in the rule base. The format of the rules comprising the rule base associated with the evidence sources and failure modes follows a familiar structure—IF antecedent THEN consequent. Developing a system of expert rules would prove a daunting and impractical task and so a default assumption will be implemented that a direct correlation exists between the modes of the fuzzy sets capturing the input and the correlated output domains. That is to say, if the input is low then the output is low, etc. Provisions may be made for accommodating exceptions to this assumption.

A fundamental example of the process is provided below in Figure 10. The input and output membership functions are displayed at the top. These are simple examples of membership functions to be used when an evidence source exhibits a linear correlation between the magnitude of the hard input (diagnostic feature) and the severity level of the particular failure mode. The next rows depict the fuzzification of the input values and the corresponding implication on the output membership function. Two hard input values are provided, 5.1 and 7.4. These values would be obtained from a diagnostic algorithm. Finally the plot on the bottom left illustrates the aggregate output fuzzy set from which the final hard value is determined. Several options are available to defuzzify the final value; the geometric centroid is one method commonly used. Intuitively, the centroid method can be viewed as a "compromise" among the output actions recommended by different rules.

The FIS approach is desirable due to its scalability and flexibility. A library of membership functions is available, representative of characteristics typically observed in a diagnostic feature's response to failure mode severity. The reasoning algorithm can then call upon the most representative MF when evaluating particular diagnostic features relative to the targeted failure modes. The application of the IF—THEN rules and aggregation is commutative so the order in which the evidence sources are evaluated does not matter. Finally, a weighting can be applied to the evidence sources and rules in instances where the developer has more confidence in some diagnostic features than in others.

B. Mid-level Reasoning

Building on the health state information provided by the low-level reasoning, the mid-level reasoner will determine the functionality available from the subsystem(s) employing the component(s) indicted by the low-level reasoning. The mid and high-level reasoning will also employ a model based approach. The approach for assessing remaining functionality will be based upon the remaining functionality of the LRU in concert with the criticality of the anomalous component in enabling the subsystem to perform its role as needed. Operational criticality alludes to the functional dependence of the parent subsystem

