NASA/TM—1998-208807

Real-Time Sensor Validation, Signal Reconstruction, and Feature Detection for an RLV Propulsion Testbed

Amy L. Jankovsky
Lewis Research Center, Cleveland, Ohio

Christopher E. Fulton
Analex Corporation, Cleveland, Ohio

Michael P. Binder, William A. Maul III, and Claudia M. Meyer
NYMA, Inc., Cleveland, Ohio

National Aeronautics and
Space Administration

Lewis Research Center

October 1998

Available from

NASA Center for Aerospace Information
7121 Standard Drive
Hanover, MD 21076
Price Code: A03

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22100
Price Code: A03

Real-Time Sensor Validation, Signal Reconstruction, and Feature Detection for an RLV Propulsion Testbed

Amy L. Jankovsky
NASA Lewis Research Center
Cleveland, Ohio

Christopher E. Fulton
Analex Corporation
Cleveland, Ohio

Michael P. Binder
William A. Maul III
Claudia M. Meyer
NYMA Corporation
Cleveland, Ohio

Abstract

A real-time system for validating sensor health has been developed in support of the reusable launch vehicle program. This system was designed for use in a propulsion testbed as part of an overall effort to improve the safety, diagnostic capability, and cost of operation of the testbed. The sensor validation system was designed and developed at the NASA Lewis Research Center and integrated into a propulsion checkout and control system as part of an industry-NASA partnership, led by Rockwell International for the Marshall Space Flight Center. The system includes modules for sensor validation, signal reconstruction, and feature detection and was designed to maximize portability to other applications. Review of test data from initial integration testing verified real-time operation and showed the system to perform correctly on both hard and soft sensor failure test cases. This paper discusses the design of the sensor validation and supporting modules developed at LeRC and reviews results obtained from initial test cases.

Introduction

The Reusable Launch Vehicle (RLV) program is a cooperative effort involving the United States government and industry to achieve relatively inexpensive and reliable access to space. To attain these goals, innovative technologies are being developed and demonstrated. One such effort is the Integrated Propulsion Technology Demonstrator (IPTD) at the NASA Marshall Space Flight Center. The IPTD is a ground-based test facility developed by Rockwell International and NASA for the purpose of demonstrating and refining propulsion system technologies before they are tested in flight. One important goal of the RLV program is to improve operational efficiency by reducing the cost, time and number of personnel required to prepare a space vehicle for launch, including between-flight maintenance. Automated monitoring of the propulsion system and associated ground support facilities, as well as detection and diagnosis of anomalies before launch and during flight are all critical to improving operational efficiency. The portion of the IPTD program which integrates monitoring and diagnostics with control issues for the test stand as well as the test article is the Propulsion Checkout and Control System (PCCS).[1]

Real time determination of maintenance requirements is an important capability to achieve the fast turn around requirements of the RLV program. The PCCS was designed to demonstrate some technologies that will help achieve this goal. The overall PCCS system includes smart sensing techniques, model-based diagnostics, and automated control capabilities to operate the test article and provide maintenance and health information in real time. The entire PCCS automated checkout software package provides diagnostic results in real time within the operating cycle of one second on a Sun Sparc 20 computer platform.

Because of their high frequency of occurrence and potential hazards to successful operation of the test article, it is important to identify sensor faults and prevent erroneous information from being passed on to other software modules. Sensor validation for the PCCS has been developed by LeRC using a combination of limit checking, redundancy management, feature detection, and model-based reasoning to detect and isolate sensor failures. The feature detection algorithms also provide characteristics found in the data stream that are unrelated to sensor failures and to known system events, such as a scheduled valve operation, to the other diagnostic modules. Because the PCCS software was developed while the IPTD testbed was being designed and built, emphasis was placed on developing software that could be easily modified. In this manner, the fidelity and functionality of the LeRC modules can be easily increased as data and failure histories become available.

