

Gyro-based maximum-likelihood thruster fault detection and identification¹

Edward Wilson², Christopher Lages², and Robert Mah³
Smart Systems Research Laboratory, NASA Ames Research Center, MS 269-1, Moffett Field, CA 94035
6 September, 2001

Abstract

When building smaller, less expensive spacecraft, there is a need for intelligent fault tolerance vs. increased hardware redundancy. If fault tolerance can be achieved using existing essential navigation sensors, cost and vehicle complexity can be reduced. A maximum-likelihood-based approach to thruster fault detection and identification (FDI) for spacecraft is developed here and applied in simulation to the X-38 space vehicle. The system uses gyro signals to detect and identify hard, abrupt single- and multiple-jet on- and off-failures. Faults are detected within one second and identified within one to five seconds.

1. Introduction

The FDI system presented here was developed through application to two specific thruster-controlled spacecraft presently under development at NASA Johnson Space Center: the X-38 [11] and the miniAERCam. Its application to the X-38 is presented in this paper.

The Crew Return Vehicle (CRV) consists of a manned space vehicle, the Entry Vehicle (EV), based on a lifting body design, and a De-orbit Propulsion Stage (DPS). The CRV is designed to remain docked to the space station in a dormant mode for several years until needed by the crew in an emergency. The X-38 (vehicle 201) is the unmanned test vehicle for the CRV. Both vehicles are designed to maneuver on-orbit, de-orbit, and land using a large parafoil.

The DPS includes a set of axial and RCS thrusters fed by three mono-propellant hydrazine tanks. Although the CRV will have pressure sensors in the thrusters to detect failures, the X-38 has only temperature sensors. In this research, a fault detection and identification system is developed that uses only gyro signals (angular rates) to detect and identify single- and multiple-jet hard-on or hard-off thruster failures.

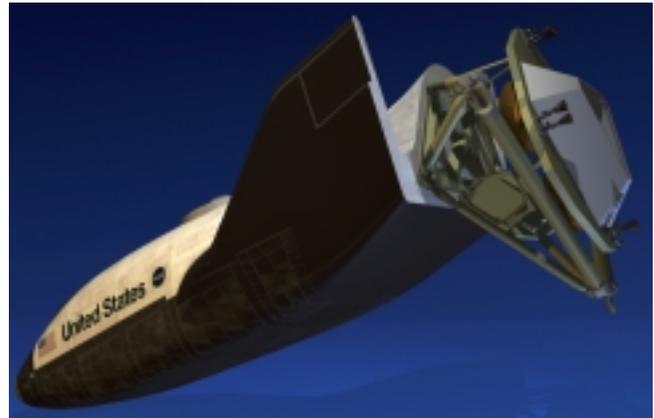


Figure 1: x38, with entry vehicle and de-orbit propulsion stage [7]

Related research

Deyst and Deckert [2] developed a maximum-likelihood based approach for detecting leaking thrusters for the Space Shuttle orbiter's RCS jets. The method for detecting soft failures was also extended to detect hard RCS jet failures. The maximum-likelihood method presented in that work is used and extended in this research.

Wilson and Rock [9] [10] developed a fault detection and identification method based on exponentially weighted recursive least squares estimation using accelerometer and angular rate sensors. A neural network then provided adaptive control reconfiguration to multiple destabilizing hard and soft thruster failures. This was applied to a 3-degree-of-freedom air-bearing vehicle.

2. Problem definition

Hard, abrupt, thruster failures resulting from a single point of failure (in valves, plumbing, electronics, etc.) are monitored. These can include single- or simultaneous multiple-jet failures in either a failed-on or failed-off condition. The DPS has 8 axial thrusters (500 Newtons thrust level each) that fire along the longitudinal axis of the vehicle, providing the required de-orbit thrust for the 13,600 kg vehicle. During the 8-to-15-minute de-orbit burn, six of the eight thrusters fire continuously, controlled open loop, with the six chosen symmetrically to produce minimal torque on the vehicle. The DPS also contains 8 reaction control

¹ This research was funded by NASA Headquarters, HQ AA, PWC 349-00.

