

Advances in Integrated System Health Management System Technologies: Overview of NASA and Industry Collaborative Activities

Sunil Dixit, Steve Brown
Northrop Grumman Corporation

Amir Fijany, Han Park, Ryan Mackey, Mark James, Ed Baroth
Jet Propulsion Laboratory

ABSTRACT

This paper will describe recent advances in ISHM technologies made through collaboration between NASA and industry. In particular, the paper will focus on past, present, and future technology development and maturation efforts at the Jet Propulsion Laboratory (JPL) and its industry partner, Northrop Grumman Integrated Systems (NGIS). Through numerous collaborations in ISHM technology development programs, including DOD's Joint Strike Fighter, NASA's 2nd Generation Reusable Launch Vehicle (RLV), Orbital Space Plane (OSP), X-33 RLV, Revolutionary Concepts (RevCon), and Intelligent Systems programs, JPL and NGIS have developed extensive experience in all aspects of end-to-end ISHM systems. In particular, the partners have worked in these areas:

- 1) **Design for ISHM** – Technologies for system analysis and trade studies, model creation, verification and validation
- 2) **ISHM for Operations** – Technologies for real-time advanced diagnostic reasoning and self-assessment.
- 3) **ISHM for Maintenance** – technologies for planning repairs, controlling inventory, training repair crews, and analyzing logistics.

NGIS and NASA continue to jointly develop and mature these and other ISHM technologies under NASA's Exploration Systems Research and Technology (ESR&T) programs in support of the nation's new human space exploration initiative (<http://www.exploration.nasa.gov/>).

We will present a review of these activities and discuss technology development and maturation, lessons learned, and new direction for further development and application of ISHM technologies.

INTRODUCTION

NGIS and the NASA team have a long, successful history of working together on health management technologies. The team has jointly developed a technology base and toolsets to support numerous applications and research in health management. Under NASA's earlier Space Launch Initiative (SLI) technology risk reduction program joint NGIS/NASA teams including JPL, Ames Research Center (ARC), Glenn Research Center (GRC), Langley Research Center (LaRC), Marshall Space Flight Center (MSFC), Kennedy Space Center (KSC) and Johnson Space Center (JSC) significantly advanced the general state of the art for Integrated Vehicle Health Management (Technology Area TA-5), as well as specific advances in health management for composite cryogenic tanks (Technology Area TA-2). This collaboration is continuing with a team including NGIS, JPL, ARC, GRC, JSC, and several industry and academic partners jointly executing a new NASA ESR&T Technology Maturation Program developing and demonstrating matured ISHM technologies in support of NASA's new human exploration systems. As one example of our team heritage, the NASA Integrated Vehicle Health Management (IVHM) Virtual Test Bed (IVTB) at JPL, which was developed under the TA-5 program, leveraging facilities from earlier NASA programs, is now being used to support the new ESR&T program, continuously maximizing the value of NASA's previous investments. In addition, the NASA Centers independently continue extensive internal development of an array of advanced health management technologies, while NGIS has recently been awarded the DARPA Structures Integrity Prognostics program. NGIS ACS and NGIS AEW&EW are performing Prognostics and Health Management (PHM) on the F-35 Joint Strike Fighter (JSF) contract for

NAVAIR, and a large-scale, embedded system for enhanced diagnostics and first-stage prognostics is under development. Together our team has developed industry-leading expertise and a substantial health-management technology base.

We have developed health-management expertise, for commercial space vehicles and military aircraft, which is directly applicable to many military applications. Tools, approaches, and knowledge of system- and vehicle-level technologies will transition to numerous military domains. There are major differences in various domains. For instance, military space presents challenges that do not exist or only exist to lesser extents in other applications: reusability, need for turnaround time measured in hours/days, conditioned-based maintenance, time to target, need to complete partial missions with reduced capabilities, and rapid in-flight health management to sustain the mission while in a degraded state. The incorporation of integrated health management will improve measures in all of these areas in commercial and military domains.

TECHNOLOGIES

ISHM-related technologies are involved in all aspects of a system lifecycle, including design, operation, and maintenance (Figure 1). ISHM technologies start at the design phase where they provide tools for system analysis and trade studies. It is also the phase in which the ISHM system and its models are created using model creation tools, verified using analytical techniques, and validated at integration test facilities. The design feeds into the operational phase where the ISHM technologies are used for real-time monitoring, diagnosis, prognosis, recovery, and capability assessment of a system or system-of-systems. This requires advanced reasoning system capable of autonomous operation and self-assessment. The information generated in the operation is used in the maintenance phase for planning repairs, controlling inventory, training repair crews, and assisting mission planners / flight controllers. The loop is finally closed when the lessons learned from maintenance then feedback to the design phase where further improvements to the ISHM system can be made and the assumptions of system trade analysis can be verified.