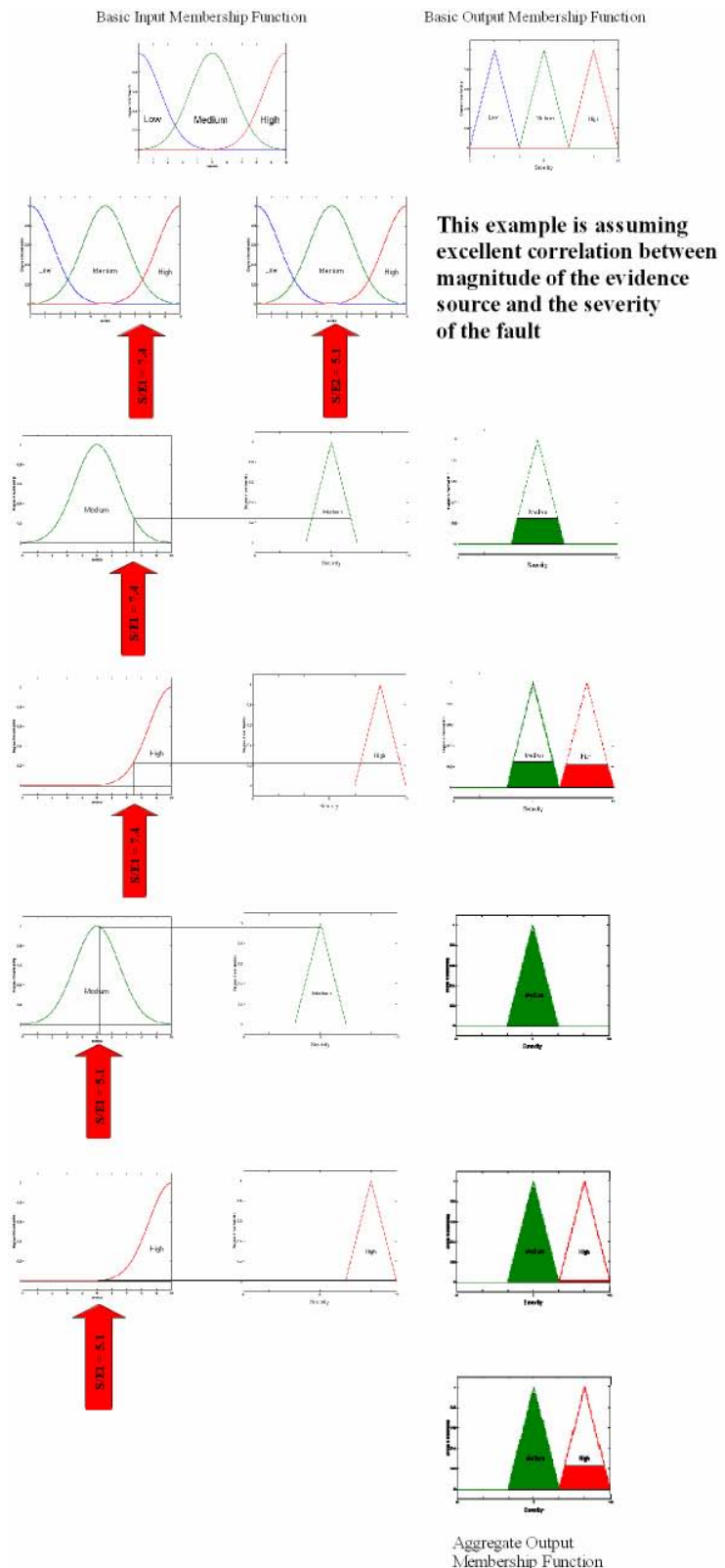


Figure 10. Fuzzy Inference Diagram of the Process Involved

employing the failing component on the faltering LRC to function as required or expected. As an example, if we consider a car's engine; the operational criticality of a timing belt breaking is great, whereas, the operational criticality of the thermostat sticking open is much lower. The criticality assessments are parameters obtained from expert knowledge and contained in the model representing the vehicle's PHM system. The operational criticality of faltering or failed components is combined into a single overall assessment of the subsystem's capacity to perform its designated role as designed.

The efficacy of the mid-level reasoner to assess subsystem functional availability is predicated on two tasks: First, the ability to initiate and execute queries of the PHM model based on results obtained from the low-level reasoning. This process traverses the functional layers of the subsystem under investigation to ensure the failure modes reported by the lower levels do not have a root cause which can be attributed to an upstream functional dependence. The effect is to traverse back up the same 'cascade down' process discussed earlier and employed in modeling the PHM system. Second, the reasoner's ability to intelligently combine results, obtained from the queries, to assess the subsystem's functional availability, as well as potentially isolate a possible root cause initiated from a component upstream in the functional dependence hierarchy. This final step combines the results obtained from querying the PHM model to verify the likely failure modes and their respective severities from the subsystem perspective and modify the indicted failure modes if necessary. These results, in concert with the operational criticality rolled up from the indicted LRC through each subsequent level in its implementation path, are used to determine the overall functional availability of the subsystem. The overall functional availability assessment made for the subsystem is then passed up to the high-level reasoner to determine its implications on the vehicle's functional availability.

The first task outlined above seeks to trace through the functional levels of the PHM system model to determine if a lower level diagnosed failure is likely caused by some upstream event. The PHM system modeling tool used here allows developers to model the propagation of failure modes from one functional area to another, illustrated in Figure 11. Utilizing this functionality one can determine the likely source of any detected failure mechanisms. The link between failure modes features a *Propagation Probability (PP)* parameter to quantify the likelihood that the upstream failure mode caused the downstream failure. A low PP value is interpreted as the higher failure mode could possibly have caused the lower fault, but not likely. A PP of one can be interpreted as the high failure mode *did* cause the lower failure mode, e.g., the severed fuel line did cause the detected drop in fuel pressure. This may not completely eliminate the reporting of the fault identified at the lower level but it may serve as a consideration when ranking them for further action.

