DYNAMICS SENSOR VALIDATION FOR REUSABLE LAUNCH VEHICLE PROPULSION

J. P. Herzog
Y. Yue
Reactor Analysis Division
Argonne National Laboratory
9700 S. Cass Avenue
Argonne, IL 60439-4842
herzog@ra.anl.gov
Tele: (630) 252-3439 Fax: (630) 252-4978

and

R. L. Bickford
Expert Microsystems, Inc.
7932 Country Trail Drive #1
Orangevale, CA 95662-2120
rando@expmicrosys.com
Tele: (916) 989-2018 Fax: (916) 989-4277

The submitted manuscript has been created by the University of Chicago as Operator of Argonne National Laboratory ("Argonne") under Contract No. W-31-109-ENG-38 with the U.S. Department of Energy. The U.S. Government retains for itself, and others acting on its behalf, a paid-up, non-exclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.

Work supported by the U. S. Department of Energy, Nuclear Energy Programs as part of a CRADA, Contract No. C9600901, between Expert Microsystems and Argonne National Laboratory under Contract W-31-109-Eng-38 and National Aeronautics and Space Administration under Contract NAS8-40874
DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, make any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.
DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.
DYNAMICS SENSOR VALIDATION FOR REUSABLE LAUNCH VEHICLE PROPULSION

J. P. Herzog, Y. Y. Yue, R. L. Bickford

Argonne National Laboratory
Argonne, IL
Expert Microsystems, Inc.
Orangevale, CA

ABSTRACT

Expert Microsystems teamed with Argonne National Laboratory (ANL), a DOE contractor, to develop an innovative dynamics sensor validation system under a Small Business Technology Transfer (SBTTR) Phase I contract with NASA. The project improves launch vehicle mission safety and system dependability by enabling rapid development and cost effective maintenance of embeddable real-time software to reliably detect process-critical sensor failures. The project focused on verifying the feasibility of two innovative software methods developed by ANL to provide high fidelity sensor data validation for nuclear power generating stations, the Sequential Probability Ratio Test (SPRT) algorithm and the Multivariate State Estimation Technique (MSET) algorithm, as core elements of a commercial Dynamics Sensor Validation System (DSVS). The research verified that ANL algorithms enable highly reliable data validation for high frequency Space Shuttle Main Engine (SSME) dynamics sensors, such as accelerometers and strain gauges. Phase I culminated in production of a prototype runtime module which validates SSME flight accelerometer data with very high reliability. The resulting sensor validation development system is widely applicable to reusable launch vehicle (RLV) and ground support control and monitoring systems.

INTRODUCTION

Real-time sensor validation prevents a vehicle controller or facility safety system from making critical decisions, such as the decision to shut down an engine or abort a launch, on the basis of anomalous sensor data. Sensor data failures are defined as any failure in the data path which corrupts the sensor signal thereby providing erroneous information to the process control or monitoring system. The algorithms developed by ANL are effective at identifying and masking failures in sensors, cables, and data acquisition electronics.

Space Shuttle history illustrates the potential value of sensor validation. Numerous test aborts, launch scrubs, and launch delays have resulted from sensor failures. Sensor failures dominate the engine anomaly reports. Operations costs are adversely impacted with each test abort, launch scrub, and anomaly investigation. In one instance, sensor failures caused the premature in-flight shut down of an SSME during ascent. Flight STS-51F aborted to orbit after the engine no. 1 controller erroneously interpreted sequential failures of redundant high pressure fuel turbopump turbine discharge temperature sensors as an engine red-line condition.

The SSME control system includes a flight accelerometer safety cut-off system (FASCOS) for real-time safety monitoring of the engine accelerometers. The system is designed to generate a cut-off signal to the engine controller, but this capability has never been activated in flight. One reason the FASCOS is not used in flight is a prevalent concern that sensor data failures could cause a false engine shutdown. FASCOS logic is enabled for some ground tests where the risk associated with engine damage is higher than that for a false shutdown.