Description of Sensor Validation, Sensor Reconstruction and System Transient Detection Software Modules

The portion of the PCCS developed by LeRC is organized into three main modules: Sensor Validation, Sensor Reconstruction, and System Transient Detection. The Sensor Validation Module scans each critical sensor signal trace and detects significant features (such as level shifts, spikes, and drifts) in the data. By screening for hard failures and comparing detected transient features among related sensors, the Sensor Validation Module determines whether these features are caused by actual system conditions or are due to sensor failures. Detected features which are not attributable to a sensor failure are processed by the System Transient Detection Module, which screens out features due to normal system events (i.e. valve movement) and reports the remaining anomalous features to the diagnostic subsystem of the PCCS. The Sensor Reconstruction Module replaces failed sensor readings with synthesized values. The three modules are initiated before the start of a propulsion system test, and are called once during each one-second operating cycle. A sensor status array, valid data array and set of detected, validated features are made available to other PCCS diagnostic modules.

Sensor Validation Module

The Sensor Validation Module is designed to detect both hard and soft sensor failures. Figure 1 shows the overall data flow diagram for this module. During the initialization of the PCCS software, specific IPTD design information, a list containing the available sensor set, and lists of requested feature extraction calls for each sensor are loaded from user input files. Each time the module is then called, the sensor validation code first performs the sensor reasonableness checks on rate and magnitude limits to identify hard sensor failures. The software then calls the appropriate feature extraction routines, which are tailored to the current IPTD operating phase, and performs a redundant channel comparison. An expert system then reasons on the features found in the data to resolve any possible soft sensor failures found in the redundant channel check routine.

Limits used for the detection of all reasonableness exceedances and features are set by the user. These thresholds are dependent on sensor characteristics and system state and are therefore set specifically for each individual sensor and operating phase.

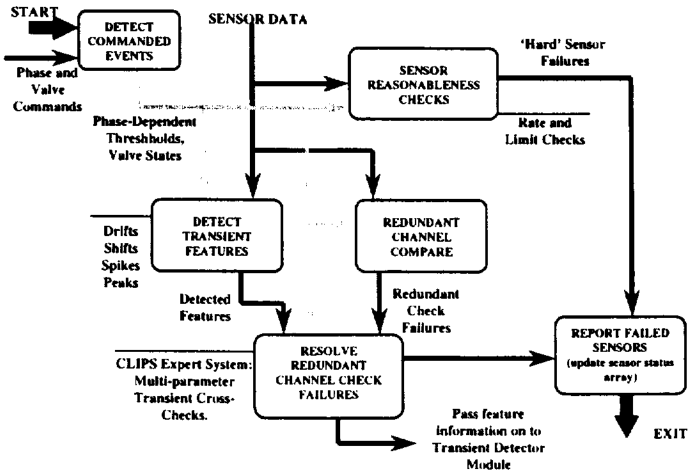

Figure 1. Sensor Validation Module.

The feature extraction and reasonableness check algorithms are based on routines developed for post-test analysis[2,3] and were modified to perform in real time on 25 hz data. Algorithms used to detect spikes, peaks, and limit and rate violations were combined into a single routine which searches through full sample-rate data for occurrences of each respective feature or limit violation. Violations that are found to occur within an operating cycle are reported immediately while longer duration violations, such as drifts, are reported when an end time is determined. To enable timely information update, long duration features that have surpassed a maximum number of cycles without exhibiting an end are reported with the current time as the end time, which is then updated with each cycle. Limit violations indicate unreasonable data magnitudes and are detected when a parameter exceeds an upper limit, or falls below a lower limit for a specified number of samples. Similarly, rate of change violations are identified when changes in the data occur faster than the maximum response rate of the sensor. Simple illustrations of a spike and a peak are shown in Figure 2. Spikes and peaks differ only in their width and are identified by monitoring the slope of the data. A simple dy/dt calculation is performed on successive data samples to find the slope. If a slope is found to exceed a threshold for a specified period of time and then return to within limits, it is considered a candidate spike or peak. These candidates are categorized by checking against predefined limits. The limits are set by the user to minimize the effects of noise while capturing spikes and peaks that are relevant to the particular system being tested.