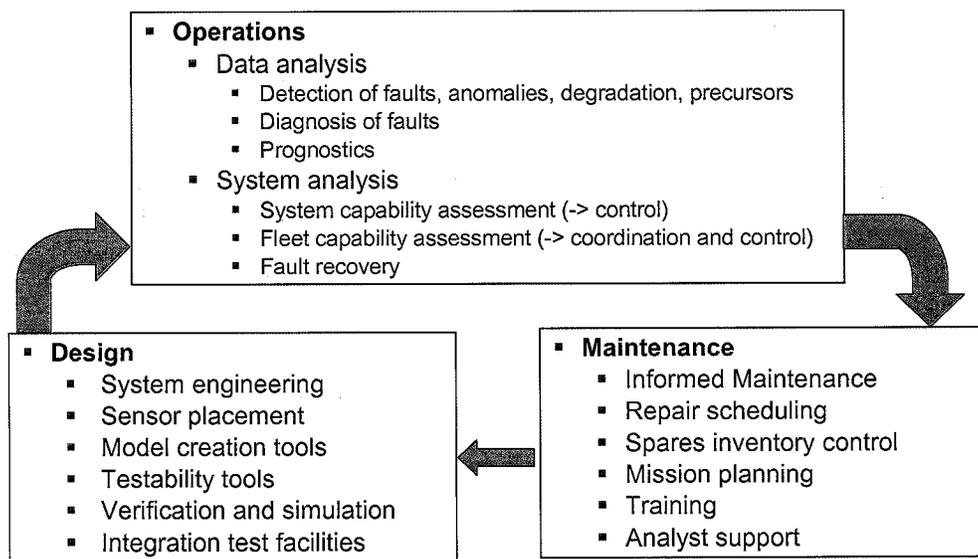


Figure 1. Involvement of ISHM-related technologies for a system lifecycle

NGIS, JPL, and NASA partners have developed extensive portfolio of technologies in all aspects of ISHM system. Below, each technology will be discussed in the context of three phases of system lifecycle.

Design for ISHM

At design phase, ISHM-related technologies have three main functions: 1) Provide tools for ISHM system design and analysis, 2) Provide environment for creating diagnostic models, 3) Verify diagnostic models and validate ISHM systems. NGIS and NASA partners have been developing system engineering

processes that will provide all three functions called the Systems Analysis and Optimization (SA&O), the IVHM Systems Management Environment (ISME), and the Diagnostics/prognostics Modeling Environment (DME). Technology-wise, JPL is developing new sensor placement approaches that will allow designers to conduct system-level trade studies with respect to cost, weight, diagnosability. For verification and validation, JPL is developing model-discovery technology that will validate diagnostic models by comparing them to actual operation of a system. The details of these processes and technologies are discussed below.

System Optimization and Analysis

The need for lifecycle process engineering in a concurrent design and development environment and one that is consistent with early development of ISHM starting from concept of the intended platform is essential for a complete representation of that platform health management system. We have developed a robust, scalable, reproducible, and well-defined systems and software engineering processes in 3-tier engineering process approach: 1) the Systems Analysis and Optimization (SA&O), 2) the IVHM Systems Management Environment (ISME), and 3) the Diagnostics/prognostics Modeling Environment (DME). These environments provide system health management design, development, integration and test (i.e., the full life cycle development - from concept to implementation), and verification validation and assurance (VV&A) using standard and formal methods. The SA&O develop standardized platform independent processes including platform independent hardware and software tools. These provide platform Cost/Benefit Analysis and Optimization of ISHM System and Component Health Manager Designs, including:

- Evaluation of Emerging, Enabling, and Current Technologies
- Dedicated Sensor Selection and Placement
- Evaluation of the Impact on Diagnostic and Prognostic Capabilities
- Evaluation of the Impact on System Operational Safety, Security and Cost.
- Evaluation of Systems and Subsystems Level Architectures

Similarly, ISME is set of platform independent processes and tools that manage concurrent and remote systems and software development in a standardized Systems Software Engineering Environment (S/SEE):

- ISME models the AOE process
- Manages concurrent development
- Captures the dependencies of the process
- Manages applicable design attributes
- Takes lower-level reliability and maintainability information and translates to system-level optimization and diagnostic tools
- Allows big-picture overview of ISHM development processes.

The fundamental goal of ISME is to provide mechanisms that allow users to specify each piece of information only once and then automatically disseminate that information to the various SA&O / DME process tools in the appropriate form for each tool. ISME's success can be measured by its ability to accomplish the data delivery process without requiring duplicate data entry or changes to any of the SA&O / DME tools.