The primary goal of the Phase I effort was to demonstrate the suitability of the MSET and SPRT algorithms for real-time validation of dynamics sensor data with very high reliability. The MSET and SPRT algorithms were applied to SSME accelerometer data from actual Space Shuttle flights to conclusively validate the applicability and performance of these algorithms within this problem domain. When implemented in the SSME process control environment, the real-time capable DSVS software will greatly reduce the risk of false shutdown due to sensor failures.
The primary function of the DSVS is the automated production of embeddable sensor validation modules which detect the occurrence of sensor failures in a real-time process control environment. The feasibility of an integrated suite of tools to construct, tune, and produce sensor validation systems for high frequency dynamics data was demonstrated in Phase I of the project. The process begins with the user's definition of the sensors to be validated and the expected redundancies which exist between them. Statistical models of these redundancies are empirically derived from operations data using the MSET and SPRT tools to capture and quantify the relationships present in the signals. The MSET models enable a real time prediction of each sensor's output given the state of all other sensors in the system. These predicted values are compared to the observed sensor output and the difference between the signals, called the residual, is used to determine the validity of the observed sensor data. Statistical analysis of the residual values for sensor validation is accomplished using the SPRT algorithm.

This paper is focused on the MSET and SPRT algorithms, which are the main computational components of the DSVS. Each algorithm is described in detail. The results of calculations with archived sensor signals from a number of Space Shuttle flights are presented. The calculations demonstrate the applicability and performance of the MSET and SPRT algorithms for the validation of high frequency SSME signals.

DESCRIPTION OF THE ALGORITHMS

The MSET algorithm is a nonlinear state estimation technique that models a physical process through the analysis of a group of sensors that monitor the process. The system model generated by MSET relies upon an examination of the totality of information available from the array of sensors used to monitor the system and a comparison of these data as a whole to similar sets of data collected from the same system operated at various conditions in the past. Based upon this comparison of the current condition of the system with its past history, an optimal estimate of the current state of the system is obtained even if there are errors in the data currently collected, i.e., some of the sensors have malfunctioned.

The MSET system model identifies the current state of a system and estimates the expected values of the system parameters. The current system state is determined by comparing a vector of signals that are closely interrelated with the process variables under consideration with previously measured sets of the same signal vectors. The MSET algorithm uses advanced pattern recognition techniques to measure the similarity or overlap between signals within a defined operational domain. For each new observation of the system, it uses the patterns developed from the learned states to estimate the "true" state of the system. System states are represented by vectors whose elements are derived from sensor signals and can range from direct values of signals from the sensors to the result of any scalar transformation of the signals. Although the state vectors do not have to be linearly independent, they do have to represent the physical and/or chemical processes that are occurring and have some level of intercorrelation. The estimated state is calculated using a weighted combination of learned states, the weighting values being determined by the degree of pattern overlap with each learned state.

If data are collected from some process over a range of operating states, these data can be arranged in matrix form, where each column vector represents the measurements made at a particular state. Thus, this matrix will have the number of columns equal to the number of states (a total of m states) at which observations were made and the number of rows equal to the number of measurements (a total of n sensors) that were available at each observation. Defining the set of measurements taken at a given time t_j as an observation vector X(t_j),

\[ X(t_j) = [x_1(t_j), x_2(t_j), ..., x_n(t_j)]^T, \]

where \( x_i(t_j) \) is the measurement from sensor i at time t_j, then the data collection matrix can be defined as the process memory D:

\[ D = \begin{bmatrix}
    x_1(t_1) & x_1(t_2) & ... & x_1(t_m) \\
    x_2(t_1) & x_2(t_2) & ... & x_2(t_m) \\
    \vdots & \vdots & \ddots & \vdots \\
    x_n(t_1) & x_n(t_2) & ... & x_n(t_m)
\end{bmatrix}. \]  

Each of the column vectors in the process memory matrix represents an operating state of the system. Any number of observation vectors can be assigned to the process memory matrix.
Training of the MSET model consists of collecting enough observation vectors from historical operation of the system during normal conditions such that the process memory matrix encompasses the full dynamic operating range of the system. Training is a simple process which avoids time-consuming error minimization methods required to train neural networks. In MSET training, all that is necessary is the construction of a process memory matrix that represents the normal operating states of the modeled system. The MSET model is trained by selecting a subset of the data from the training period that spans all normal operating states of the system. MSET uses a two-step process to extract representative observation vectors from the training data. In the first step, MSET selects training data that correspond to the extrema of the normal operating states. For each sensor, the minimum and maximum sensor measurements from the training period data are found. The observation vectors containing these measurements are added to the process memory matrix. In the second step, the remaining observation vectors from the training period are ordered by their Euclidean norms. Evenly spaced elements from the ordered set are selected and the corresponding observation vectors are added to the process memory matrix.