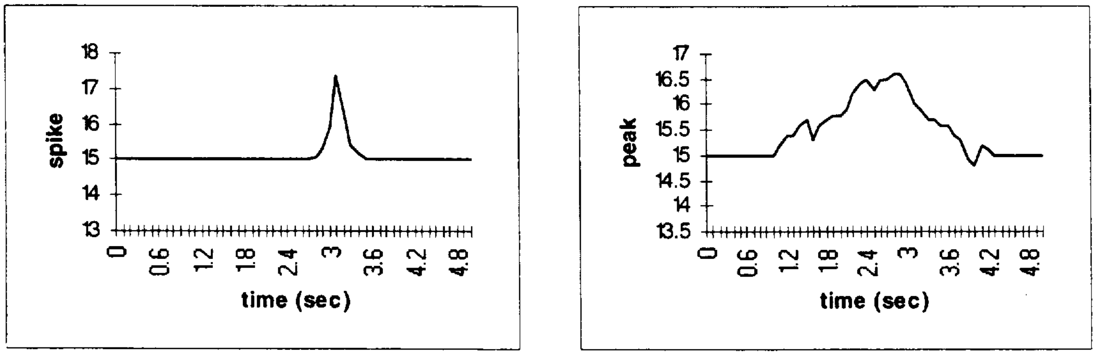

Figure 2. Examples of parameters exhibiting a spike and a peak.

Level shifts and drifts in the data are also detected and used as possible indicators of failed sensors. Examples of these two features are shown in Figure 3. In order to identify level shifts and drifts, a slope calculation is performed for each one-second averaged data point, centered about that point. If a slope value is found to exceed the threshold, a potential drift or level shift is flagged. The end time for the feature is then sought by determining when the slope returns to the nominal (flat) range. To minimize the effects of noise in determining end times, a small, preset number of slopes are permitted to dip below the threshold without declaring the end of a drift feature. This is especially useful in detecting long drifts in noisy signals. Once the end of the feature has been identified, a magnitude check is performed to ensure that the change in the parameter is large enough to warrant reporting. Level shifts and drifts can be distinguished from one another by examining the feature duration: level shifts occur over a shorter time window than drifts.

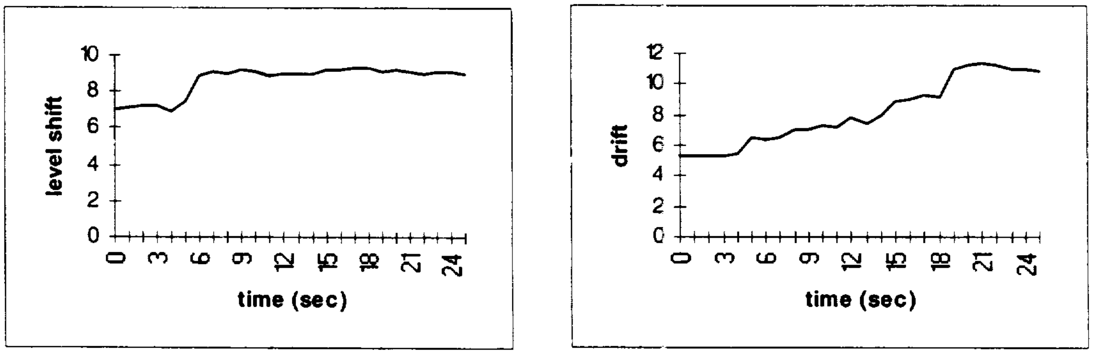

Figure 3. Examples of parameters exhibiting a level shift and a drift.

The Sensor Validation Module also includes an expert system which compares features detected in multiple sensors to determine if they represent actual system phenomena and should therefore be passed on to the system diagnostic modules (confirmed features) or should be attributed to instrumentation anomalies. This is accomplished by taking advantage of sensor hardware redundancy (multiple sensor channels at the same location) as well as analytical redundancy in the system. Analytical redundancy is implemented through the use of related parameter lists. The logic takes into account the state of the system in order to assess how parameters are related to one another. For example, when a valve is open, pressure sensor parameters on either side of the valve are considered to be related. When a valve is closed, relationships that cross the valve are removed from consideration. Hardware redundancy receives precedence over analytical redundancy in the logic. If two hardware redundant sensors exhibit the same feature, that feature is automatically confirmed. If only one in a pair of hardware redundant sensors exhibits a feature, related parameters are enlisted to corroborate or discount the feature in question. Confirmed features are passed on to other PCCS modules for further consideration. Unconfirmed features are indicative of instrumentation anomalies and are not passed on to other modules.