The second aspect of the mid level reasoning process endeavors to roll up the functional limitations brought about by the detected failure modes and provide a remaining functionality assessment for the subsystem(s) containing the indicted LRU(s) and their parent functional areas. This process begins by determining the functional availability remaining at the LRU in the presence of the fault condition. Any given LRU may be comprised of a number of components providing services towards the overall execution of its designed purpose. The internal components may or may not have failure mechanisms addressed by a PHM approach. For those which are, a technique is applied to determine the effect the current diagnosed health state has on the overall capability of the LRU to function as required. This process first looks at the effect each individual failure mechanism within the component has in diminishing the overall functionality of the LRU. The functional availability is based on a combination of the severity of the diagnosed fault and the functional criticality parameter. The functional criticality (*FC*) is a zero to one value, determined by system experts and stored in the PHM model, which reflects the dependence of the LRU on the lost functionality contained in the indicted failure mode.

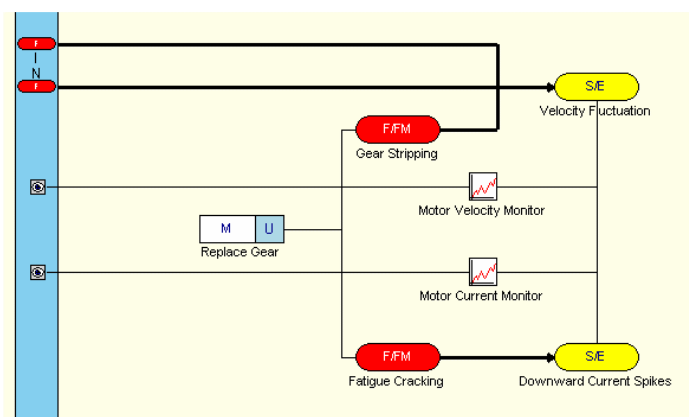


Figure 11. Failure Mode Propagation from External LRU

FC_{FM} is the functional criticality parameter for a given failure mode

Figure 1 illustrates a fault tree diagram for engine failure. The top event is 'Engine Failure' (red oval). The diagram shows the hierarchical structure of the engine failure, including components like 'Engine Control Actuators', 'Engine Performance', 'Engine Bearings', and 'Fuel System'. The diagram also includes a timeline on the left showing the sequence of events. Callouts indicate False Acceptance (FA) rates for different components: FA = 0.05 for 'Engine Control Actuators', FA = 0.2 for 'Engine Performance', and FA = 0.7 for 'Engine Bearings'.

$$F_{LRU} = \min(\forall FA_{FM}) \quad (4)$$

where FA_{LRU} is the remaining functionality of the LRU

Diagram illustrating the relationship between engine components and forces:

- Area: $F_i = A$
- Engine Control Actuators: $F_i = B$
- Engine Performance: $F_i = C$
- Engine Bearings: $F_i = D$
- Fuel System: $F_i = D$

Operational criticality is considered in cases where functionally independent components are combined. Considering first the case of independent functional areas (LRUs), as represented in Figure 13, a weighted average is applied to determine remaining functionality of the parent system, Eq. (5).

$$F_{\text{subsystem}} = \frac{\sum_{i=1}^{\text{number of constituent items}} W_{OC_i} * F_{LRU_i}}{\sum_{i=1}^{\text{number of constituent items}} W_{OC_i}} \quad (5)$$

where :

$F_{\text{Subsystem}}$ is the remaining functionality of the subsystem

W_{OC_i} is the weighting factor based on the operational criticality of the i^{th} LRU

Serial dependencies are addressed with a robust approach which has been used in the reliability domain. In a series configuration, all the components of the subsystem must be functional to maintain the required function of the subsystem. Thus, a failure of any one item of the subsystem will cause failure of the system as a whole.¹⁰ Figure 14 provides an example of a subsystem with serial dependencies among the LRU as it appears in the PHM model. Here, a product rule, Eq. (6), is applied due to the fundamental understanding that the successful operation of an LRU is contingent upon the operation of the upstream LRU and that a degraded upstream LRU lowers the best possible operation of downstream components. There is also an underlying assumption that all the components making up the parent system, monitored by PHM functionality, are equally critical to the overall operation of the subsystem and thus no operational criticality parameter is employed to the components within the upper-most child level of the subsystem.