Once the process memory matrix has been constructed, MSET is used to model the dynamic behavior of the system. At any point in time, an observation of the system (\(X_{\text{obs}}\)) is made. MSET compares the current observation vector to the stored operating states to calculate an estimate of the current system state. The estimate of the current state (\(X_{\text{est}}\)) is an n-element vector that is given by the product of the process memory matrix and a weight vector, \(W\):

\[
X_{\text{est}} = D \cdot W. \quad (3)
\]

The weight vector represents a measure of similarity between the estimate of the current state and the process memory matrix. To obtain the weight vector, the error vector \(\epsilon\) is minimized, where:

\[
\epsilon = X_{\text{obs}} - X_{\text{est}}. \quad (4)
\]

The error is minimized for a given state when:

\[
W = \left(D^T \otimes D\right)^{-1} \cdot (D^T \otimes X_{\text{obs}}). \quad (5)
\]

This equation represents a "least square" minimization when the operator \(\otimes\) is the matrix dot product. The MSET method uses advanced pattern recognition techniques to determine the non-linear operator \(\otimes\) that fits the input data vector. Once the weight vector is found, the resulting estimate of the current state of the system is given by

\[
X_{\text{est}} = D \cdot \left(D^T \otimes D\right)^{-1} \cdot (D^T \otimes X_{\text{obs}}). \quad (6)
\]

The SPRT algorithm uses MSET predictions to validate sensor signals. The SPRT technique is based on user-specified false alarm and missed alarm probabilities, allowing control over the likelihood of false or missed detection. The SPRT algorithm, as employed by the DSVS, consists of a set of four statistical hypothesis tests to detect deviations in normal behavior of residual signals, thereby providing detection of sensor faults at the earliest possible stage of development. A residual signal is the difference between the measurements made by a sensor and the values predicted by MSET for that sensor. Let \(r_n\) represent the difference between a sensor measurement and a MSET estimate at a given moment \(t_n\) in time. Then the residual signal is represented by the sequence of values \(\{r_n\} = r_0, r_1, ... r_n\).

Each SPRT is a binary hypothesis test. A SPRT analyzes a residual signal to determine whether it is consistent with normal behavior or with some specific abnormal behavior. When a SPRT reaches a decision about current residual signal behavior, either that the signal is behaving normally or abnormally, the decision is reported and the test continues analyzing the data from the signal. There is no other procedure with lower error probability or shorter average sampling time than decision tests based on the SPRT, provided the signal noise is Gaussian.

The SPRT algorithm is especially useful for analog signals where noisy data are present or slight changes in mean signal value may precede catastrophic failure. The SPRT technique provides a superior prognostic tool because it is sensitive not only to disturbances in signal mean, but also to very subtle changes in the statistical quality (variance, skewness, bias) of the signals. For sudden, gross failures of a signal source, the SPRT will announce the disturbance as fast as a conventional threshold limit check. However, for slow degradation, the SPRT can detect the incipience or onset of the disturbance long before it would be apparent with conventional threshold limit checks.
For any SPRT, normal signal behavior is defined to be that the signal data adheres to a Gaussian probability density function (pdf) with mean 0 and variance $\sigma^2$. Normal signal behavior is referred to as the null hypothesis, $H_0$. Each of the four tests determines whether current signal behavior is consistent with the null hypothesis or one of four alternative hypotheses. The four tests are known as the positive mean test, the negative mean test, the nominal variance test, and the inverse variance test. For the positive mean test, the corresponding alternative hypothesis, $H_1$, is that the signal data adhere to a Gaussian pdf with mean +M and variance $\sigma^2$. For the negative mean test, the corresponding alternative hypothesis, $H_2$, is that the signal data adhere to a Gaussian pdf with mean -M and variance $\sigma^2$. For the nominal variance test, the corresponding alternative hypothesis, $H_3$, is that the signal data adhere to a Gaussian pdf with mean 0 and variance $\sigma^2/\nu$. For the inverse variance test, the corresponding alternative hypothesis, $H_4$, is that the signal data adhere to a Gaussian pdf with mean 0 and variance $\sigma^2/\nu$. 

