JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 32, No. 5, September–October 2009

Fault Detection and Isolation for Deep Space Satellites

Walton R. Williamson,* Jason L. Speyer,† Vu T. Dang,‡ and James Sharp§
SySense, Inc., Burbank, California 91502

DOI: 10.2514/1.41319

A method for detecting faults in the navigation and control system of deep space satellites is presented. A new method for computing the probability of a fault given multiple different types of residuals processors is presented. The method uses the Shiryayev sequential probability ratio test to estimate the probability of the presence of a fault signal given the residuals generated from either parity relationships or fault detection filters, a fault map of the impact of each fault signal on the residuals, and an adaptive fault estimation scheme that enables processing with fewer residuals. This new methodology is applied to the detection of the fault signals in the attitude control system and navigation system of deep space satellites. First a sensor fusion process is presented for blending star tracker data, gyro data, accelerometer data, and information from the vehicle control system to form the best estimate of the navigation state. Then a set of fault detection filters are developed that detect and uniquely identify faults in each of the sensors or actuators. Decision-making is handled through the sequential processing. Simulation results for a single-satellite system are presented.

Nomenclature

A^I_{CG}	=	acceleration of the vehicle at the center-of-gravity point represented in the inertial frame
$[a\times]$	=	cross-product matrix representation of the vector a
$\bar{b}_{ian}$	=	negative adaptive fault-signal estimate of the fault signal μ_i
$\bar{b}_{iap}$	=	positive adaptive fault-signal estimate of the fault signal μ_i
C^B_A	=	cosine rotation matrix to rotate a vector in the A reference frame into the B Frame
$\bar{c}_{ian}$	=	negative adaptive bias threshold for the fault signal μ_i
$\bar{c}_{iap}$	=	positive adaptive bias threshold for the fault signal μ_i
F_i	=	a posteriori probability associated with hypothesized fault i
F_μ	=	discrete-time fault model direction for the fault signal μ
f_i	=	probability density function associated with hypothesized fault i
f_μ	=	fault model direction for scalar fault signal μ
H_i	=	hypothesized fault i
I_B	=	vehicle inertia
L^C_{AB}	=	fixed lever-arm distance from point A to point B in the C reference frame
M	=	a priori covariance of the state x
m	=	vehicle mass
P	=	a posteriori covariance of the state x
P^I	=	position in the inertial frame
p, q, r	=	angular velocity elements of ω
$Q^A_B \otimes Q^B_C$	=	quaternion rotation from C frame to A frame
Q^B_A	=	quaternion representation of a rotation from the A reference frame to the B frame
t	=	time variable
u	=	command input
V^I	=	velocity in the inertial frame
$\|x\|$	=	$L2$-norm of the vector x
$\dot{x}$	=	time derivative of the vector x
$\bar{x}$	=	a priori estimate of the parameter x
$\hat{x}$	=	a posteriori estimate of the parameter x
$\tilde{x}$	=	measurement of the parameter x
Δt	=	change in time variable between updates
δx	=	perturbation about the parameter x (truth minus a priori estimate)
Λ	=	a priori variance of the test residual
μ	=	unknown fault signal
κ_i	=	fault map of the effect of the fault signal i on the residual
Φ	=	discrete-time state transition matrix
ϕ_i	=	a priori probability associated with hypothesized fault i
ω^C_{AB}	=	angular velocity of B frame relative to the A frame represented in the C frame

Presented as Paper 7475 at the AIAA Guidance, Navigation and Control Conference and Exhibit, Honolulu, HI, 18–21 August 2008; received 29 September 2008; revision received 17 March 2009; accepted for publication 3 May 2009.

*Principle Investigator, 300 East Magnolia, Suite 300. Member AIAA.
†Scientist, 300 East Magnolia, Suite 300. Senior Member AIAA.
‡Software Engineer, 300 East Magnolia, Suite 300.
§Research Engineer, 300 East Magnolia, Suite 300.