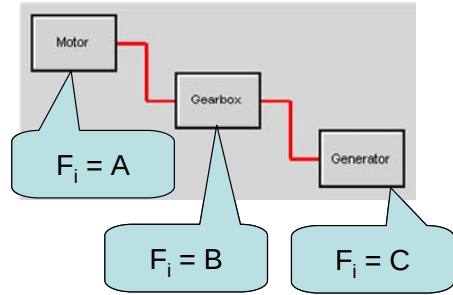


Figure 14. Subsystem Comprised of Serially Dependent LRUs

$$F_{\text{Subsystem}} = \prod_i F_i \quad (6)$$

The approach applied to subsystems with parallel dependencies among internal components also uses principles from the reliability domain. In a parallel configuration, the subsystem fails only when all the items in the subsystem fail. Thus, a failure of any one item of the subsystem will not cause failure of the system as a whole, only the failure of every item.¹⁰ Generally, all the items contained in the subsystem are presumed to be identical in function as in the case of redundant systems. Figure 15 provides an example of a subsystem with parallel dependencies among the LRU as it appears in the PHM model. Here, Eq. (7) is applied to determine the overall functionality remaining in the subsystem. Again, the same underlying assumption that all the components making up the parent system, monitored by PHM functionality, are equally critical to the overall operation of the subsystem and thus no OC parameter is employed to the components within the upper-most child level of the subsystem. It is important to recall that at this point we consider only the dependency of the parent on the child in order to function as expected when discussing operational criticality. Clearly, if we broaden our discussion to incorporate considerations of risk the criticality of the third LRU in Figure 15 is much greater than that of the first.

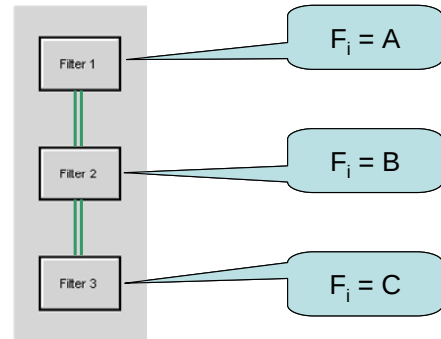


Figure 15. Subsystem Comprised of Parallel Dependent LRUs

$$F_{\text{Subsystem}} = 1 - \prod_i (1 - F_i) \quad (7)$$

The procedures outlined above are utilized to traverse from functional level to functional level within the PHM model until all child functionality indications are combined to determine the remaining capability at the parent subsystem to operate as required.

C. High-level Reasoning

The high-level (system or vehicle-level) reasoner builds upon the results obtained from the low and mid-level reasoners. The role of system-level reasoning is to examine the functional capabilities of all the constituent subsystems. This information aids in determining the overall ability of the vehicle to perform the functions or actions required to maintain operations within a predetermined list of mission critical requirements. The same mechanics are employed in determining the functional availability at the vehicle level as those employed at the mid-level. The highest level of the PHM model is interpreted, based on the presence and type of connections linking the primary functional areas, in determining which equations to call upon to combine mid-level functionality assessments.

The approach for assessing degradation of the functional availability of the vehicle is based upon an additional set of Operational Criticality values provided for the primary constituent subsystems, analogous to the approach employed at the mid-level with respect to the subsystems dependence upon the operational criticality of the LRCs. Here, the operational criticality alludes to the dependence the parent system has on faltering subsystems to function as required or expected. However, at the vehicle level this parameter can reflect safety, flight, or mission criticality. These criticality assessments are also provided as expert knowledge and embedded in the model representing the PHM system employed by the vehicle. The operational criticality of faltering or failed subsystems is combined into a single overall assessment of the system's capacity to perform its designated role as required, but can be quickly broken down into the primary constituent functional areas for a more detailed view of the current health and operational state of the vehicle.