² Drs. Wilson and Lages are with Intellization, ed.wilson and chris.lages@intellization.com

³ Dr. Mah is with NASA Ames Research Center, rmah@mail.arc.nasa.gov

system (RCS) thrusters (106 Newtons thrust level each) that are fired in sets of two or four by the attitude control system to control the roll, pitch, and yaw about the body axes. The entry vehicle (EV) has a completely separate set of RCS thrusters for use after DPS separation — those are not considered here. Figure 2 is a rear-view schematic of the X-38 showing EV RCS, DPS RCS, and DPS axial thrusters. The axial thrusters fire directly back along the x-axis, and the DPS RCS thrusters fire in the y-z plane, with no x-axis component.

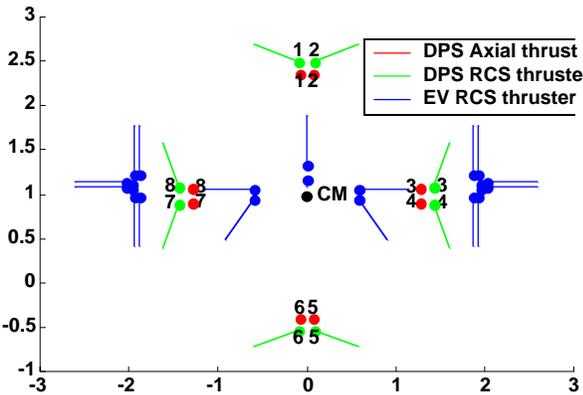


Figure 2: X-38 thruster configuration, coordinates in vehicle structural frame

Accelerometers and ring-laser gyros in the Honeywell Space Integrated GPS / INS (SIGI) [8] are available for monitoring vehicle motions. Temperature sensors in the thrusters provide failure information as well, but the response time limits the ability to detect failures quickly — they rise in about one second, but cool over a period of minutes. At this point, the Fault Detection and Identification (FDI) system has been developed without using temperature information, but it could be added at a later point. Thruster faults will be detected by comparing vehicle motions (at this point, only gyros are used, but accelerometers would enable even more accurate FDI) to the vehicle motions that would result if certain failures have occurred.

Equations of motion

Starting with Euler's dynamical equation, and assuming the spacecraft inertia matrix is constant, the equations of motion (EOM) are [1]

$$\dot{\omega} = I^{-1}(\tau - \tilde{\omega}I\omega)$$

or

where I is the spacecraft inertia matrix, ω is the angular velocity of the body-fixed frame with respect

to an inertial reference frame, and τ is the sum of all torques on the body. $\tilde{\omega}$ represents a matrix-multiply implementation of the cross product.

Simulation

Several unknown random variations are added to this dynamic model, including (values given are the 3-sigma value of a Gaussian distribution about the true or nominal value): gyro noise of 0.03 degrees/second; pulse-to-pulse thruster strength variability of 15%; constant thruster strength bias of 5%; inertia matrix elements constant bias of 5%; constant mass bias of 1%; and center of mass (CM) location offset of 5 mm along x- and y-axes and 25 mm along the z-axis. These values are all conservative estimates (i.e., at least as large as the actual) based on the actual X-38 design.

A dynamic simulation was developed using MATLAB [6]. As with the X-38 design, the control loop runs at 10 Hz, and unfiltered gyro data is read at 50 Hz. The FDI runs at 10 Hz. A controller that regulates to a commanded attitude calculates the thruster commands, the EOM from above, including the random variations, are integrated, the FDI system detects and identifies failures, and a MATLAB-based visualization displays the vehicle status and FDI results as shown in Figure 4.

3. Fault detection and identification

There are several FDI approaches reported in the literature [4], all of which perform well on certain applications. The on-off nature of the thrusters present in the class of applications addressed here limits the viability of many general methods. For example, if a thruster has failed off, but is not commanded to fire, it appears to be working correctly.