I. Introduction

TO INCREASE the level of automation in deep space missions and to reduce the cost of ground support, a method for automatic fault detection of a deep space vehicle navigation and actuation systems is described. The goal of the methodology is to define a minimum instrumentation set required to maintain navigation function in the presence of an erroneous measurement signal in any sensor subsystem or an uncommanded response from actuator control system. This erroneous signal is defined as a deviation from the truth beyond assumed statistical uncertainty in the measurements or deviation from a command beyond the statistical uncertainty in the actuation response. It is modeled as an unknown input into the measurements and dynamical system and will be referred to as the fault signal to be detected throughout this paper. As satellite formation-flight concepts and development mature, such as Terrestrial Planet Finder [1] (TPF) requiring distributed automatic control of satellites [2], the need to ensure that each individual spacecraft is healthy increases, because the level of complexity increases with each satellite added.

Systems for attitude control system (ACS) estimation have been well studied in the literature. Extended Kalman filter methods such as those employed by Scharf et al. [3] have been applied to the TPF experiment, which is the approach taken in this paper. Methods for fault detection in Earth-orbiting satellite navigation systems have been studied, for example, by Chen et al. [4], although the dynamics numerically estimated in that work are now explicitly defined

through kinematics, and the filter structure presented here differs significantly. A reduced-order filter structure for detecting star trackers and gyros is developed here that only relies on the star tracker and gyro measurements. A new parity relationship is developed for detecting thruster and momentum wheel faults that explicitly identifies which thruster failed based on parity filters.

Model-based fault detection has been employed in previous flight-test experiments, such as by Lee and Brown [5] for the Cassini satellite. Lee and Brown used the output of the thruster control system combined with an estimate of the angular rate from the attitude control system to generate a parity relationship in angular acceleration. The error was integrated over time and compared with a threshold. The method here performs a similar function, but requires additional accelerometers and gyro measurements to expressly identify thruster faults and leaks. The method is able to identify any fault signals in the thrusters, momentum wheels, or additional accelerometers introduced. The advantage is that the accelerometer method is able to detect smaller thruster faults in a shorter time, which is important for ensuring mission longevity and for avoiding collisions between multiple vehicles operating as part of a TPF-like mission.

The fault-tolerant navigation system devised has several components. A sensor fusion process is implemented using an extended Kalman filter (EKF) to process the actuator commands as well as the instruments to form a state estimate of the position, velocity, and attitude of the vehicle. A bank of residual processors consisting of fault detection filters [6] or parity relationships is used to process the measurements and actuator commands to generate test residuals. The residuals are tested in the multiple-hypothesis Shiryayev sequential probability ratio test [7] (MHSSPRT), a hypothesis-testing scheme used for detecting statistical changes in the residuals in a minimum time, signifying a transition from healthy to unhealthy. The output of the MHSSPRT is the probability of the existence of a fault signal that is used to declare and identify a fault. The goal of the fault detection filter is to generate a residual that forces the fault signal into a particular residual subspace. The MHSSPRT is then used to test that portion of the subspace to identify the source of the particular fault signal. Where appropriate, parity relationships, which do not require memory in the form of state estimation or covariance updates, can provide a simple means of generating residuals that form faults in particular directions. The advantage of this method is that a single MHSSPRT may examine all of the residual processes and make decisions based on statistical performance testing.

Two enhancements to the MHSSPRT are addressed. First, a method for mapping the multiple fault modes and the response of all of the fault detection methods is presented that allows the MHSSPRT to correctly identify the fault mode that may appear in multiple fault detector residual outputs. The result is a fault map showing the effect of each fault signal type on the entire residual space. Second, an adaptive bias scheme is implemented that improves false alarm rejection and fault detection performance. In the healthy case, in which no fault signal exists, all of the residual processors tend to produce a zero-mean residual that can generate ambiguous results in the MHSSPRT due to numerical uncertainty. The adaptive bias estimation scheme uses the fault map to adaptively estimate the fault signal associated with a particular hypothesized fault type. By setting minimum bias limits based on the assumed statistics of the residuals, the adaptive scheme enables rejection of a hypothesized fault when the system is healthy and drives the MHSSPRT to the correct hypothesis when a fault signal appears. Note that neither enhancement requires knowledge of the fault signal, although fault signal sign (positive or negative) is hypothesized for some faults.