V. IMPLEMENTATION HARDWARE

One of the factors significantly influencing the depth and effectiveness of an IVHM system is the processing capability of the supporting hardware with which the system is implemented. The design objective when considering hardware implementation was to support the advanced multi-rate signal acquisition and processing requirements of an IVHM system with maximum use of commercial-off-the-shelf (COTS) hardware. Advances in sensor technology, data acquisition, and data processing have facilitated this level of support and allow the boundaries of system health and usage monitoring to continuously expand. While sensor availability, selection, and placement are also very important factors, this discussion focuses on the data processing hardware and supporting software.

Throughout the hardware selection and accompanying software design processes, many overall design objectives needed to be considered. Objectives such as light weight, small form factor, and significant COTS support in the marketplace were primary considerations. Additional equally important considerations included environmental conditions, vehicle interfacing requirements, on-board data archiving requirements, ease of development, and data throughput, including sensor type and required bandwidth.

All of the HUMS software modules described in this paper require sensory input in order to produce meaningful output. The multi-rate data acquisition and processing requirement was motivated by the dependence on accelerometer data to support the structural, drive train, and electromechanical actuator health monitoring modules. The high frequency techniques require that the signals be acquired at high sampling rates, on the order of 50-150 kHz, even though typical sensors have a nonlinear response at these frequencies. In other cases, low bandwidth data inputs on the order of less than 100 Hz are sufficient for parameters that do not change at high rates for a given process, such as temperature and pressure or control parameters. The embedded HUMS hardware must provide signal conditioning and a constant current power supply for accelerometers, allowing high bandwidth vibration data to be directly acquired

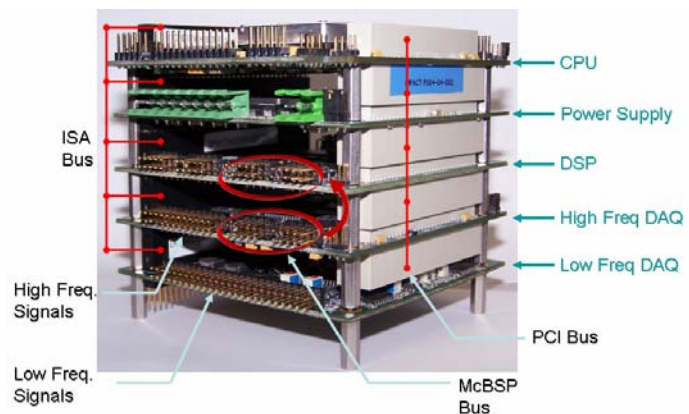


Figure 16. Embedded Reasoner COTS Hardware Configuration

from the sensor. When available, sensors already in place on the vehicle for other purposes can be leveraged to provide the necessary input. The required signals can either be acquired directly from the leveraged sensor in parallel, or the digitized data from other data acquisition systems can be accepted into the HUMS modules.

To meet the data throughput requirements of the embedded system, a COTS solution, presented in Figure 16, is employed which features two parallel acquisition and processing paths; one dedicated to high bandwidth data, the other focusing on lower speed processing requirements. Separate data acquisition (DAQ) cards are employed. The low bandwidth DAQ module is capable of acquiring 16 differential or 32 single ended analog input channels at a 500 k-samples/sec aggregate sampling frequency and 12-bit resolution. Higher bandwidth signals, such as those containing vibration data, are sampled by the high bandwidth DAQ module; this module supports 1.25 M-samples/sec on a total of 8 differential or 16 single ended 12-bit analog channels.

The digital signal processing DSP module consists of a 500 MHz fixed point processor with 128 MB of synchronous dynamic random access memory (SDRAM). Numerically intensive algorithms performed on the high bandwidth sensor data (such as vibration analysis, etc) are executed on the DSP, leaving the CPU free for system processes, and less processor intensive PHM algorithms.