The logic performs many time management functions. It updates end times for drifts and for redundant channel violations, since these features may occur over several operating cycles. The code also monitors the time elapsed from when a feature is posted until it is confirmed. Efforts to confirm the feature are terminated if the time has exceeded a default maximum. Old information is continuously discarded in order to enhance real-time operation. In addition, the logic allows for the time it takes for an event to propagate through the system by fuzzifying the matching of times used when corroborating features.

Sensor Reconstruction Module

The functional diagram for the Sensor Reconstruction Module is shown in Figure 4. During the initialization phase of operation, the sensor reconstruction module loads a set of relations describing how each sensor parameter is related to other sensor parameters in equation form. These relationships may change from one system operating phase to the next. The code was designed to accommodate different operational phases where parameter relationships may be phase-dependent. As an example, three redundant pressure transducers P1, P2 and P3 would be represented by the following table entries:

Y VARIABLE	PHASE	RELATION	# X VARIABLES	X VARIABLES
P1	all	equals	2	P2, P3
P2	all	equals	2	P1, P3
P3	all	equals	2	P1, P2

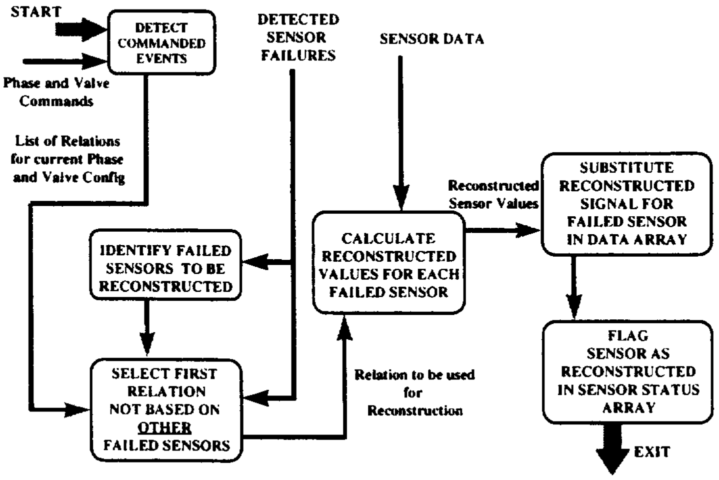

Figure 4. Sensor Reconstruction Module.

The Sensor Reconstruction Module is called after all sensors have been processed by the Sensor Validation Module. Each time the Sensor Reconstruction Module is called, a sensor status array is scanned for instances of a failed sensor found by the Sensor Validation Module. Once a failed sensor is found, the equations relating this sensor to other system sensors are consulted. Because the system is designed to handle multiple sensor failures, it is necessary to verify that the sensors used in these relations are valid themselves. If an equation is found where all related sensors are valid (neither failed nor reconstructed), then calculations are performed and the data array is updated to replace the failed sensor data. The sensor status array is also updated to indicate that the sensor has been reconstructed.

For the IPTD implementation, replacement of a failed sensor with a redundant sensor value was used as the sole method of reconstruction. This was due to the lack of empirical data or models which would have enabled the creation of more complex relationships among parameters. As data is obtained and higher fidelity models can be created to relate the sensors, the system can be updated simply by editing the user-defined tables.

System Transient Detection Module

The main purpose of the System Transient Detector Module is to filter through features which have been detected and confirmed by the Sensor Validation Module, and report only those features which are not attributable to normal system events. User-defined tables are loaded during the initialization phase to provide information on transient features that are expected during each valve action and operating phase. This information includes a maximum settling time for each valve event, a list of sensors effected by that event, and a list of features normally expected during each phase of operation for each sensor. This information is then used by the sensor timer functions to filter out features which can be attributed to normal system events and operating characteristics, as illustrated in Figure 5.