The DME contains a suite of tools that are used to model and reason about a system. NGIS has independently developed a suite of diagnostics and prognostics reasoners that are currently being in the development of the Joint Strike Fighter (JSF) Prognostics and Health Management. The DME suite also contains reasoners that have been developed by NASA ARC and NASA JPL; some of these will be discussed in a later section. The DME provides a suite of visual modeling tools which are used to build the diagnostic and prognostic models. These models are VV&A via formal methods (model checkers) developed by NGIS and NASA collaboration.

Sensor Placement

The quality and efficiency of a diagnosis system depends on the availability and relevance of the information it can retrieve from the diagnosed plant. The quality of the measurements is expressed by the diagnosability degree, i.e., given a set of sensors, which faults can be discriminated? There is no straightforward relation between the number of sensors and diagnosability of the systems; increasing the number of sensors alone does not guarantee a higher level of diagnosability. The relevance of information provided by an additional sensor and its correlation with information provided by other sensors must also be taken into account. Besides the issue of diagnosability, we should also consider the economical/practicality issues in sensors deployment. We must provide a sensor system that achieves a desired degree of diagnosability at the lowest possible deployment cost. The different issues regarding sensor placement problem can be summarized as follows.

- *Diagnosability Degree.* Determining the diagnosability degree of a system, i.e., characterizing the set of the faults that can be discriminated.
- *Minimal Sensor Set.* Finding a minimal additional sensor set that guarantees a specific degree of diagnosability.
- *Minimal Cost Sensors.* In the case that different sensors are assigned with different costs, finding the minimal cost additional sensors which achieve a specific degree of diagnosability.

Our new approach for solving these problems is motivated by our successful method for solving the diagnosis problem [3-5]. The structural analysis of the system and the potential information carried by each sensor provide a set of relations usually called the *Analytical Redundant Relations (ARRs)* [6]. We can also consider the additional sensors (the potential sensors that will provide the desired degree of diagnosability) and their corresponding ARR. The information of all these ARR can be summarized in a *fault signature matrix* [7-10]. Then the above sensor optimization problems can be formulated as combinatorial problem regarding the signature matrix.

However, the existing and current methods for solving these combinatorial problems usually boil down to exhaustive search methods [7,8]. This severely limits the application of such methodologies for any practical system of even small size. For solving the diagnosis problem, we have recently developed an efficient branch-and-bound technique which has achieved an order of magnitude speedup over the standard algorithms. Combination of a new branch-and-bound technique and the signature matrix approach would provide a powerful efficient technique for solving the difficult problem of sensor placement optimization. To our knowledge, this is the first systematic (i.e., non exhaustive search) approach to sensor placement optimization problem. We are currently conducting further benchmark studies on this new technique.

Model-Discovery

Given a set of software test and acceptance criteria, and software that executes but does not satisfy the test conditions, the challenge is to assist a software development team in locating, understanding, and correcting the sources of inaccuracy. These inaccuracies may be caused by a wide variety of problems ranging from simple to extremely difficult in nature. Example software errors include typos in mathematical formulas, incorrect expressions, usage of approximate formulas that are themselves correct but inaccurate over the range of values required, and interaction effects between software modules such as race conditions leading to mathematical inaccuracy.

The solution of this problem is similar to model-based diagnosis, using individual test conditions and test results as trials of the software, with discrepancies between anticipated and actual results considered as faults in the system. We then apply methods of model-based reasoning to isolate the causes of such discrepancies, analyze their structure, and propose new tests (refining the software test cases) to verify or refine our conclusions.

Our new model-based diagnosis technique is a two-step method. First, the potentially errant lines in the program are isolated by dynamically mapping them onto subsets of the code, known as slices. This step is equivalent to forming a Hitting Set problem. Next, based on these slices, the minimal set of lines causing the errant results is determined through solution of the Hitting Set problem. Following the solution of the Hitting Set problem, we will have identified all potential sources of inaccuracy with respect

to variable and line number within the code. It then remains to rank these sources, determine dependency between them in the case of ambiguity, and extrapolate the probable correct form of software expressions. This approach will consider a potentially massive volume of data, both in terms of "signals" (variables) and "time" (differing input variable values), to calculate signal dependencies in a variety of overlapping "system modes" (differing static input conditions). Our analysis algorithms will identify dynamic signal dependencies and characterize them according to mode dependence or independence, sensitivity, delay, and response to previous (logged) fault events. Some of these characteristics will be anticipated – i.e. present in the software truth – with the balance representing the hidden behavior of the system. Such hidden behavior may be found in the software truth itself, in which case it suggests new expressions that must be added to our software, or it may be found in the test software, in which case it represents incorrect expressions or unanticipated interactions between variables. In either case, this two-tiered approach isolates where in the software bugs can be found, how significant each bug is to the overall result, and how each bug is manifest and may be corrected.