The SPRT technique operates as follows. At each timestep in a calculation, a test index is calculated and compared to two threshold limits A and B (defined below). The test index is equal to the natural logarithm of the likelihood ratio $L$, which for a given SPRT is the ratio of the probability that the alternative hypothesis for the test ($H_j$, where $j$ is the appropriate subscript for the SPRT in question) is true to the probability that the null hypothesis ($H_0$) is true:

$$L_n = \frac{\text{Probability of } \{R_n\} \text{ given } H_j \text{ true}}{\text{Probability of } \{R_n\} \text{ given } H_0 \text{ true}}.$$  (7)

If the logarithm of the likelihood ratio is greater than or equal to the logarithm of the upper threshold limit (i.e., $\ln(L_n) \geq \ln(B)$), then it can be concluded that the alternative hypothesis is true. If the logarithm of the likelihood ratio is less than or equal to the logarithm of the lower threshold limit (i.e., $\ln(L_n) \leq \ln(A)$), then it can be concluded that the null hypothesis is true. If the logarithm of the likelihood ratio falls between the two limits (i.e., $\ln(A) < \ln(L_n) < \ln(B)$), then neither hypothesis can be concluded to be true and sampling continues. The threshold limits are related to the misidentification probabilities ($\alpha$ and $\beta$) by the following expressions:

$$A = \frac{\beta}{1 - \alpha} \quad \text{and} \quad B = \frac{1 - \beta}{\alpha},$$  (8)

where

- $\alpha$ is the probability of accepting $H_j$ when $H_0$ is true (i.e., the false alarm probability)
- $\beta$ is the probability of accepting $H_0$ when $H_j$ is true (i.e., the missed alarm probability).

The first two SPRTs for normal distributions examine the mean of the residual signal. The goal of the mean tests is to declare that the system is degraded if the residual signal exhibits a nonzero mean, e.g., a mean of either $+M$ or $-M$, where $M$ is the preassigned system disturbance magnitude for the mean test. Assuming that the difference sequence $\{R_n\}$ adheres to a Gaussian pdf, then the likelihood ratio $L_n$ for the positive mean test can be shown to be:

$$L_n = \exp \left[ \frac{-1}{2\sigma^2} \sum_{k=1}^{n} M \left( M - 2r_k \right) \right].$$  (9)

The SPRT index for the positive mean test ($\text{SPRT}_{p\text{me}}$) is given by taking the logarithm of the likelihood ratio:

$$\text{SPRT}_{p\text{me}} = \frac{M}{\sigma^2} \sum_{k=1}^{n} \left( r_k - \frac{M}{2} \right).$$  (10)

The SPRT index for the negative mean test ($\text{SPRT}_{n\text{me}}$) can be derived by substituting $-M$ for each instance of $M$ in Equation (10),

$$\text{SPRT}_{n\text{me}} = \frac{M}{\sigma^2} \sum_{k=1}^{n} \left( -r_k - \frac{M}{2} \right).$$  (11)

The other two SPRTs for normal distributions examine the variance of the sequence. In the variance tests, the system is declared to be degraded if the sequence exhibits a change in variance by a factor of $V$ or $1/V$, where $V$, the preassigned system disturbance magnitude for the variance test, is a positive number. Assuming that the difference sequence $\{R_n\}$ adheres to a Gaussian pdf, then the likelihood ratio for the variance test can be shown to be:
\[ L_n = V^{-n/2} \exp \left[ -\frac{1}{2\sigma^2} \left( \frac{V - 1}{V} \right) \sum_{k=1}^{n} r_k^2 \right] \] (12)