The CPU module chosen for this implementation utilizes a 650 MHz Celeron processor with 256 MB of SDRAM running embedded Linux operating system. Low bandwidth processing required by the HUMS and reasoning modules, and algorithms that require floating point precision are processed here, along with any system processes required for operation.

Implementing the IVHM modules on hardware not only requires a suitable processing platform, but also an efficient data management architecture integrated into the hardware to manage and monitor data flow throughout the system. Onboard the CPU, all data transfer and algorithm interaction is handled by a proprietary communication system based on a multicast distribution approach. This architecture allows for data transfer to occur from one process to one process, one process to many processes, or from many processes to one process. The system itself is separated into four distinct layers of interaction as depicted in Figure 17. Each block within a layer is a separate system level process, this segregation of functionality not only provides logical distinction between processes and their functions, but also provides functional separation in case any process should experience an error causing it to fail.

The Input layer is the closest logical layer to the hardware; it gathers raw data from either the command shell (system inputs), the low bandwidth DAQ card, or the processed results posted from the algorithms running on the DSP and posts the results to the ioBroker residing on the Communication layer.

The Communication layer consists solely of the ioBroker, the backbone of the software architecture. Individual processes communicate with the ioBroker through use of a proprietary messaging system based on Unix sockets. Processes can either choose to subscribe to messages coming from a designated source (like data from the low bandwidth DAQ) or post messages, depending on the function of the process.

The Process layer encompasses all of the on-board IVHM processes dedicated to data analysis and reasoning. Each process chooses to subscribe to data that is required by any algorithms it requires to execute; when data is available from that source, the ioBroker will pass the data to the receiving process where it is analyzed and, if applicable, a result is returned to the ioBroker.

The Output layer is an analog of the input layer; any data that needs to be transmitted to a remote location, or stored locally for offline analysis is routed to the output layer. Processes can be added to the output layer to support communication with off-board systems. The PC-104 architecture supports a wide range of protocols such as Ethernet, USB, RS-232, RS-485, as well as a number of military grade protocols for use in almost any application.