ISHM for Operations

For operations, the ISHM must provide real-time monitoring, diagnosis, prognosis, recovery, and capability assessment of a system or system-of-systems. This requires advanced reasoning system capable of autonomous operation and self-assessment. NGIS, JPL, and NASA partners have made progress in the areas of advanced ISHM architectures and reasoning algorithms to achieve such a system. NGIS and NASA are working towards an ISHM architectures that will support plug-and-play, i.e., interchangeable, reasoners and hardware, as well as hierarchical reasoning system. In addition, JPL has made significant progress in advanced algorithms that can diagnosis and isolate faulty components faster, thereby increasing the size or complexity of problems that an ISHM system can diagnosis. Similarly, hybrid reasoning technologies will allow great number of types of problems, such as mixed continuous and discrete states, to be diagnosed. Additionally, JPL is developing technology for self-assessment known as the Capability Assessment Manager that will simplify interface between the ISHM system and the autonomous mission managers or the flight crew. The details of these technologies are discussed below.

Plug-and-Play ISHM Architecture

In order for ISHM systems to be affordable and sustainable, it must be modular such that software and hardware components can be interchangeable. This will ensure that the ISHM system can be easily upgradeable in the future when new algorithms or sensors are developed after it is put into operational service. For ISHM systems to be modular and plug-and-play, ISHM architecture and interface standards must be developed.

NGIS, JPL, and NASA partners are working towards the design, evaluate, and validate the operation of a modular plug-and-play ISHM architecture and to support an end-to-end ISHM system. Leveraging and extending Open Systems Architecture for Condition Based Maintenance (OSA-CBM) standards, the team is defining interface standards for subsystem and system-level health management systems. Identified as the most representative subsystems for space operated vehicles, the first subsystems for which plug-and-play interface standards will be defined are power, life support, propulsion, and structures. The subsystem interfaces will be consistent with OSA-CBM as well as accommodate the characteristics of avionics, guidance, navigation and control (GN&C) sensors, actuation, and other subsystems. As an example, modular power system interfaces will be developed that can automatically manage the process of safely connecting power systems and use the combined power resources of an assembled configuration of multiple intelligent modular systems such as the automatic reconfiguration of power systems as mediated by ISHM interactions after linkup of two or more mobile habitation modules. Such system would be vastly improved over the process of reconfiguration for the International Space Station (ISS) assembly.

System-level Reasoning

NGIS, JPL, and NASA partners are developing hierarchical and distributed architectures for system-level reasoning. Related to the plug-and-play ISHM architecture, the goal is to develop interfaces and protocol between subsystem and system-level health managers to support ISHM systems that can detect and isolate faults across subsystems. The challenges include appropriate level of abstraction of health state

at the subsystem level, i.e., minimize exchange of direct sensor readings to the system-level reasoners but at the maintaining sufficient information for system-level reasoner to isolate the root cause, and integration of diverse set of algorithms at the subsystem and system levels.

Advanced Model-based Diagnosis Algorithm

In the current state of practice, the most disciplined approach to diagnosis is the “model-based” approach, employing knowledge of device operation and connectivity in the form of models [13]. This approach, which reasons from first principle, provides far better diagnostic coverage than traditional approaches based upon collection of symptom-to-suspect rules. However, there are two major drawbacks in current model-based diagnosis systems that severely limit their practicality. First, these systems tend to be large, complex, and difficult to apply. Second, in order to find the minimal diagnosis set (i.e., the minimal set of components that, if faulty, would fully explain the anomalous behavior detected), they rely on algorithms with an exponential computational cost, which makes them highly impractical for application for many systems of interest.

The most widely used approach to model-based diagnosis consists of a two-step process: (1) generating conflict sets from symptoms; (2) calculating minimal diagnosis set from the conflicts. Here a conflict set is a set of assumptions on the modes of some components that is not consistent with the model of the system and observations, and a minimal diagnosis is a set of the consistent assumptions of the modes of all components with minimal number of abnormal components. For finding minimal diagnosis from the conflicts, the most common algorithm is based on Reiter's algorithm [14], which requires both exponential time and exponential space (memory) for implementation.

We have developed a new approach for Model-based Diagnosis by addressing the problem of generating the minimal diagnosis from the conflicts [3-5]. This problem can be formulated as the well-known Hitting Set Problem. Our approach starts by mapping the Hitting Set problem onto a special case of 0/1 Integer Programming Problem that enables us, for the first time, a priori determination of the lower and upper bounds on the size of the solution. Based on these bounds, we introduce a new concept of solution window for the problem. We have also developed a new branch-and-bound technique that not only is faster than the current techniques in terms of number of operations (by exploiting the structure of the problem) but also, using the concept of solution window, allows a massive reduction (pruning) in the number of branches. Furthermore, as the branch-and-bound proceeds, the solution window is dynamically updated and narrowed to enable further pruning. The concept of window also allows us to propose a new portfolio approach, i.e., combining several different algorithms, for the problem. In this sense, several other fast algorithms (e.g., Randomized Algorithm), which usually lead to sub-optimal solutions, can be run in parallel with the branch-and-bound algorithm. The sub-optimal solutions generated by these algorithms are then used for further dynamic updating of the solution window. We have conducted extensive benchmarking and have presented the results of the performance of the new algorithm on a set of test cases in [5]. These results clearly show the advantage of our new algorithm over the best known branch-and-bound algorithm for the problem; in fact the new algorithm has achieved several orders of magnitude speedup over the standard algorithms. These results clearly demonstrate the feasibility of fast model-based diagnosis of many systems of interest of large size.