The primary instruments considered for this experiment are the use of accelerometers, angular rate gyros, and star trackers. The star trackers provide estimates of the vehicle attitude for the star field. A set of gyros provide angular rate measurements and multiple accelerometer triads are used to measure both vehicle linear acceleration and vehicle angular acceleration. A sensor fusion process is developed for estimating the vehicle attitude state using the available measurements as well as the vehicle actuator commands. Two types of actuation are considered: thrusters and momentum wheels. Thrusters are assumed to be distributed around the perimeter of the vehicle to generate both a linear acceleration and a rotational moment. The momentum wheels provide a pure angular acceleration used to orient the vehicle. The filter methods presented include position measurements. The NASA Deep Space Network (DSN) could provide position updates at low rate to improve estimation and also aid in fault detection. If the satellite is in low Earth orbit (LEO), in which a position measurement such as the Global Positioning System (GPS) may be available. No faults in the positioning measurements are considered, because position fault detection must be performed on Earth in the case of the DSN or because GPS receiver autonomous integrity monitoring techniques are well developed and understood, an example of which is given by Chan and Speyer [8].

Through the design process, the trade space for redundancy is created and the designer may select the number of instruments and instrument types required to ensure navigation system function in the presence of a fault as well as to provide the ability to detect faults in actuators automatically. Simulation results for a preferred minimum configuration are demonstrated at the end of this paper.

II. Fault Detection and Identification Theory

The fault-tolerant navigation system presented in this paper consists of three basic filter types combined with a residual processor. Figure 1 shows the generalized block diagram. The inputs to the system are measurements and actuator commands. The goal of the fault-tolerant navigation system is to detect and isolate the fault signal in any of these inputs. The measurements and actuator commands are processed through a set of filters, including the standard state estimator filter, a bank of fault detection filters, and a bank of parity relationships. The residuals from each of these filters are passed to a residual processor consisting of a hypothesis-testing scheme. A separate hypothesis is proposed for each fault signal assumed. The residual processor provides an estimate of the probability of the fault signal. The navigation state estimation for this process consists of an EKF, which will provide a residual associated with the hypothesis that the system is healthy and no fault signal is present.

A. Residual Generation Using Fault Detection Filters

To detect fault signals in the inputs, test residuals are created that combine the different inputs. The residuals are derived from one of three types of filters: an EKF defining the healthy hypothesis, a fault detection filter (FDF), or a parity relationship. The particular type of fault detection filter used for this paper is based on the disturbance attenuation problem developed by Chung and Speyer [6] and then discretized by Mutuel and Speyer [9]. The filter structure is derived as a minimization problem with the cost function defined in Eq. (1) subject to the dynamic constraint in Eq. (2) and the measurement function in Eq. (3). The dynamics of Eq. (2) define how the state x propagates forward in time as a function of the input control u and process noise w and how fault signals μ_T and μ_N affect the state. The signal μ_T is referred to as the target fault and μ_N is referred to as the nuisance fault. The effect of the target fault on the state is to be amplified, whereas the effect of the nuisance fault is to be attenuated. The strategy is to build filters that are immune to certain nuisance

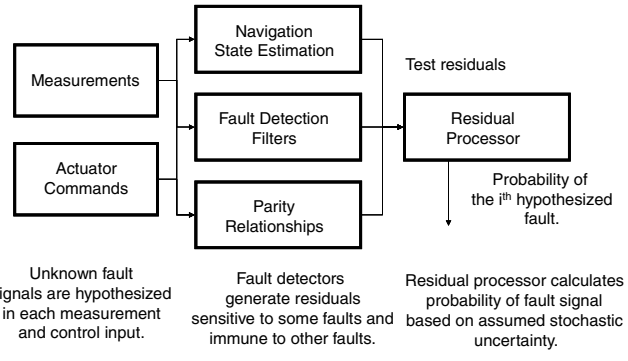

Fig. 1 Fault detection general block diagram.