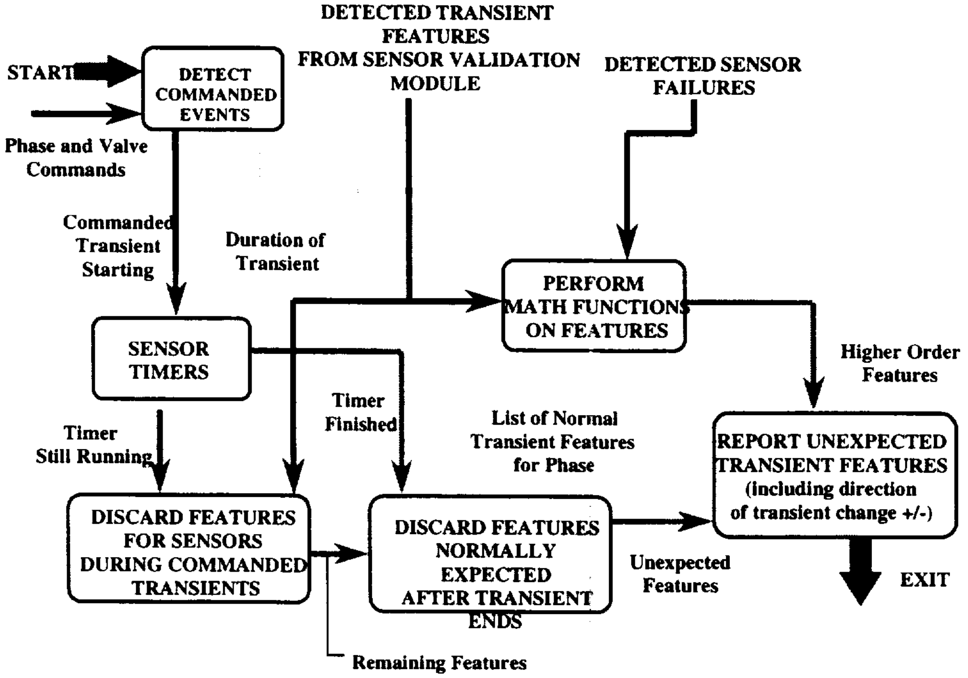

Figure 5. System Transient Detection Module.

During a commanded valve opening, for example, features found in parameters effected by the valve action are filtered out during the time period when the valve-initiated transients might be occurring. For the IPTD implementation, valves on the oxygen side affected all oxygen side sensors, and similarly all fuel side sensors were affected by any fuel side valve change.

Features indicative of expected behavior during a particular phase of system operation are also filtered out. As an example, during the chilldown phase, certain temperatures are expected to change throughout the entire process. Features, such as level shifts and drifts, found in these parameters during this phase are then filtered out. Any features not filtered out by the System Transient Detector are reported to other modules in the PCCS for further consideration as potential anomalies.

The System Transient Detector Module also performs simple mathematical operations on some of the remaining features to provide additional information to the remaining IPTD modules. Capabilities were built into the software to perform addition, subtraction, multiplication and division of like features. For the IPTD implementation, the subtraction operation was used to compute delta features between certain pressure parameters.

Case Studies

The PCCS was assembled and checkout testing was performed at the Marshall Space Flight Center. During this checkout testing, hard sensor failures were intentionally injected into the system in order to test the sensor validation and sensor reconstruction functions as well as the system diagnostic functions of the PCCS. These failures included physically disconnecting a line pressure transducer, E42P1024D, and electronically changing the gain on a feedline manifold pressure transducer, E42P1017D. A valve was also manipulated to give the system dynamic behavior, allowing for the checkout of the feature extraction functions. All PCCS modules were found to perform within the 1 second operating cycle, processing 128 sensor signals, sampled at 25 hz, during the checkout testing. No missed detections or false alarms were generated by the LeRC modules, while correct operation of the modules was verified for the cases tested.

Figure 6 shows E42P1024D, the pressure measurement that was disconnected to mimic an open condition, and the redundant sensor measurement E42P1025D. The sensor validation software found both a limit violation and a rate exceedance at 14.0 seconds and declared E42P1024D a failed sensor, replacing the values in the data stream with those of the redundant parameter, E42P1025D.