Hybrid Reasoning System

The space exploration systems will require very robust, on-board diagnostic/prognostic reasoner at the core of our ISHM system if we are going to significantly increase the safety margins for crews on complex spacecraft and habitats at very long distances. As a system-of-systems, the health management systems will have to handle various types of data, e.g. quantitative, qualitative, static, dynamic, etc., and isolate complex interacting failures while operating in real-time on a distributed network. Currently, all of these capabilities do not exist in one system, and NGIS, JPL, and NASA partners are working to addressing these needs by developing hybrid reasoning systems and architectures.

The goal is to develop a general approach to synergistically combining algorithms, in order to create a hybrid reasoning system with the strengths of all while minimizing or eliminating their weaknesses. The process starts with the requirements for the hybrid reasoner for a particular application and desired characteristics. Through a systematic survey of the strengths and weakness of different classes of

algorithms, a list of potential algorithms that match the requirements will be developed. These may include model-based, rule-based, neural network, statistical, and data-mining techniques. The algorithms will be then combined in a manner that will emphasize the strengths of each class while reducing or eliminating its shortcomings with a complementary class(s). The goal is develop set of standards and interfaces that allow disparate classes of algorithms to co-exist and share information. The issues that the team is tackling in combining or integrating disparate algorithms are model compatibility, model consistency, architectural constraints, interface definitions, scalability, and temporal responsiveness.

Capability Assessment

The state information that the ISHM system provides should be more than simply the current state. It should provide a full capability assessment, i.e., a capability assessment manager (CAM). This is because an ISHM system with capability assessment can dramatically simplify the decision process. For a human crew or operator, it can reduce the decision process to a set of options rather than a mental construct of the entire system and the effect of the fault on the system. Likewise, it can greatly simplify the interface between the automated planner and the ISHM system since only high-level goals and options need to be exchanged rather than full details of the fault. Furthermore, the modeling effort is reduced since the planner does not need to maintain as a detailed model of the system for recovery planning. Finally, capability assessment serves as an excellent testing point for semi-autonomous systems by assessing the efficiency and accuracy of the decisions made by either humans or machines when presented with a set of possible options.

The requirement for a CAM is the capability to estimate the present state and future states that are achievable in the future. Formally stated, the CAM must answer: given N observations and current state S_t , what states (P_1, P_2, \dots, P_v) are reachable in $t+M$ steps (Figure 2). Such a CAM can be built by extending model-based reasoners such as MEXEC [1]. Given N set of previous states, MEXEC can compute an $N-1$ step reconfiguration plan for a target state if such a plan exists. By enumerating all the relevant and achievable target states, it is possible to generate a list of possible future states or goals, i.e., capability assessment.

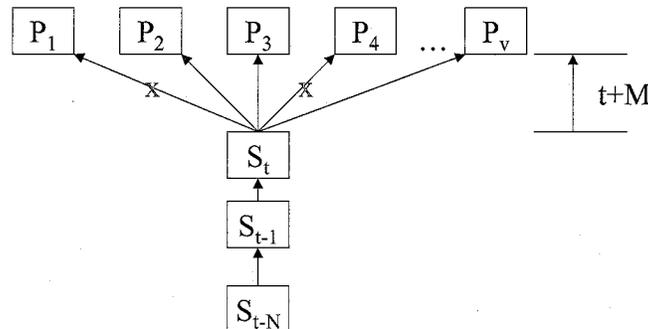


Figure 2. CAM: Given N observations and current state S_t , what states (P_1, P_2, \dots, P_v) are reachable in $t+M$ steps (time)? X denotes unreachable state in time $t+M$.

ISHM for Maintenance

The last phase that ISHM is involved in is the maintenance. From civil to military, NGIS has been involved in all aspects of maintenance including planning repairs, controlling inventory, training repair crews, and analyzing logistics. The recent work that NGIS and NASA partners have initiated include informed maintenance system for long-duration space flight. At very long distances (with lengthy communication delays) or long duration missions, it is simply impractical to have mission control involved in the real-time supervision and guidance of on-board maintenance activities. Furthermore, members of a relatively small crew operating a large, complex spacecraft, of necessity, need to be generalists in terms of the breadth of their knowledge of spacecraft systems. As a result, when the ISHM system determines that it is necessary to perform an unexpected on-demand maintenance operation, or even for scheduled maintenance activities, the crew should have ready and effective access to the encyclopedic quantities of maintenance information and procedures readily contained in IETMs and made available to the crew via convenient PMAs.