faults and susceptible to target faults so that the residual process will transmit the target faults and show no effect when a nuisance fault occurs. No assumption is made about either signal μ_T or μ_N. Only the fault signal directions F_T and F_N are assumed known:

$$\min_{\mu_T}\max_{\mu_N}\max_{x(t0)}\frac{1}{2}\int_{t0}^{t}(\|\mu_T\|^2_{Q_T^{-1}}-\|\mu_N\|^2_{Q_N^{-1}}-\|y-Cx\|^2_{V^{-1}})\mathrm{d}t - \frac{1}{2}\|x(t_0)-\hat{x}_0\|^2_P \tag{1}$$

$$\dot{x}=Ax+B_u u+F_T\mu_T+F_N\mu_N+B_w w \tag{2}$$

$$\tilde{y}=Cx+v \tag{3}$$

The measurement function in Eq. (3) shows how the measurement y is derived from the state with additive noise v. Measurement faults may be converted to actuator faults, as discussed by Chung and Speyer [6]. In this case, the unknown fault signal μ enters the measurement in the direction of E, as shown in Eq. (4). A new fault direction f_μ is chosen to satisfy Eq. (5). Note that the choice f_μ may not be unique in Sec. IX, Simulation Results, of this paper. With the selection of f_μ, the actuator fault direction F_μ is defined in Eq. (6) and consists of two directions, as discussed by Chung and Speyer:

$$y=Cx+v+E\mu \tag{4}$$

$$E_\mu=Cf_\mu \tag{5}$$

$$F_\mu=[Af_\mu \quad f_\mu] \tag{6}$$

Questions about output separability, or the ability to detect and isolate different fault signals using a common set of measurements and dynamics as unique within the filter structure, has been shown to consist of a rank test of the observability of all of the fault models through the measurements, as shown by Chung and Speyer [6].

Given the cost function presented and the modeling of the fault input, Mutuel and Speyer [9] present the discrete-time version of the filter structure, as shown in Eqs. (7) and (8), where $\bar{x}$ is the a priori state estimate with associated covariance P and $\hat{x}$ is the a posteriori estimate with associated covariance M. The measurement noise v and the process noise w are assumed to be Gaussian with covariance V and W, respectively. The state transition matrix is defined such that $\Phi(t+\Delta t,t)=e^{A\Delta t}$. The filter structure is very similar to a Kalman filter in structure, but is significantly different due to the introduction of the nuisance and target fault directions with associated weighting functions Q_N and Q_T:

$$\begin{aligned}\hat{x}(t)&=\bar{x}(t)+PH^TV^{-1}(\tilde{y}-C\bar{x}(t))\\ \bar{x}(t+\Delta t)&=\Phi(t+\Delta t,t)\hat{x}(t)\end{aligned} \tag{7}$$

$$\begin{aligned}M&=P-PC(CPC^T+V)^{-1}C^TP\\ P&=\Phi M\Phi^T+W+F_NQ_NF_N^T-F_TQ_TF_T^T\end{aligned} \tag{8}$$

The residual process that will be tested in the MHSSPRT is defined in Eq. (9) using a projector to annihilate the effect of the nuisance fault on the residual. The annihilator is defined in Eq. (10), with n being the smallest integer greater than or equal to zero that ensures that the fault transmission matrix $C\Phi^nF_N$ has full column rank. Note that if the no value of n creates full rank, then the fault is not observable with the set of measurements:

$$\tilde{r}=H_N(\tilde{y}-C\bar{x}) \tag{9}$$

$$H_N=I-(C\Phi^nF_N)[(C\Phi^nF_N)^T(C\Phi^nF_N)]^{-1}(C\Phi^nF_N)^T \tag{10}$$

The use of the projector on the residual in Eq. (9) annihilates the direction in which the fault signal enters into the residual. If the fault signal exists, this residual will remain zero mean. Without the projector, the residual would show the effect of the fault signal transmitted through $C\Phi^nF_N$. Of course, the target faults that are not annihilated by H_N are still visible in the residual, as described by $C\Phi^nF_T$, where n is the appropriate integer greater than or equal to zero such that $C\Phi^nF_T$ has full rank, if it exists. If no nuisance faults are modeled, then no annihilator is necessary in this scheme.