It is generally true in identification or detection systems that reducing the degrees of freedom to be considered or otherwise constraining the problem will improve identification or detection performance. As will be discussed at the end of this section, some alternative approaches were initially used to solve this problem that attempted to find the strengths of the unfailed thrusters as well as finding the failures. This approach worked well on a simplified version of the problem, but failed when all 16 thrusters were present, both on and off failures were considered, mass properties were allowed to vary within tolerance, and in the presence of gyro noise. This led to the approach described below, which solves the problem

taking full advantage of the problem statement — namely that only a single failure mode can be present, and that it will appear abruptly.

3.1 Summary of the algorithm

At every control update, the measured (estimated) angular acceleration is compared to that expected by a dynamic model of the nominal system. This vector is compared with the vector of angular accelerations corresponding to each possible failure mode. When a clear match is found (the likelihood is sufficiently higher than all other possibilities), a failure is declared. Specifics regarding filtering and other calculations follow.

3.2 Cataloging failure modes

For every possible failure mode, the unexpected acceleration, $\alpha_{unexpected}$, that would result if the mode were present and active is calculated. An off failure is said to be active if the thruster is commanded to fire; an on failure is active if it is not commanded to fire. Multiple-jet failure modes require cataloging of each combination of thrusters that may be active. For example, failure mode #1 corresponds to RCS jet 1 being failed off. The $\alpha_{unexpected}$ for this mode is [-0.0144, -0.0015, 0.0045] rad/sec² for the body roll pitch and yaw axes, meaning that if RCS jet 1 is commanded to fire, and it has failed off, the actual measured angular acceleration should exceed the expected acceleration by these values. This cataloging is done pre-flight, and the acceleration values are updated periodically based on the state of the blowdown (the strength of all thrusters drops as the tanks empty).

3.3 Estimating angular acceleration

The angular acceleration of the vehicle is calculated at each FDI update (10 Hz) based on the previous 6 gyro samples (sampled at 50 Hz, so 6 points cover one full control interval including both end points). Assuming small angular rates (so axes are decoupled), and that acceleration is constant during each control time period (corresponding to thruster firing times), the acceleration is estimated by fitting a line to the data and taking the slope as shown by the solid lines in Figure 4 below. This least-squares fit is implemented as a linear FIR filter, and is computationally efficient. An alternate approach fit three contiguous line segments to 16 points (covering 3 control intervals), and took the slope of the middle segment. This was marginally better, but was not implemented because of the extra computation and the need to wait one sample period for the information.

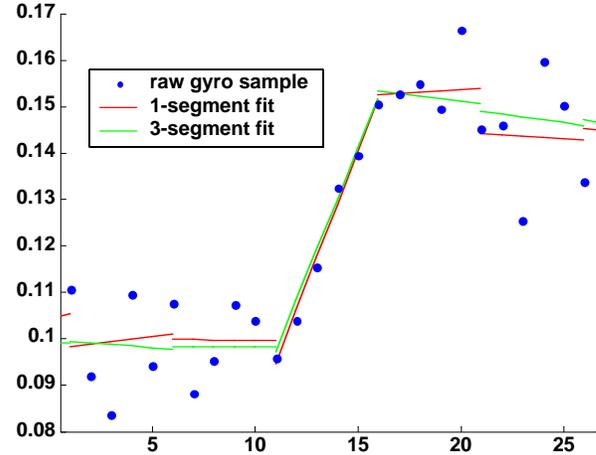


Figure 3: Estimation of angular acceleration

3.4 Calculating expected (nominal) acceleration

The expected acceleration is calculated assuming no failures are present and all physical parameters are at their nominal value. N is the number of thrusters. $F_{nom,k}$, an N-by-1 vector containing the expected force from each individual thruster at time step, k , is calculated as:

$$F_{nom,k} = T_k F_{nom} \Gamma_{blowdown}$$

Where T_k is an N-by-1 vector of 1 s and 0 s indicating which thrusters are commanded to fire, F_{nom} is the nominal strength of each thruster at full tank pressure, and $\Gamma_{blowdown}$ is a scalar multiplier representing the reduction in thrust with reduced tank pressure.