NGIS and NASA partners are maturing informed maintenance systems that leverage COTS and military-informed maintenance systems. These include PMA system that could be made available in a shirt sleeve version as well as in a wearable "eyes up" version for space-suited extravehicular activity (EVA) activities. The same system could also be used for multiple other purposes including:

- To access other essential information in science, medicine, and engineering.
- To record and play back personal audio/video/email communications.
- Entertainment including movies, electronic books, and video games.

The system, called MIMAS, will be based on OSA-CBM principles and standards. MIMAS will support both scheduled and unscheduled maintenance activities. Unscheduled maintenance utilizes both the hardware and software of the ISHM to detect, isolate, and remediate the immediate effects of an anomaly and to present information to the flight crew and ground-based mission management personnel.

MIMAS will also incorporate a scheduled maintenance support capability that informs the crew of regular maintenance actions that are required. Scheduled maintenance activities can also be updated using the prognostics capabilities of the ISHM system, so that MIMAS can recommend a convenient time to replace or repair the deteriorating hardware before a failure occurs. MIMAS will be supported by a PMA that will be designed and developed with ISHM and modular subsystem interfaces. The PMA will be designed to host IETM that will step the user through required procedures to affect repairs as required and verify corrective actions.

ISHM PROGRAMS

The technologies discussed in the previous section were (are being) developed in part by these selected ISHM technology programs by NASA and DOD. Particularly for space application, NASA's 2nd Generation Reusable Launch Vehicle Program Technical Area 5 (TA-5) laid the foundations for process for ISHM system design, and NASA's Exploration Systems Research and Technology program will lay the foundation for hybrid reasoning, modular ISHM systems, and on-board informed maintenance systems. In addition, NASA's Intelligent Systems program has provided funding for advanced research such as model-based diagnosis technology. For terrestrial applications, DOD's Joint Strike Fighter (JSF) will provide a first production ISHM system platform for large-scale, embedded system for enhanced diagnostics and prognostics.

Past

X-33 Avionics Flight Experiment

The X-33 Avionics Flight Experiment (AFE) was a flight processor that was intended to provide a platform for demonstrating a collection of automated tools and techniques for reducing operational and maintenance costs in space planes. Built at JPL, the AFE was to be flown on the NASA X-33 to validate automated techniques.

The need for AFE arose because of a lack of readily accessible flight hardware for sophisticated AI software tools in validating AI approaches aimed at reducing operational and maintenance costs in space planes. The effect of not having a flight hardware testbed has been to either hamper the artificial intelligence programming techniques that could be used effectively in an application or to cause AI programmers to redevelop their own software tools to be used on flight hardware [2]. The limitations of such software tools have historically included such problems as documentation, poor access to facilities at different levels, lack of modularity, poor run-time efficiency, inadequate debugging facilities and access to source code and lack of tools to support the most advanced reasoning techniques.

The AFE project used two JPL-developed tools to examine of how AI tools can be used for reducing operational and maintenance costs. Test Specification Language (Tspec) was used for the automated testing and verification of portions of the AFE's software and hardware. Spacecraft Health Inference Engine (SHINE) was used for monitoring, analysis and diagnosis of portions of the AFE hardware. SHINE was to detect and diagnose in-flight faults and record the resolutions on the Vehicle Health

Manager (VHM) log of the X-33 through the 1773 system bus. Both these tools were used on the AFE during system integration and Test (I&T) and were proposed for flight demonstration. Both SHINE and Tspec was designed to run well in environments where system resources such as processor cycles and memory are at a premium. Both of these systems have been demonstrated in stand-alone advisory systems for human operators as well as components of embedded systems. Both of the tools generate C++ code which allows them to run efficiently in flight systems with real-time operating systems such as VxWorks.

Second Generation RLV Program TA-5

The 2nd Generation Reusable Launch Vehicle Program Technical Area 5(TA-5) focused on reducing risk for IVHM technology required for NASA's proposed 2nd Generation RLV which had requirements of high flight rates and quick turn-around times. Led by NGIS with NASA partners ARC, JPL, and GRC, the project demonstrated key technologies in the development of (1) an IVHM System Requirements Document (SRD) and database, (2) system cost models indicating a 50 percent reduction in operations and supportability costs, (3) systems analysis and optimization (SA&O) process handbook, (4) SA&O tool suite, (5) diagnostic modeling and (6) verification & validation (V&V) tool suite. Key flight technology accomplishments were (1) Next Generation Launch Technology (NGLT) baseline IVHM concept of operations, system functional architecture, and interface requirements, (2) IVHM system/subsystem reference software architecture, (3) "laptop" IVHM functional architecture demonstrator (IFAD), (4) high-fidelity simulation-based subsystem health management validation (Propulsion IVHM Technology Experiment (PITEX)), and (5) simulation-based system health management validation using the IVHM virtual test bed (IVTB).