B. Residual Generation with Parity Relationships

Some fault signals are visible in measurements without the use of dynamics. For instance, the output of two instruments of the same type may be compared directly without the need for a dynamic process such as the ability to detect an uncommanded action in thrusters using accelerometer measurements, as will be described. In such cases, simplified parity relationships can be used to generate residuals. Given two measurements $\tilde{y}_1$ and $\tilde{y}_2$, either or both susceptible to a scalar fault signal μ through unique fault directions E_1 and E_2 of the form in Eqs. (11) and (12), then a residual $\tilde{r}$ may be formed as shown in Eq. (13) with associated projector H_E defined in Eq. (14) designed to remove the effect of the fault signal μ from residual $\tilde{r}$. It is assumed that the parity relationship has covariance associated with the zero-mean Gaussian noise variables v_1 and v_2, as shown in Eq. (15) for the case when no fault occurs:

$$\tilde{y}_1=y+E_1\mu+v \tag{11}$$

$$\tilde{y}_2=y+E_2\mu+v \tag{12}$$

$$\begin{aligned}\tilde{r}&=H_E(\tilde{y}_1-\tilde{y}_2)=H_E((E_1-E_2)\mu+v_1-v_2)\\ &=H_E(v_1-v_2)\approx 0\end{aligned} \tag{13}$$

$$H_E=I-(E_1-E_2)[(E_1-E_2)^T(E_1-E_2)]^{-1}(E_1-E_2)^T \tag{14}$$

$$E[(\tilde{y}_1-\tilde{y}_2)(\tilde{y}_1-\tilde{y}_2)^T]=V_1+V_2\text{: }\mu=0 \tag{15}$$

Using the methods defined, a single residual $\tilde{r}$ may be developed that combines the residuals generated from each EKF, fault detection filter, and parity relationship into a single residual at each time. The combined covariance Λ is derived from the stochastic relationships of the filter and parity estimates and represents the assumed uncertainty in the residual process. A fault signal is an input that causes the residual output to change significantly relative to the assumed zero-mean white-noise process with covariance Λ.

C. Residual Processing Using the MHSSPRT

The residual processor used is the MHSSPRT [7]. This scheme tests the incoming residual process against a set of hypothesized probability density functions and then calculates the probability that one hypothesis is correct relative to other hypotheses. The scheme assumes independent identically distributed probability density functions. As will be seen, the method presented here differs in two ways from the method presented by Malladi and Speyer [10]. First, the probability update is restricted to assume that only a single fault signal occurs at a time at the exclusion of all of the other modeled faults. This minor philosophy difference results in an additive probability propagation stage rather than multiplicative. Second, an adaptive bias estimation scheme is developed that enables a single MHSSPRT to process residuals from multiple residual generators rather than implementing ad hoc decision trees based on the output of several different MHSSPRT solutions. Although Malladi and Speyer developed adaptive schemes, the adaptation operated on the estimation structure of an EKF. The adaptive method presented here

operates in cascade from the estimator or residual generators on all residuals processed in the MHSSPRT. The adaptation is not used to provide feedback within the residual generation process.

Given a set of l hypotheses including the healthy hypothesis H_0, the initial probability that each hypothesis is correct is defined by $\phi_i(t_0)$, where the index i is the index of the hypotheses. At each time step, the measurements at that time step are used to generate a residual conditioned on each hypothesis $\tilde{r}_i(t)$. All hypotheses operate on the same measurement history, but each may only test a portion of the residual. Each hypothesis assumes that the residual conforms to a probability density function $f_i(\tilde{r}_i(t))$.

The MHSSPRT update equation calculates the updated probability that a particular hypothesis is correct based on the residual history using the method defined in Eq. (16):

$$F_i(t) = \frac{\phi_i(t) f_i(\tilde{r}_i)}{\sum_{j=0}^{l} \phi_j(t) f_j(\tilde{r}_i)} \tag{16}$$