$\tau_{nom,k}$, an N-by-1 vector containing the expected torque on the vehicle about the nominal center of mass (CM) due to the thrusters is calculated as

$$\tau_{nom,k} = (L \times D)^T F_{nom,k}$$

Where L is an N-by-3 matrix containing the x-y-z location of each thruster in the body frame, D is an N-by-3 matrix of unit vectors indicating the direction of thrust in the body frame, and $F_{nom,k}$ is from above.

The nominal angular acceleration, is then calculated using the equation of motion from Section 2, the nominal torque calculated above, the nominal spacecraft inertia, I_{nom} , and $\tilde{\omega}_{meas}$ and ω_{meas} which come directly from the gyros.

$$\alpha_{nom,k} = I_{nom}^{-1} (\tau_{nom,k} - \tilde{\omega}_{meas,k} I_{nom} \omega_{meas,k})$$

The $\alpha_{unexpected}$ is calculated as the difference between the expected (nominal) and measured angular accelerations.

3.5 Windowing

If the signal-to-noise ratio were high enough, maximum likelihood FDI analysis of the $\alpha_{un\ expected}$ readings could be carried out on the values at each time step as was done in [2]. However, in this application, values from multiple time steps must be combined to reduce the noise. Since it is known that failures will occur abruptly, a windowing method is preferred over an IIR (e.g., exponential) filter that would carry through information for longer. In this application a window size of 10 (equal to one second) was found to provide a good balance between speed of response and noise reduction. Also, a minimum of 5 samples was required before maximum likelihood FDI analysis would proceed for a given failure mode.

3.6 Collecting measurements for each failure mode

As mentioned earlier, one of the challenges of FDI for systems with on-off actuators is that failures are only observable at certain times. For example, off failures are observable only when the jets are commanded to fire. For each failure mode, only the relevant $\alpha_{un\ expected}$ measurements are stored. So for failure mode #1, any time RCS jet 1 is commanded to fire, the resulting $\alpha_{un\ expected}$ is logged. These two steps of windowing and collecting data can be considered a type of filtering; however implementation as described here avoids introducing any phase lag between the cause (thruster firings) and effect (vehicle motions) that would bias the FDI.

3.7 Maximum likelihood

Although the acceleration estimator is nonlinear and non-optimal, it is reasonable to assume that the estimated unexpected acceleration readings, $\alpha_{un\ expected}$, are normally distributed about the true unexpected acceleration values, $\alpha_{un\ expected}$. So the probability density for the true unexpected acceleration values, $\alpha_{un\ expected}$, conditioned on the measurement history M, is [2] [3]

$$p(\alpha_{un\ expected} | M) = (2\pi)^{-3/2} |P_\alpha|^{-1/2} e^{-\frac{1}{2}(\tilde{\alpha}_{un\ expected}^T P_\alpha^{-1} \tilde{\alpha}_{un\ expected})}$$

where $\tilde{\alpha}_{un\ expected} = \alpha_{un\ expected} - \alpha_{un\ expected}$

and P_α is the estimation error covariance matrix of the unexpected acceleration.

Given unexpected acceleration measurements, $\alpha_{un\ expected}$, and knowing the true unexpected acceleration values, $\alpha_{un\ expected,i}$, for each possible failure mode, the most likely failure mode is found by

finding the $\alpha_{un\ expected,i}$ that maximizes this probability density function. The subscript i indicates the failure mode number corresponding to the unexpected acceleration. This function is maximized when the likelihood argument, λ_i , in the following expression is minimized:

$$\lambda_i = (\alpha_{un\ expected,i} - \alpha_{un\ expected})^T P_\alpha^{-1} (\alpha_{un\ expected,i} - \alpha_{un\ expected})$$

This expression is used both to detect and to identify failures.