The SPRT index for the nominal variance test (\( \text{SPRT}_{\text{nom}} \)) is given by taking the logarithm of the likelihood ratio

\[ \text{SPRT}_{\text{nom}} = \frac{1}{2\sigma^2} \left( \frac{V - 1}{V} \right) \sum_{k=1}^{n} r_k^2 - \frac{n}{2} \ln V. \] (13)

The SPRT index for the inverse variance test (\( \text{SPRT}_{\text{inv}} \)) can be derived by substituting \( 1/V \) for each instance of \( V \) in Equation (13),

\[ \text{SPRT}_{\text{inv}} = \frac{1}{2\sigma^2} \left( 1 - \frac{1}{V} \right) \sum_{k=1}^{n} r_k^2 + \frac{n}{2} \ln V. \] (14)

The SPRT algorithm performs both mean and variance tests on a residual signal. To initialize the module for analysis of a residual signal, the user specifies the system disturbance magnitudes for the tests (\( M \) and \( V \)), the false alarm probability (\( \alpha \)), and the missed alarm probability (\( \beta \)). The system disturbance magnitude for the mean tests specifies the number of standard deviations the distribution must shift in the positive or negative direction to trigger an alarm. The system disturbance magnitude for the variance tests specifies the fractional change of the variance necessary to trigger an alarm.

At the beginning of the monitoring phase, all four SPRT indices are set to 0. Then during each timestep of the calculation, the SPRT indices are updated using Equations (10), (11), (13), and (14). Each SPRT index is then compared to the upper \([ \ln((1-\beta)/\alpha)]\) and lower \([ \ln(\beta/(1-\alpha)]\) threshold limits, with these three possible outcomes:

1) the lower limit is reached, in which case the signal is declared healthy, the test statistic is reset to zero, and sampling continues;
2) the upper limit is reached, in which case the signal is declared degraded, an alarm flag is raised indicating a fault, the test statistic is reset to zero, and sampling continues; or,
3) neither limit has been reached, in which case no decision concerning the signal can yet be made and the sampling continues.

RESULTS OF SSME CALCULATIONS

Each main engine of the Space Shuttle contains two turbopumps, which supply fuel and oxygen to the main combustion chamber. Each turbopump assembly is monitored by six high frequency (10.24 kHz) sensors. The sensors consist of three accelerometers and two strain gauges on each pump body plus two accelerometers on the engine gimbal bearing. Thus, each SSME is monitored by a total of twelve high frequency sensors. Because of its high frequency, each sensor generates an enormous amount of data during a flight. For a typical signal duration of 542 sec, the high frequency sensors on just one of the Space Shuttle's three main engines produce about 540 MBytes of data.

Several models were prepared to investigate the accuracy of the MSET simulations of the SSME high frequency sensors. The models contain varying numbers of sensor signals. The main model tested, the 6 sensor model, uses signals from the accelerometers that monitor the two turbopumps. Additional models include the 8 sensor model that uses signals from the six turbopump and two gimbal bearing accelerometers, the 12 sensor model that uses signals included in the 8 sensor model plus the four strain gauges, and the 26 sensor model that uses signals included in the 12 sensor model plus fourteen engine performance sensors. The engine performance sensors are low frequency (25 Hz) sensors that monitor flowrate, temperature, pressure, etc. of various engine components. The 6 sensor model was the most thoroughly tested model, because the signals included in the model presented the smallest computational burden. The results presented here pertain only to calculations using the 6 sensor model.

In an initial group of calculations, the signals from a single SSME recorded during a typical Space Shuttle flight (i.e., flight STS-58, engine 1) were analyzed with the 6 sensor model. The model was trained with the full SSME data set. Because the accuracy of the MSET estimates is a function of the size of the process memory matrix, a series of five process memory matrices of various sizes were extracted from the data set to simulate the response of the engine. The process memory matrices ranged in size from 100 observation vectors to 487 observation vectors. Since the data set contains about 5.5 \( 10^6 \) data points (i.e., a signal duration of 542.4 sec at a frequency of 10.24 kHz), the smallest process memory matrix contained only 0.0018% of the
data in the signals, while the largest contained 0.0083% of the data in the signals. It is important to note that the extremely compact training matrices created by MSET's training algorithm demonstrate at once both the power and the simplicity of the MSET pattern recognition tool, as contrasted to other approaches such as neural network and Kalman filter methods.