If the updated healthy probability $F_0(t)$ drops below a certain threshold P_F, then a fault is declared, but the filter continues to operate. If the probability of a particular hypothesis $F_i(t)$ exceeds a certain threshold P_I, then the fault signal source is identified. Processing stops and reconfiguration logic must be invoked to deal with the particular fault mode. Until that point occurs, the processing continues. The probabilities are propagated forward in time using the propagation formulation:

$$\phi_i(t + \Delta t) = F_i(t) + p\left(1 - \sum_{j=1}^{l} F_j(t)\right): j \neq i \tag{17}$$

$$\phi_0(t + \Delta t) = 1 - \sum_{j=1}^{l} \phi_i(t + \Delta t) \tag{18}$$

The propagation updated the probability of a hypothesized fault from time t to time $t + \Delta t$ by assuming a fixed probability of fault between measurements of p. The probability that no fault has occurred ϕ_0 is updated according to Eq. (18). The update in Eqs. (17) and (18) differs from that of Malladi and Speyer [10], which assumed that the transition may occur from the healthy state to a combination of any of the hypotheses. The difference is in the fundamental assumption of how a fault may act within the system. Malladi and Speyer assumed that simultaneous fault signals may exist, and Eq. (17) assumes that only one fault may occur at a time at the exclusion of other faults.

D. Generalized Fault Maps

Given these different types of residual processors, a fault detection strategy may be formulated in which residuals are generated that are susceptible to certain faults and are immune to other faults. A mapping is required to relate the susceptibility of each residual type to a particular fault.

A simple fault map may be defined that shows how each residual is susceptible to a particular fault. This fault map consists of a matrix κ consisting of column vectors κ_i that define the direction that fault μ_i acts on all residual processes. Residuals designed to block a particular nuisance fault will have zeros in the appropriate rows, and residuals that are susceptible to these faults will not. Given l residuals as shown in Eq. (19), each designed to reject the fault model F_j, then the mapping κ_i for F_i can be calculated as shown in Eq. (20), in which where is noted that $H_i C \Phi^{nii} F_i = 0$ by design if F_i is a nuisance fault. At least one residual should actually be designed as a target fault so that the fault signal is clearly visible in at least one residual, unless the goal is simple elimination and the designer is willing to forego detection and isolation. Each of the integers nji is chosen as the minimum integer value greater than or equal to zero such that $H_j C \Phi^{nji} F_i \neq 0$. Note that it is perfectly acceptable to have $H_j C \Phi^{nji} F_i = 0$ showing that a particular residual r_j is not susceptible to a particular F_i, if that is the case. In that instance, $\kappa_{ji} = 0$ by definition:

$$\tilde{r} = [\,\tilde{r}_1 \quad \tilde{r}_2 \quad \cdots \quad \tilde{r}_i \quad \cdots \quad \tilde{r}_q\,]; \qquad E[\tilde{r}\tilde{r}^T] = \Lambda \tag{19}$$

$$\kappa_i = [\,(\kappa_{1i})^T \quad (\kappa_{2i})^T \quad \cdots \quad (\kappa_{ii})^T \quad \cdots \quad (\kappa_{li})^T\,]^T \tag{20}$$

$$\kappa_{ji} = H_j C \Phi^{nji} F_i; \qquad \kappa_{ii} = C \Phi^{nii} F_i \tag{21}$$

An entire fault map κ may be constructed for all l faults such that $\kappa = [\,\kappa_1 \quad \kappa_2 \quad \cdots \quad \kappa_i \quad \cdots \quad \kappa_l\,]$. Note that κ contains not just direction information, but also relative magnitude.

E. Adaptive Bias Estimation in the MHSSPRT

If no fault exists, the residual generators as presented are all designed to produce residuals that have a zero mean with covariance Λ. If a zero-mean Gaussian is selected as the density function for all of the residuals, then all tests are consistent with the hypothesized density function and the MHSSPRT cannot distinguish between hypotheses. Because an annihilator sets a portion of the residual to zero, the MHSSPRT is more likely to select the projected residual as most nearly matching a Gaussian distribution than a white-noise nonzero residual process generated by the best state estimator.