3.8 Fault detection

At each FDI update, for each possible failure mode, λ_i is evaluated using the windowed readings (as described earlier, each failure mode's window contains readings for samples only when the failure could be observable — e.g., when failed-off thruster is commanded to fire). The likelihood argument corresponding to no failure, λ_{i0} , is evaluated using the same windowed readings, but with zero substituted for $\alpha_{un\ expected,i}$. When the ratio of likelihood arguments, λ_i/λ_{i0} is falls below a threshold, a failure is declared as having been detected. Since it is important to identify the correct failure mode before declaring it, further tests are performed, as described below. Evaluation of individual λ_{i0} s for each failure mode is critically important — evaluating λ_{i0} based on all (windowed) data may not indicate a failure if a failed-off thruster has not fired recently.

3.9 Fault identification

Once a fault has been detected, at each FDI update, the likelihood arguments, λ_i , are compared to certain thresholds and to each other. For a fault to be declared identified, λ_i must be below a low threshold, while no other faults are below a high threshold. To prevent flickering of fault declarations that would be caused by noise, once a fault has been declared identified, it will remain that way until λ_i rises above a medium threshold. This hysteresis is effective at preventing flickering of fault identifications.

Some faults are virtually indistinguishable from one another, such as this set of four (referring to Figure 2): axial 1 off, axial 2 off, axial 5 on, axial 6 on. This is because each mode produces nearly identical unexpected accelerations. The on vs. off thrusters could be distinguished if translational accelerations were measured and used for FDI. The approach taken here to identify the correct failed thruster is to alter the axial firing pattern (e.g., changing from 1-2-3-5-6-7 on to 2-3-4-6-7-8 on) while maintaining symmetry. Since the firing pattern is adjusted to identify the failed thruster, once the failure has been identified, the pattern is left in a state that does not cause a problem. So if the thruster has failed off (on), it is commanded

to stay off (on), providing reconfiguration as well as fault detection and identification in this case.

3.10 FDI based on RLS analysis

An initial attempt at solving the FDI problem for the X-38 was based on recursive least squares (RLS) analysis. As had been done in [9], thruster parameters were identified using an exponentially weighted RLS algorithm. This approach did not provide sufficiently reliable FDI for the X-38 application for three main reasons:

1. Relatively high noise levels were present (due primarily to gyro noise and pulse-to-pulse thruster variations).
2. Exponential weighting meant that thrusters fired relatively sparsely (e.g., RCS thrusters as compared to axial thrusters) were not identified well
3. Since axial thrusters are fired six at a time, and held on continuously, observability of those parameters was very low.

A second, targeted RLS-based approach used multiple RLS algorithms, each one identifying the strength of a single thruster with the assumption that all other thrusters are operating nominally. This effectively addressed problems 2 and 3 above, but problem 1 remained. Also, the assumption that all other thrusters are nominal causes partial false positives when the failed thruster fires at the same time a good thruster fires. Methods were developed to address the remaining problems, but results were not sufficiently reliable, leading to development of the maximum likelihood-based method.

3.11 Efficiency, Extensions

Many of the terms needed in this analysis, such as $\alpha_{un\ expected,i}$, can be pre-computed or computed periodically. The algorithm is then relatively efficient. It scales better than linearly as more failure modes are added, since some information is shared between analyses of different failure modes (e.g., estimating the unexpected acceleration).

This method can be extended to include translational acceleration as well as angular accelerations. This will provide better discrimination between faults since the comparison space is of higher dimension. It was not included here since the gyros provided sufficient performance for the X-38 application, and to demonstrate that the method will work for systems with gyros only.

