Extended Abstract:

General Purpose Data-Driven System Monitoring for Space Operations

for the InfoTech@AeroSpace 2009 Special focus session on Integrated System Health Management chaired by Fernando Figueroa

David Iverson, Rodney Martin, Mark Schwabacher, Lilly Spirkovska - *NASA Ames Research Center*
Ryan Mackey – *California Institute of Technology, Jet Propulsion Laboratory*
J. Patrick Castle - *Mission Critical Technologies*
William Taylor - *NASA Ames Research Center*

As modern space propulsion and exploration systems improve in capability and efficiency, their designs are becoming increasingly sophisticated and complex. Determining the health state of these systems using traditional parameter limit checking, model-based, or rule-based methods is becoming more difficult as the number of sensors and component interactions grow. Data-driven monitoring techniques have been developed to address these issues by analyzing system operations data to automatically characterize normal system behavior. System health can be monitored by comparing real-time operating data with these nominal characterizations, providing detection of anomalous data signatures indicative of system faults or failures.

Data-driven techniques have a number of advantages over other methods for monitoring complex space vehicles. Unlike model-based systems, the developer does not need to understand or encode the internal operation of the system. The knowledge required to monitor the system is automatically derived from archived data from system operation. Unlike rule-based systems, data-driven systems do not require system analysts to define nominal relationships among sensors. Analysts can and often do determine these relationships for a system with few sensors; it is more difficult to analytically determine the nominal relationship among a large number of sensors. Data-driven techniques are not limited to low-dimensional spaces and work as effectively with dozens of parameters as they do with a few. Knowledge bases formed by data-driven techniques are also easy to update. As the operating envelope of the monitored system is expanded, data-driven techniques can be quickly retrained to incorporate the new behavior into the knowledge base. The expertise and time-consuming process of updating a model or rule base to maintain consistency with the new operation is not required.

The Inductive Monitoring System (IMS) is a data-driven system health monitoring software tool that has been successfully applied to several aerospace applications. IMS uses a data mining technique called clustering to analyze archived system data and characterize normal interactions between parameters. This characterization, or model, of nominal operation is stored in a knowledge base that can be used for real-time system monitoring or analysis of archived events. System data is compared with the nominal IMS model to produce a measure of how well current system behavior matches the normal behavior defined by the training data. Significant deviations from the nominal

system model can provide alerts to system malfunctions or precursors of significant failures.
The scope of IMS based data-driven monitoring applications continues to expand with current development activities. Successful IMS deployment in the International Space Station (ISS) flight control room to monitor ISS attitude control systems has led to applications in other ISS flight control disciplines, such as thermal control. It has also generated interest in data-driven monitoring capability for Constellation, NASA's program to replace the Space Shuttle with new launch vehicles and spacecraft capable of returning astronauts to the moon, and then on to Mars. Several projects are currently underway to evaluate and mature the IMS technology and complementary tools for use in the Constellation program. These include an experiment on board the Air Force TacSat-3 satellite, and ground systems monitoring for NASA's Ares I-X and Ares I launch vehicles.

The TacSat-3 Vehicle System Management (TVSM) project is a software experiment to integrate fault and anomaly detection algorithms and diagnosis tools with executive and adaptive planning functions contained in the flight software on-board the Air Force Research Laboratory TacSat-3 satellite. The TVSM software package will be uploaded after launch to monitor spacecraft subsystems such as power and guidance, navigation, and control (GN&C). It will analyze data in real-time to demonstrate detection of faults and unusual conditions, diagnose problems, and react to threats to spacecraft health and mission goals. The experiment will demonstrate the feasibility and effectiveness of integrated system health management (ISHM) technologies with both ground and on-board experiments. Initially, the TVSM software will run open loop, providing system health information and recommendations to ground operators, without automatically performing fault-mitigating corrective actions. After the end of the satellite's mission, closed loop tests combining TVSM monitoring and diagnosis with reactive capabilities by the flight software will be performed. In addition to monitoring for long periods of actual operation, the experiment will include fault injection into TacSat-3 data as well as commanded operations to test and evaluate automatic ISHM monitoring and recovery under controlled conditions.