An MSET simulation of the flight STS-58, engine 1 signals was executed for each of the five process memory matrices. The MSET pattern recognition tool produces extremely accurate simulations of turbopump accelerometer behavior. In Figure 1, the relative errors of the six signals are averaged for each MSET calculation, and plotted as a function of the number of observation vectors in the process memory matrix. The figure shows that the accuracy of the model improves as the size of the process memory matrix increases. The relative error for each sensor signal is measured by the ratio of the standard deviation of the residual signal to the standard deviation of the sensor signal. Even for modest memory requirements, namely, 487 observation vectors in the process memory matrix, the error in the MSET estimates is on average a tiny fraction (0.9%) of the natural variation in the sensor signal.

In Figure 2, the results for the simulation of one of the accelerometers are shown in detail. In the top plot, the highly dynamic signal from one of the SSME turbopump accelerometers is shown as a function of time from the launch of the Space Shuttle. In the middle plot, MSET's estimate for the calculation trained with the largest process memory matrix is shown. Even though the sensor signal is wildly erratic, MSET simulates the accelerometer signal with very high accuracy. The residual signal for the calculation is shown in the bottom plot. The relative error for the signal is only 0.86%. The results shown in Figures 1 and 2 illustrate MSET's ability to accurately model highly dynamic signals, even when only a tiny fraction of available data are used to represent the sensor behavior.

MSET calculations with the signals from flight STS-58 were repeated with a filtered version of the data set. A simple extraction filter removed 99% of the data in the signals, passing every one-hundredth observation vector only. The results of the calculations with the filtered data set were comparable to the results with the full data set. Although the amount of information in the full data set is 100 times greater than

![Figure 1. Accuracy of the MSET Model as a Function of Process Memory Matrix Size](image-url)
Figure 2. MSET Analysis of the Signal from a Typical SSME Accelerometer
The errors in the simulated signals vary from 0.51% to signal standard deviations from flight to flight. The greater variability in the relative error rate is due to the variability in sensor the full range of accelerometer signals used for training.

The standard deviations of the residual signals indicates that the MSET calculations are consistently accurate across the maximum value of 0.0987 g. The slight variation in the signal signature from flight to flight, the MSET model was not trained with the signal from the failed accelerometer. Because the model was not trained with the signal from the failed sensor, MSET yields estimates that do not correspond to the behavior of the failed sensor. The process memory matrix used to train the MSET model defines a closed vector space. All estimates produced by the MSET algorithm are confined to the vector space upon which it was trained. Since the extreme data reported by the failed sensor lie outside of the vector space defined by the process memory matrix, MSET's estimates differ significantly from the data reported by the failed sensor. Therefore, the residual signal (i.e., the error in the MSET estimate signal) of the failed accelerometer is substantially larger than the residual signal of a normal accelerometer, as can be seen by comparing the residual signal in Figure 3 to that from Figure 2.

The most frequent failure mode for the SSME accelerometers is the failure of the insulation on the accelerometer cables. Cable failures result in a significant change in the mean and variance of the signal, which are easily detected by the MSET and SPRT algorithms. The residual signals from the MSET calculation of the flight STS-50, engine 2 data set were analyzed with the SPRT algorithm. The SPRT detected the sensor failure at about 3 seconds after the launch, which is essentially the same time that the failure becomes noticeable in the raw sensor signal.