4. FDI applied to the X-38

The FDI algorithm was applied to the X-38, with 40 different failure modes simulated, including each of the 8 RCS and 8 axial thrusters being failed-off or failed-on (32 single-jet failures) and 4 pairs of RCS jets being failed off or on (8 multiple-jet failures). Every mode has been tested multiple times and detection and identification is always accurate and within 5 seconds. Fault detection usually takes only 0.5 seconds, and most failures are identified within about 1.0 second¹. The switching of axial thrusters to distinguish between similar failure modes in some cases causes the time for identification to approach 5 seconds.

An example case is discussed here and shown in Figure 4, for RCS jet 1 failed off. The top part of Figure 4 shows the thruster firing history during this 33-second run. The first 8 rows show the RCS jets pulsing to regulate attitude. Then next 8 rows show that axial jets 2-3-4-6-7-8 were on continuously during this run. The next 4 rows correspond to the multiple-jet failure cases, and show when at least one of the jets was commanded to fire. Below that is a zoomed in shot of the detection and identification of RCS jet 1 failure. Below that is a legend corresponding to the thruster history as well as the animation screen below. The bottom part of the figure shows a view of the rear of the vehicle with thrusters firing, torque monitors indicating the net torque produced by the axial and RCS thrusters, and the fault identification result along with a visualization of the likelihood argument, λ_i .

In this simulation, the vehicle starts off with initial angle and rate errors that are corrected by thruster firings in the first two seconds. RCS jet 1 abruptly fails off at 3 seconds, indicated by the gray rectangle, but it is not detected until it fires 23 seconds later. The fault is detected at 26.5 seconds (indicated by the vertical red line), after 0.6 seconds of firing, and is identified at 29.7 seconds (indicated by the change in color from green to red), after a total of 1.0 seconds of firing.

The animation screen at the bottom of Figure 4 was from the final update of the simulation. RCS jets 1 and 6 are both commanded to fire (as also seen in the thruster history screen), but RCS jet 1 is drawn red, indicating that it has failed. The axial-thruster torque

¹ Since failure modes may not be observable depending upon whether their thrusters are commanded to fire, the detection and identification times listed indicate the total duration for which the failure was observable (fired for failed-off, or not fired for failed-on).

monitor shows minimal torque since the axial thrusters are fired symmetrically and the CM is near the center of the jets. The RCS-thruster torque monitor shows a yaw and a roll torque, caused by RCS jet 6. The likelihood monitor bars on the right side are drawn with width equal to $\exp(-0.5 \cdot \lambda_i)$ so they approach 1.0 if the failure is true. RCS jet 1 is close to 1.0, and since it has been declared failed, it is highlighted in red. RCS jet 2 failed on produces an unexpected acceleration signature close to that of RCS jet 1 failed off, which is why it reads above zero (about 0.25). The situation is similar for RCS jet 4 failed on.