To counteract this problem, an adaptive bias estimation scheme is proposed. This scheme estimates a bias $\bar{b}_i$ in the residual process associated with the direction κ_i. A separate bias is estimated for each fault μ_i, and the bias is only estimated on those elements of the residual having nonzero value for κ_i. A minimum bias value $\bar{c}_i$ is defined for $\bar{b}_i$, which forces a bias into the residuals hypothesizing a fault signal. The bias $\bar{b}_i$ is estimated and removed from the residual before testing in the MHSSPRT. If no fault signal exists, the unhealthy hypotheses are appropriately weighted relative to their density functions and the healthy hypothesis most closely matches its density function. If a fault signal exists, the bias tracks the effect of the fault signal on the residuals removing its effect. The density function hypothesizing that fault signal most closely matches a zero-mean Gaussian. Because the bias is removed from the residual before use in the density function, both a positive and negative bias must be used, which is equivalent to hypothesizing two different fault signals for each unknown fault input μ_i: one positive and one negative. Although this approach does effectively bias the unhealthy hypotheses, the cost is that both a positive and negative bias must be estimated, effectively doubling the number of hypotheses.

The residual density functions are assumed to be Gaussian, as shown in Eq. (22), although other density functions may be assumed. In this case, N is the size of the residual vector $\tilde{r}$, Λ is the assumed covariance of the residual, and $\bar{b}_i$ is a bias to be adaptively estimated as part of this process:

$$f_i(\tilde{r}) = \frac{1}{2\pi^{N/2}|\Lambda|} e^{\frac{-1}{2}(\tilde{r}-b_i)^T \Lambda^{-1}(\tilde{r}-b_i)} \tag{22}$$

A minimum bias $\bar{c}_{ip}$ is defined as shown in Eq. (23), where $\sqrt[c]{\Lambda}$ is the Cholesky decomposition of the covariance. The bias $\bar{c}_{in}$ is the negative of the positive bias $\bar{c}_{ip}$. Because no assumptions were made on the fault signal μ_i, the adaptive scheme takes into account the fact that the signal may be positive or negative. A positive μ_i results in changes in the residual associated with the fault map κ_i, whereas a negative μ_i results in changes in the residual associated with the negative of the fault map $-\kappa_i$:

$$\bar{c}_{ip} = \kappa_i \sqrt[c]{\Lambda}; \qquad \bar{c}_{in} = -\bar{c}_{ip} \tag{23}$$

The bias $\bar{b}_{iap}$ is estimated adaptively using a low-pass filter such as an averaging filter depicted in Eq. (24). In this case, k is the number of steps over which to average, κ_{ia} is the ath element of κ_i, and $\tilde{r}_a(t)$ is the ath element of the residual $\tilde{r}$ at time t. Likewise, a negative bias $\bar{b}_{nap}$ is calculated as in Eq. (25). Each bias is estimated over time:

$$\bar{b}_{iap}(t) = \frac{(k-1)}{k} \bar{b}_{iap}(t - \Delta t) + \kappa_{ia} \frac{1}{k} \tilde{r}_a(t) \tag{24}$$

$$\bar{b}_{ian}(t) = \frac{(k-1)}{k} \bar{b}_{ian}(t - \Delta t) + \kappa_{ia} \frac{1}{k} \tilde{r}_a(t) \tag{25}$$

$$\bar{b}_{iap} = \max(\bar{b}_{iap}, \bar{c}_{iap}) \tag{26}$$

$$\bar{b}_{ian} = \min(\bar{b}_{ian}, \bar{c}_{ian}) \tag{27}$$

At each time step and on an element-by-element basis, if the bias $\bar{b}_{iap}$ drops below the minimum value set by $\bar{c}_{iap}$, then the value is reset to the minimum value of $\bar{c}_{iap}$. Similarly, if the value of the negative bias $\bar{b}_{ian}$ increases above the value set by $\bar{c}_{ian}$, then the bias value is reset to the minimum value of $\bar{c}_{ian}$. The value of $\bar{c}_{iap}$ and $\bar{c}_{ian}$ effectively set a "dead band" in which faults that do not change the residual above the values associated with $\bar{c}_{iap}$ do not register as faults in the system, which is why the biases $\bar{c}_{iap}$ and $\bar{c}_{ian}$ are chosen with respect to the assumed covariance. The designer may choose to set minimum fault threshold $\bar{c}_{iap}$ as a one-sigma value N-sigma value by appropriately weighting Eq. (23).