To examine the ability of the model to simulate flight conditions with which it was not trained, MSET calculations were performed for each of the 10 testing data sets. The results of the testing group calculations are consistent with the results from the training group calculations, except for the test of the data set from flight STS-50, engine 3. The residual signal for the channel B2 accelerometer on the oxidizer preburner pump has a standard deviation (37.7 g) that is at least 350 times larger than typical values from the training data set calculations. Post-flight examination of engine 2 revealed that the electronic cable to the channel B2 accelerometer had failed during the flight. In Figure 3, the results from the MSET calculation for the failed sensor are shown. In the top plot, the signal from the failed SSME turbopump accelerometer is shown. The signal fluctuates between two extreme values that are much larger than the limits reached by a typical accelerometer. In the middle plot, MSET's estimate of the accelerometer signal is shown. MSET produces estimates that are consistently small relative to the signal from the failed accelerometer. Because the model was not trained with the signal from the failed sensor, MSET yields estimates that do not correspond to the behavior of the failed sensor. The process memory matrix used to train the MSET model defines a closed vector space. All estimates produced by the MSET algorithm are confined to the vector space upon which it was trained. Since the extreme data reported by the failed sensor lie outside of the vector space defined by the process memory matrix, MSET's estimates differ significantly from the data reported by the failed sensor. Therefore, the residual signal (i.e., the error in the MSET estimate signal) of the failed accelerometer is substantially larger than the residual signal of a normal accelerometer, as can be seen by comparing the residual signal in Figure 3 to that from Figure 2.

The most frequent failure mode for the SSME accelerometers is the failure of the insulation on the accelerometer cables. Cable failures result in a significant change in the mean and variance of the signal, which are easily detected by the MSET and SPRT algorithms. The residual signals from the MSET calculation of the flight STS-50, engine 2 data set were analyzed with the SPRT algorithm. The SPRT detected the sensor failure at about 3 seconds after the launch, which is essentially the same time that the failure becomes noticeable in the raw sensor signal.
Figure 3. MSET Analysis of the Signal from a Failed SSME Accelerometer
Subtle accelerometer failures were simulated by applying a small drift to the mean of a typical accelerometer signal. A disturbance was applied to the mean of one of the accelerometer signals in the flight STS-58 engine 1 data set. The other signals in the data set were not altered. The disturbance was applied 100 seconds into the flight. The mean of the signal was increased at a rate of 2% of the signal standard deviation per second. As shown in the top plot of Figure 4, the mean of disturbed signal increases by about 12 g for every minute that the disturbance is applied. The residual signal for an MSET calculation of the failed sensor is shown in the bottom plot. The residual signal was generated by the MSET model trained with the process memory matrix that contained 487 observation vectors. The disturbance begins to appear in the residual signal about 300 seconds into the flight, which is 200 seconds after the initiation of the disturbance. The residual signal for the failed sensor was analyzed with the SPRT algorithm at varying values of the false alarm probability and the system disturbance magnitude (M). As shown in Figure 5, the time at which the SPRT algorithm detects the subtle drift failure of the sensor is a weak function of both parameters.

The SPRT technique can detect a subtle sensor failure long before simple threshold limits and thus provides a highly effective prognostic technique. At the same time, the SPRT technique can detect a dramatic failure, such as a cable failure, as rapidly as conventional threshold limits based on redundant signals and may thus be used to reduce the system requirements for physical redundancy. The SPRT algorithm is constructed such that the user may specify the false alarm and missed detection rates. Very low inherent false alarm rates, on the order of 1 error in $10^5$ controller decision cycles were obtained during the analyses of the SSME accelerometer signals.

![Failed Sensor Signal](image1)

![Residual Signal for Failed Sensor](image2)

**Figure 4. MSET Analysis of a Subtle Sensor Failure**
CONCLUSIONS

The algorithms tested in the STTR Phase I project demonstrated the ability of the core elements of the DSVS to accurately simulate high frequency SSME sensor behavior and provide rapid detection of dramatic and subtle sensor failures. Tested with a large set of accelerometer signals from numerous Space Shuttle flights, the MSET model exhibited excellent simulation capabilities, with error rates that are 1% to 2% of the observed signals. The analysis of the flight STS-50, engine 2 data set revealed the ability of MSET and SPRT to rapidly detect a sensor failure due to the most common failure mode of SSME accelerometers. By projecting a slow drift onto the mean of a sensor signal, the ability of MSET and SPRT to detect a subtle sensor failure was demonstrated.

The results of Phase I validation testing conclusively demonstrated the feasibility of the Dynamics Sensor Validation System innovations and approach. With the recent award of STTR Phase II funding, development of the DSVS will continue. The main goal of the Phase II effort will be to develop the tools that find the optimum balance between run-time module performance, parameter estimation accuracy, sensor validation error rates, and sensor validation system robustness to multiple sensor failures.

REFERENCES