Present

NASA ESR&T Program

NGIS, JPL, and NASA partners are currently working on a task entitled "Intelligent Modular ISHM Systems" that seeks to develop technologies for a "plug and play" ISHM system. The extended duration and distances of lunar and deep space missions require timely, reliable, and robust diagnostic and prognostic capabilities for real-time decision-making, reconfiguration of safety critical systems, and on-board maintenance planning and support. To maximize overall system flexibility and effectiveness these capabilities must be seamlessly integrated into the "Intelligent Modular Systems" (IMS) that will provide the fundamental building blocks of reconfigurable spacecraft, mobile surface habitats, and other modular systems.

The project addresses the full range of technologies and implementation issues associated with providing "plug and play" Integrated System Health Management (ISHM) capabilities for Intelligent Modular Systems. The project will address four major technology areas integral to the operation of a complete on-board ISHM system:

Modular ISHM System Design and Analysis - Based on experience gained by the NGIS team during NASA's Space Launch Initiative (SLI) TA-5 Integrated Vehicle Health Management (IVHM) program, an open systems ISHM architecture compatible with the goals of the project will be developed. It will also be enabled by the development of automatic "plug and play" methods for integrating independent subsystems into a unified ISHM system when such units are assembled. System Analysis & Optimization (SA&O) tools will be applied to quantify benefits.

Robust Hybrid Diagnostic/Prognostic Reasoner - In order to provide the advanced level of fault detection, isolation, and recovery capabilities required for future NASA systems, a hybrid combination of diagnostic and prognostic reasoning methods be developed that will demonstrate that a robust on-board diagnostic reasoning capabilities that can address even unexpected multi-level failure modes that could endanger mission and crew.

Subsystem Health Management - To effectively develop and demonstrate a complete ISHM system in a relevant environment, it is essential that key representative subsystems be addressed. In particular, power, environmental control, life support, and structural health monitoring systems will be developed.

Special attention will be paid to the design, development, and demonstration of subsystem interfaces that are consistent with OSA-CBM standards.

On-Board Informed Maintenance (IM) System. The need to guide and train crew members in the performance of potentially complex, but infrequent, maintenance tasks leads to a requirement for the development and incorporation of a "modular informed maintenance System" (MIMAS). This system, based on informed maintenance methods currently applied to military and commercial aircraft, will include the development of portable maintenance aids (PMAs) and interactive electronic technical manuals (IETMs) to demonstrate the effectiveness of the system in reducing maintenance man hours per flight hour (MMH/FH) and to maintain crew proficiency on long voyages.

NASA Intelligent Systems (IS) Program

JPL is currently working on a project funded by NASA's IS program with the objective of building and demonstrating a hybrid reasoning engine with two unique features: prognostic forecasting and reasoning on incipient faults, and improved operation in degraded systems. Three new capabilities are being developed and include:

1. Data Forecasting: Added capability to anomaly detection. Uses a combination of physical modeling and statistical methods to forward-project data values for virtually any time-varying sensor signal. Provides future "high water-mark" values of requested signals along with estimated time of event and confidence.

2. Prognostic Hypothesis Scenario Generator: This is a meta-reasoning capability that repeatedly calls a diagnostic engine/model to provide prognoses and prognosis tracking. The SG takes as its input the current state of a system, including probabilistic information from Data Forecasting. Using model-based reasoning techniques it returns an ordered list of fault scenarios that could be generated from the current state, i.e. the plausible future failure modes of the system as it presently stands. The SG models a Potential Fault Scenario (PFS) as a black box whose input a set of states tagged with priorities and whose output is one or more potential fault scenarios tagged by a confidence factor. The results from the DSG are used by a model-based diagnostician to predict the future health of the monitored system.

3. Enhanced Diagnosis Engine: We require powerful diagnosis to positively detect hard faults from the given symptoms. We also require a method to discriminate partial symptoms, i.e. the majority of prognostic cases from non-recurrent intermittents, which we propose to do through forecasting.