The ongoing Ares I-X Ground Diagnostics Prototype project is evaluating the same set of software tools as the TVSM project. They will be used for detecting and diagnosing faults in the Ares I-X first-stage solid rocket booster (SRB) thrust-vector control (TVC) system and associated ground support equipment. These tools will be used at the Kennedy Space Center (KSC) during Ares I-X vehicle integration and testing activity in the vehicle assembly building (VAB) and while the vehicle is on the launch pad. IMS will be integrated with two other software tools: TEAMS, a model-based reasoning tool from Qualtech Systems Inc., and SHINE (Spacecraft Health Inference Engine), a rule-based expert system from the Jet Propulsion Laboratory (JPL). SHINE rules will be used to determine TVC system operating modes. IMS will have a dedicated knowledge base for specific operational modes or tests, and will dynamically load the appropriate knowledge base when SHINE reports a mode change. Since the Ares I-X TVC systems are expected to be very similar to Space Shuttle systems, IMS knowledge bases will be constructed using historic SRB data from the Shuttle program. Anomalies detected by

IMS in the monitored Ares I-X systems will be used to corroborate and possibly identify precursors to anomalies detected by the TEAMS tool, which will use system dependency models hand coded by NASA engineers to diagnose the cause of those anomalies. The three tools will be interfaced with live data from the Ares I-X vehicle and ground hydraulics. The outputs of the tools will be provided to Ares I-X mission controllers located at KSC.

We are also evaluating IMS for continued use after the Ares I-X mission to monitor vehicle and ground systems for the Constellation program. We are developing a prototype IMS application to monitor the liquid hydrogen (LH2) ground support equipment (GSE) required to fuel the Ares I launch vehicle. In addition to proving feasibility, one goal of that effort is to perform a trade study to determine what types of systems lend themselves well to a data-driven approach. Another goal is to document the effort (time and cost) required for building an IMS application. Because the Constellation vehicles and much of the supporting ground equipment are still under development, archived data is not currently available. Working with existing equipment to determine the characteristics of suitable systems and the effort required to build an application will allow us to effectively plan future deployments of IMS and similar data-driven systems. Lessons learned from IMS application to Ares I-X systems will also provide valuable insight for determining effective uses of data-driven techniques in the Constellation program.

The data-driven approach can also be applied to fleet supportability tasks. Fleet supportability is typically associated with large fleets of similar equipment, for example, a fleet of F/A-18 aircraft. The operating characteristics of the fleet are used to develop a failure distribution. This profile can then be used to predict the failure of an individual component on similar instances of the fleet type. Thus, we can infer that a fuel pump may fail in as few as 500 hours or as many as 2000 hours but the majority of pumps fail at 1000 hours. This information can be used to extend (or shorten) maintenance periods to maintain a desired in-service failure rate. The fleet size of the Constellation program will be much smaller than the typical aircraft fleet. Rather than evaluating the performance of components in thousands of instances, a data-driven approach, like IMS, can also be used to evaluate the performance of reusable (limited iterations) and expendable components. Performance-degrading conditions can occur throughout a component's lifetime, from design through launch and reentry to refurbishment. Performance of new components can be adversely affected during design, manufacturing, transport, or prelaunch activities. Performance of recovered components can be adversely affected during any previous launches, reentry and recovery, or refurbishments. Numerous prelaunch tests verify compliance with expected performance. We anticipate that IMS can complement these tests by detecting that the performance on a test is still within limits but is different than on previous tests either on this system or on previous systems of the same type. It may also be possible to train IMS on data used to construct fleet supportability failure distributions. In this way, a supervised machine learning paradigm can be used in lieu of the standard unsupervised machine learning paradigm, where IMS is trained only on nominal data.

In summary, as a common thread of discussion in this paper we will employ the evolution of a candidate data-driven technique, IMS, as related to several ISHM elements. Thematically, the projects listed will be used as case studies. We will demonstrate the maturation of IMS via projects where it has already been deployed, or is currently being integrated to aid in fault detection. We will also show how IMS can be used to complement a suite of other ISHM tools, providing initial fault detection support for diagnosis.