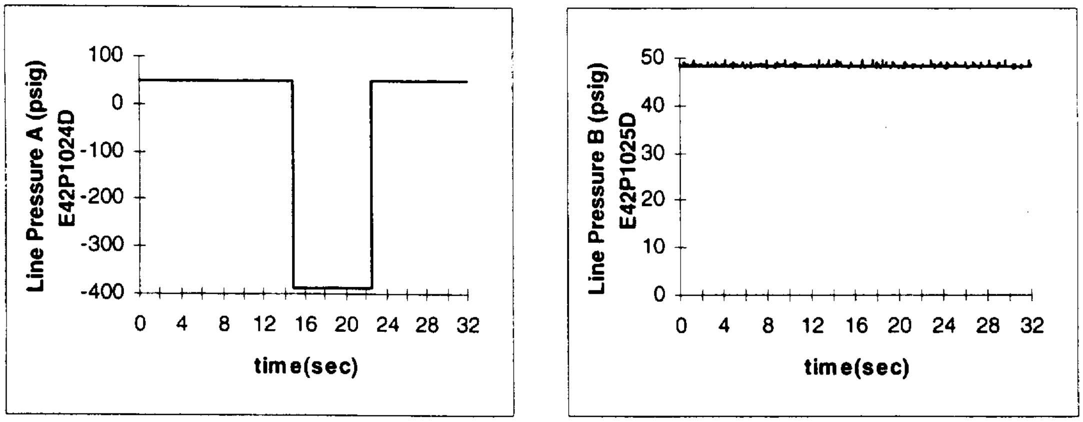

Figure 6. Open circuit test case showing failed sensor A and valid, redundant sensor B.

The redundant transducers whose signals are shown in Figure 7 were manipulated to have different gains. During a ramp in helium flow through the system, the two measurements deviate from one another. Although the software flags a redundant channel check violation, no sensor failure is declared. This is because there was not any corroborating evidence which could be used to determine which sensor was reading correctly. In the absence of evidence supporting either E42P1016D or E42P1017D to be the failed sensor of the pair, no sensor reconstruction was performed. However, because both redundant parameters were found to exhibit a drift in value that was not expected, a drift feature was confirmed and passed on to the other PCCS elements.

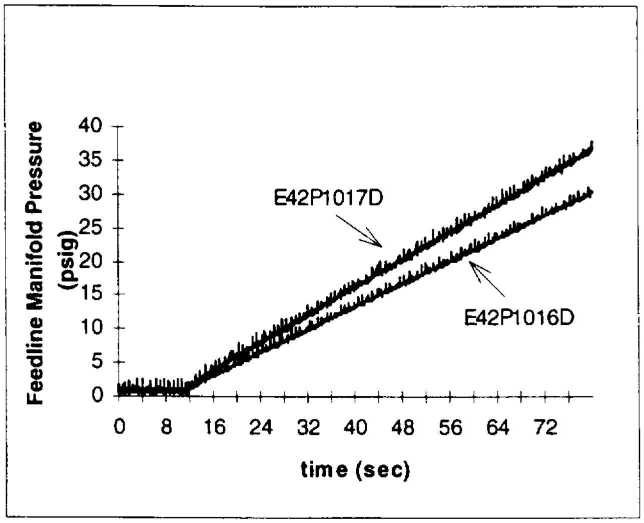

Figure 7. Drifts in redundant parameters test case.

In Figures 8 and 9, level shifts and drifts are depicted that were found by the feature extraction software during the cycling of a valve. For this test, helium was flowed through the LOX system, and a bleed valve was cycled. The opening and closing of the bleed valve were not provided in the tables as scheduled events in order to simulate an inadvertent valve opening and to test diagnostic functions of the PCCS. Because related parameters provided corroboration of features, no sensor failures were indicated. As no scheduled event was expected, features were not filtered out during the cycling of the valve, but were reported by the System Transient Detector. Figure 8 shows a typical profile of a pressure transducer located near the bleed valve. Each pressure rise and fall is characterized by a level shift followed by a drift downward. The temperature profile in Figure 9, the LOX feedline manifold temperature, is typical of the surrounding temperature traces and shows the associated drifts and level shifts found.