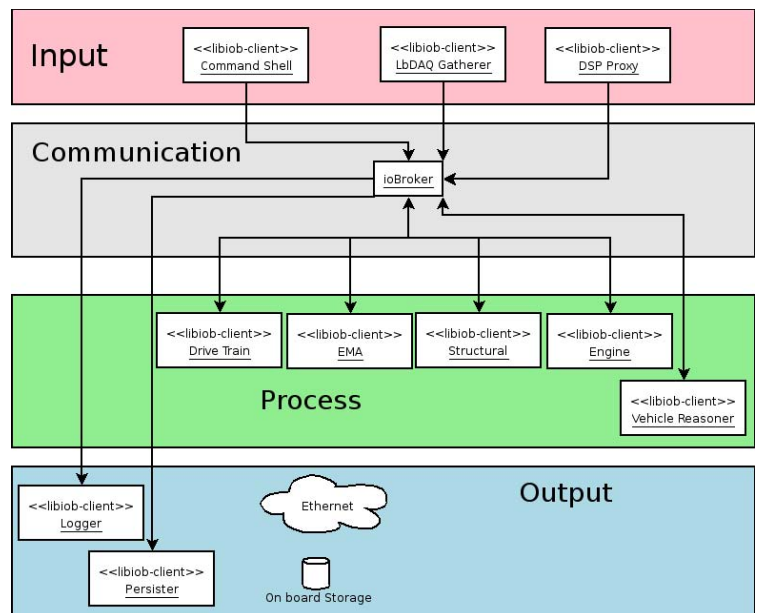


Figure 17. Block diagram of Data Communication Architecture

VI. Conclusion

The discussion provided here outlines a reasoning approach in support of Integrated Vehicle Health Management which provides real-time health state and remaining available functionality information from the various levels of functionality; LRU, assembly, subsystem, and overall system or vehicle. The architecture of the IVHM approach incorporates enhanced PHM algorithms within an enhanced health and usage monitoring system (HUMS) to provide condition indicators (CI) to the advanced reasoning architecture. The reasoner itself is tightly integrated into all functional levels of the vehicle, or system, targeted by health monitoring functionality, via the PHM model. Details extracted from the modeled PHM system are used to facilitate the hierarchical reasoning process by aiding in root cause isolation, at the low levels, and also by guiding the appropriate approach for rolling up the implications of detected low level failure mechanisms on higher level functionality throughout the various functional levels, at the mid and high levels. Implementing this functionality within an embeddable COTS data acquisition and processing platform summarizes the on-board aspect of the IVHM system. However, the on-board facet of IVHM is only a portion of the functionality required to users. Recall, IVHM was introduced as a decision support tool for operations personnel. As such, the other key aspect is presenting the health and functional availability information to personnel on the ground in an intuitive and useful manner. The ground station functionality warrants mentioning here even though outside the scope of this paper.

The ground station receives output from the embedded hardware via a combination of periodic wireless transfers and post-flight downloads. The wireless downloads contain the current state-of-health information output by the reasoner. This packet of information includes any detected failures in the replaceable units as well as the resultant functional availability assessments. Post-flight downloads also include a host of condition indicators obtained from the subsystem specific diagnostic modules and archived using onboard solid-state memory. The ground station provides a means to organize and present the health and availability information to operations personnel; offering an overview panel for easy one-source viewing of the current health state of all monitored subsystems, while also providing a means to probe deeper in the event anomalous operation is reported. Functionality is also provided to archive and trend CI data from the diagnostic modules to aid in prognostic assessments. The end result is an intuitive interface between the vehicle and its health status and the people charged with its operations.

References

-
- ¹ Fox, J.J., Glass, B.J., "Impact of Integrated Vehicle Health Management (IVHM) Technologies on Ground Operations for Reusable Launch Vehicles (RLVs) and Spacecraft", IEEE Proceeding of Aerospace Conference 2000.
 - ² Ofsthun, S., "Integrated Vehicle Health Management for Aerospace Platforms," *IEEE Instrumentation & Measurement Magazine*, September 2002, pp. 21 - 24
 - ³ Roemer, M.J., Ge, J., Liberson, A., Tandon, G.P., Kim, R.Y., Autonomous Impact Damage Detection and Isolation Prediction for Aerospace Structures, IEEE Aerospace Conference, Montana, March 2005.
 - ⁴ Orsagh, R., Sheldon, J., Klenke, C., Prognostics/Diagnostics for Gas Turbine Engine Bearings, IEEE Aerospace Conference, Montana, March 2003.
 - ⁵ Byington, C.S.; Orsagh, R., Pattada, K., Sheldon, J., DeChristopher, M.; Amin, S., Hines, J., Recent Case Studies in Bearing Fault Detection and Prognosis, IEEE Aerospace Conference, Montana, March 2006.
 - ⁶ Lee, H., Orsagh, R., An Enhancement to TSA and Filtering Techniques for Rotating Machinery Monitoring and Diagnostics, 60th Meeting of the MFPT, Virginia, April 2006.
 - ⁷ Walsch, Fletcher, Gas Turbine Performance, (pp. 113, 115, 565 & 585) Blackwell Science Ltd., ASME Press, 1998
 - ⁸ Roemer, M.J., Orsagh, R.F., Schoeller, M., Scheid, J., Friend, R., Sotomayer, W., Upgrading Engine Test Cells for Improved Troubleshooting and Diagnostics, IEEE Aerospace Conference, Montana, 2002.
 - ⁹ Josephson, J.R., Josephson, S.G., *Abductive inference; computation, philosophy, technology*, Cambridge University Press, 1994, pg. 5
 - ¹⁰ Kumar, U.D., Crocker, J., Knezevic, J., El-Haram, M., *Reliability, Maintenance, and Logistic Support—A Life Cycle Approach*, Kluwer Academic Publishers, Norwell, 2000, Chap. 4