Joint Strike Fighter

NGIS ACS and NGIS AEW&EW are performing this program for NAVAIR. A portion of the JSF contract is to develop the prognostics and health management (PHM) system for the Joint Strike Fighter (JSF). A large-scale, embedded system for enhanced diagnostics and first-stage prognostics is under development. Model-based-reasoning (MBR) algorithms for advanced diagnostics, diagnostic models of complex on-board systems, and a toolset to support model creation and testing are also being developed. A baseline embedded PHM architecture and detailed design has been developed. A software object design for the baseline PHM system is being implemented. First stage diagnostic algorithms, initial subsystem diagnostics models and prognostic candidates, and initial prediction methods have been developed. The F-35 design has demonstrated the efficacy of this approach and how NGIS's model-based reasoning can effectively integrate disparate sensors as well as integrate the results from other sub technologies, such as neural networks.

Reliable Software Systems Development

The Reliable Software Systems Development activity is a JPL project awarded as a NASA ESR&T program. The team members are JPL, NASA ARC, academia, and NGIS. We and NASA fit loosely under the heading of code synthesis. The objective is to develop and use formal specification languages to

generate source code and to generate "proof obligations" where an obligation is a theorem whose truth entails some useful property of the generated code. Our specific domain is display systems for safety and mission critical applications. Proof obligations fall into two buckets: domain-specific obligations such as top level graphics objects do not overlap or there are conditions, for every object, such that they will be displayed; and application-specific obligations that are conditions included in the formal specification such as the propulsion system display must be visible during powered flight. The basic ideas are (1) discharging the proof obligations will greatly reduce the V&V effort necessary to certify the resulting code and (2) the formal specification step will provide a readable concise summary of code behavior that will also contribute to reduced V&V effort.

ACKNOWLEDGEMENTS

The research of the first two authors described in this paper was carried out at Northrop Grumman Integrated Systems Western Region, under several contracts with the National Aeronautics and Space Administration (NASA), Air Force, and Navy.

The research of JPL authors, described in this paper, was performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (NASA).

REFERENCES

- [1] Barrett, A., "Model Compilation for Embedded Real-Time Planning and Diagnosis," Proceedings of the 4th International Workshop on Planning and Scheduling for Space, Darmstadt, Germany, 2004.
- [2] Atkinson, D., "Artificial Intelligence for Monitoring and Diagnosis of Robotic Spacecraft," Chalmers University of Technology, Sweden, Technical report 237, 1992.
- [3] A. Fijany, F. Vatan, A. Barrett, M. James, C. Williams, and R. Mackey, "A novel model-based diagnosis engine: Theory and Applications," Proc. 2003 IEEE Aerospace Conference, March 2003.
- [4] A. Fijany and F. Vatan, New High Performance Algorithmic Solution for Diagnosis Problem, Proc. of 2005 IEEE Aerospace Conference, March 2005.
- [5] A. Fijany and F. Vatan, A New Approach for Efficient Diagnosis of Large and Complex Space Systems, Full paper accepted for presentation at Int. Conference on Recent Advances in Space Technologies (RAST), June 2005.
- [6] Cordier, M-O., Dague, P., Levy, F., Montmain, J., Staroswiecki, M., and Travé-Massuyès, L., "Conflicts versus analytical redundancy relations: a comparative analysis of the model based diagnosis approach from the artificial intelligence and automatic control perspectives," IEEE Transactions on Systems, Man, & Cybernetics, Part B: Cybernetics, Vol. 34, No. 5, Oct. 2004, pp. 2163–2177.
- [7] Spanache, S., and Escobet, T., "Fault diagnosability: a component-oriented approach," Proceeding of Workshop on Identification, Optimization, and Control of Hybrid System, Valencia, Spain, January 2004.
- [8] Travé-Massuyès, L., Escobet, T., and Milne, R., "Model-based Diagnosability and Sensor Placement Application to a Frame 6 Gas Turbine Subsystem," Proceedings of IJCAI, 2001, pp. 551–556.
- [9] Travé-Massuyès, L., Escobet, T., and Spanache, S., "Diagnosability analysis based on component supported analytical redundancy relations," Proceedings of 5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, 2001, pp. 897–902.
- [10] Yan, Y., "Sensor placement and diagnosability analysis at design stage," Proceedings of MONET Workshop on Model-Based Systems at ECAI, Valencia, Spain, 2004.

[11] A. Fijany and F. Vatan, "A New Method for Sensor Placement Optimization", Proc. of 41st AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Tuscan, AZ, July 2005.

[12] A. Fijany and F. Vatan, "A Unified and Efficient Algorithmic Approach to Model-based Diagnosis and Optimal Sensor Placement", To appear in Proc. of 8th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS), Munich, Germany, September 2005.

[13] J. de Kleer, A. K. Mackworth, and R. Reiter, "Characterizing diagnoses and systems," Artificial Intelligence, vol. 56, 1992.

[14] F. Wotawa, "A variant of Reiter's hitting-set algorithm," Information Processing Letters, vol. 79, 45-51, 2001.

DISTRIBUTION STATEMENT

Approved for public release, distribution is unlimited.