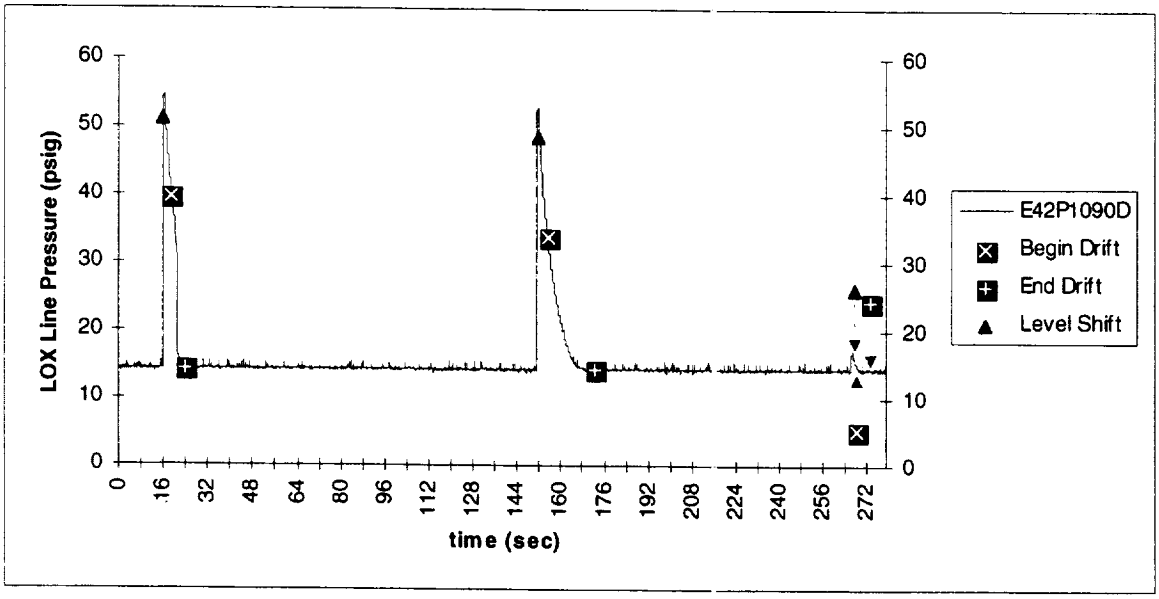

Figure 8. Example of level shifts and drifts found in pressure readings during valve cycling.

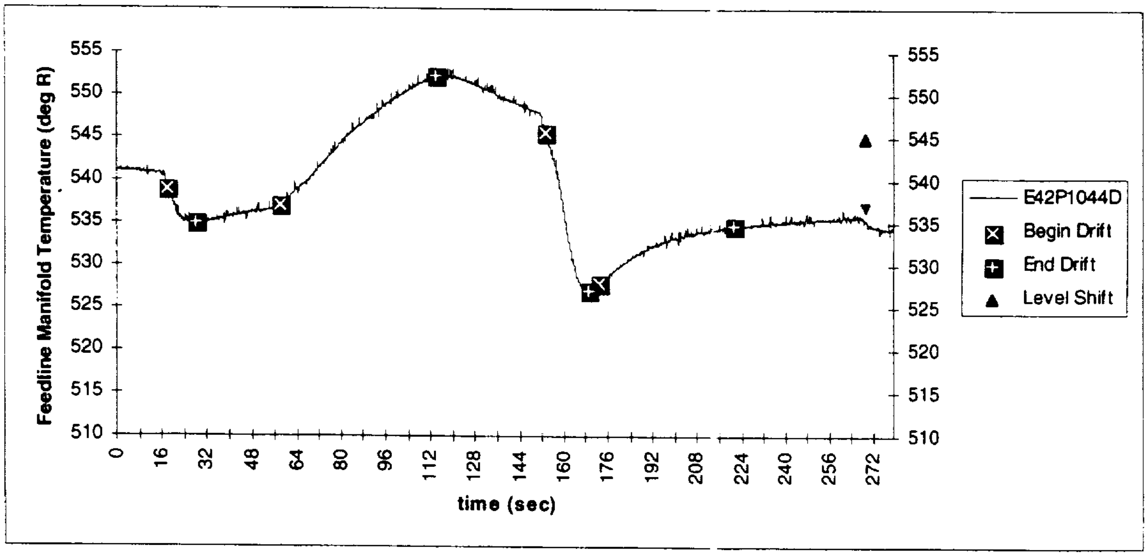

Figure 9. Example of level shifts and drifts found in temperature readings during valve cycling.