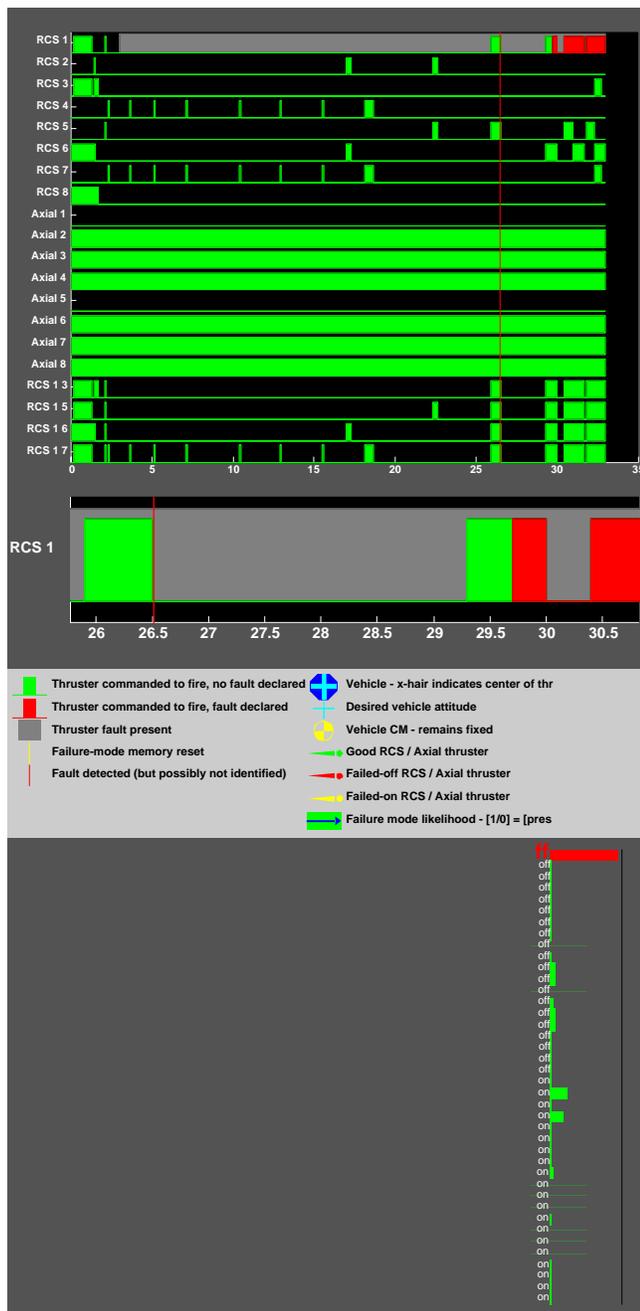


Figure 4: Example simulation run for the X-38

5. Conclusions

A maximum-likelihood-based thruster fault detection algorithm has been presented and applied in simulation to the X-38 spacecraft. The algorithm is capable of reliably detecting and identifying hard, abrupt single- and multiple-jet on or off failures within 1-5 seconds. The algorithm as presented uses gyro signals only, making it applicable to a large number of spacecraft; however, extension to additionally use accelerometer signals is possible, and would provide better discrimination between similar failures. The algorithm is computationally efficient and scales better than linearly with the number of failure modes to be identified.

Acknowledgements

The authors wish to thank Rodolfo Gonzalez, Steven Fredrickson, and Tim Straube of NASA Johnson Space Center and Dave Hammen of LinCom Corp. for providing valuable insights and information for the X-38 and miniAERCam FDI applications.

References

- [1] Chobotov, *Spacecraft Attitude Dynamics and Control*, 1991, p.14
- [2] Deyst, J. J. and Deckert, J. C. Maximum likelihood failure detection techniques applied to the Shuttle RCS jets. *Journal of Spacecraft and Rockets*, vol. 13, no. 2, 65-74, February 1976
- [3] Gelb, A, *et al. Applied Optimal Estimation*, MIT Press, Cambridge, Mass., 1974.
- [4] Isermann, R. Process Fault Detection Based on Modeling and Estimation Methods — A Survey, *Automatica*, Vol. 20, No. 4, pp. 387-404, 1984.
- [5] Lee, A.Y. and Brown, M.J. A model-based thruster leakage monitor for the Cassini spacecraft. In *Proceedings of the American Control Conference*, 1998, vol. 2, pp 902—904, 1998.
- [6] MATLAB is a registered trademark of The MathWorks, Inc., 24 Prime Park Way, Natick, MA 01760, 508-647-7000.
- [7] NASA Figure.
- [8] Willms, B. Space integrated GPS/INS (SIGI) navigation system for Space Shuttle. In *Digital Avionics Systems Conference Proceedings*, 1999.
- [9] Wilson, E. and Rock, S.M. Reconfigurable control of a free-flying space robot using neural networks. In *Proceedings of the American Control Conference*, Seattle WA, June 1995.
- [10] Wilson, E. *Experiments in Neural Network Control of a Free-Flying Space Robot*. PhD thesis, Stanford University, Stanford, CA 94305, March 1995.
- [11] <http://www.dfrc.nasa.gov/Projects/X38/>