Summary

Software has been developed at the NASA Lewis Research Center to perform sensor validation, data reconstruction, and system transient detection for a propulsion system checkout and control system (PCCS) as part of the RLV program. The system uses analytical redundancy to validate sensor readings and to reconstruct failed sensors. This capability can be used by the PCCS to provide timely replacement of faulty sensors and to remove faulty signals from consideration by other control and monitoring software.

The LeRC supplied software processed 128 sensor signals and successfully achieved real-time operation for an operating cycle of one second on a Sun Sparc 20 platform. Because the software was developed concurrently with the IPTD hardware, it was designed without any prior data or failure history. Detection of failed sensors was accomplished using a qualitative model-based approach. All LeRC modules were designed to be easily modified as test data are obtained to increase the system's ability to detect and isolate sensor faults, identify system transient features and create replacement values for failed sensors.

Due to budget and time constraints, limited testing opportunities were available while the IPTD system was being checked out. During these sessions, the LeRC software correctly isolated sensors which had been disconnected, replacing the faulty data with that of an appropriate redundant parameter. In addition, the software correctly identified different ramp rates in redundant sensors and detected unscheduled valve activity.

References

1. Rogers, James P., Vesperman, Curtis L., "Integrated Propulsion Technology Demonstrator - Propulsion Module Design Review", N95-79091, April 17, 1995.

2. Zakrajsek, J. F Fulton, C.E., Meyer, C.M.,., "Feature Extraction for Post-Test Diagnostics", Advanced Earth-to-Orbit Propulsion Technology Conference 1994, May, 1994.

3. Erickson, T. ., Sue, J. J., Zakrajsek, J. F., Jankovsky, A. L., Fulton, C. E., Meyer, C. M., "Post-Test Feature Extraction Applied to Martin Marietta Atlas/Centaur Data", AIAA 94-3224, June, 1994.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)

2. REPORT DATE
October 1998

3. REPORT TYPE AND DATES COVERED
Technical Memorandum

4. TITLE AND SUBTITLE
Real-Time Sensor Validation, Signal Reconstruction, and Feature Detection for an RLV Propulsion Testbed

5. FUNDING NUMBERS
WU–242–33–01–00

6. AUTHOR(S)
Amy L. Jankovsky, Christopher E. Fulton, Michael P. Binder, William A. Maul III, and Claudia M. Meyer

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135–3191

8. PERFORMING ORGANIZATION REPORT NUMBER
E–11392

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, DC 20546–0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NASA TM—1998-208807

11. SUPPLEMENTARY NOTES

Amy L. Jankovsky, NASA Lewis Research Center; Christopher E. Fulton, Analex Corporation, Cleveland, Ohio; Michael P. Binder, William A. Maul III, and Claudia M. Meyer, NYMA, Inc., Cleveland, Ohio. Responsible person, Amy L. Jankovsky, organization code 5530, (216) 977–7498.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified - Unlimited
Subject Category: 15 Distribution: Nonstandard

This publication is available from the NASA Center for AeroSpace Information, (301 621–0390.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

A real-time system for validating sensor health has been developed in support of the reusable launch vehicle program. This system was designed for use in a propulsion testbed as part of an overall effort to improve the safety, diagnostic capability, and cost of operation of the testbed. The sensor validation system was designed and developed at the NASA Lewis Research Center and integrated into a propulsion checkout and control system as part of an industry-NASA partnership, led by Rockwell International for the Marshall Space Flight Center. The system includes modules for sensor validation, signal reconstruction, and feature detection and was designed to maximize portability to other applications. Review of test data from initial integration testing verified real-time operation and showed the system to perform correctly on both hard and soft sensor failure test cases. This paper discusses the design of the sensor validation and supporting modules developed at LeRC and reviews results obtained from initial test cases.

14. SUBJECT TERMS
Controllers; Sensors; Rocket test facilities, Diagnosis; Real time operation; Expert systems

15. NUMBER OF PAGES
15

16. PRICE CODE
A03

17. SECURITY CLASSIFICATION OF REPORT
Unclassified

18. SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

19. SECURITY CLASSIFICATION OF ABSTRACT
Unclassified

20. LIMITATION OF ABSTRACT

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102