Each hypothesized residual $\tilde{r}_i$ is appropriately biased during the healthy condition. If a positive fault signal μ_i enters the system, then the $\bar{b}_{iap}$ terms adaptively remove the effect of the fault signal on the residual process, ensuring that the residual $\tilde{r}_i$ remains zero mean, whereas other residuals without the precise adaptation suited to fault μ_i are nonzero mean. If a negative fault μ_i enters the system, then the $\bar{b}_{ian}$ elements will adaptively estimate and remove the fault from the system. This bifurcation between positive and negative fault signals is introduced into the MHSSPRT as additional hypotheses. Two sets of hypotheses for each fault μ_i are assumed at this point: one for a positive fault and the other for a negative fault signal. If l faults are defined, then the MHSSPRT has $2l + 1$ hypotheses, l positive faults, l negative faults, and the healthy hypothesis. The residual process for the ith positive hypothesis is defined in Eq. (28) and the residual process for the ith negative hypothesis is defined in Eq. (29):

$$\tilde{r}_{ip} = \tilde{r} - \bar{b}_{ip} \tag{28}$$

$$\tilde{r}_{in} = \tilde{r} - \bar{b}_{in} \tag{29}$$

Two explicit hypotheses are generated for each fault signal (one positive and one negative); each pair is actually only a single fault. It is suggested that the resulting pair of probabilities for the positive and negative fault signals are summed after updating the MHSSPRT to properly identify a single failure, although this is not necessary.

The method presented here is different from that previously derived by Malladi and Speyer [10], in which the adaptive scheme was applied to the Kalman filter. In this scheme, the adaptive bias estimate is instead applied only to the residual processes. The bias estimation scheme presented here does not interfere with any of the filter processes. The advantage of this technique is that a single MHSSPRT may be used to estimate the probability of a fault in any of the l hypothesized faults without discrete decision-making processes external to the MHSSPRT. The next sections apply this scheme and the methods suggested to a single-satellite estimation problem.

III. Single-Satellite Vehicle Model and Dynamics

The vehicle model used for this experiment is reasonably generic, but similar to designs for the Jet Propulsion Laboratory TPF experiment [1,3]. The experiment, in the planning stages, is currently set to operate at the $L2$ libration point. The assumed location at the L_2 libration point enables the use of simplified dynamics. No gravity terms are currently used. The state of a satellite is described in Eq. (30). The position is described by the vector P^I, the velocity with respect to inertial is represented by V^I, the quaternion rotation from the body frame to the inertial is represented by Q_I^B, and the angular velocity from the inertial to the body frame in the body frame is represented by ω_{IB}^B:

$$x = \begin{bmatrix} P^I \\ V^I \\ Q_I^B \\ \omega_{IB}^B \end{bmatrix} \tag{30}$$

The state is propagated in time using nonlinear equations of motion. The dynamics of the position and velocity are described in Eqs. (31) and (32):

$$\dot{P}^I = V^I \tag{31}$$

$$\dot{V}^I = \frac{1}{m} C_B^I \sum_{i=1}^{N} B_{Tti} u_{Ti}^B \tag{32}$$

The term B_{Tti} represents the sensitivity of the acceleration dynamics to the input from the ith thruster command u_{Ti}^B. Thruster commands are specified in units of newtons. The mass of the vehicle is defined by the variable m. The cosine rotation matrix C_B^I is calculated from the quaternion Q_B^I and rotates the thruster commands into the inertial frame. All of the thruster inputs are summed to form the acceleration. The rotational dynamics are defined in Eqs. (33–35) using the methods defined by Zipfel [11] for relating input moments to angular velocity:

$$\dot{Q}_I^B = \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} Q_I^B \tag{33}$$

$$\omega_{IB}^B = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{34}$$

$$\dot{\omega}_{IB}^B = I_B^{-1}([\omega_{IB}^B \times] I_B \omega_{IB}^B) + I_B^{-1} \left(\sum_{i=1}^{N} B_{Tri} u_{Ti}^B + \sum_{j=1}^{S} B_{Mj} u_{Mj}^B \right) \tag{35}$$

The inertia of the vehicle in the vehicle body frame is defined as I_B and the command inputs drive the dynamics resulting from the N thruster commands u_{Ti}^B and the S momentum wheel commands u_{Mj}^B. In Eq. (33), p, q, and r are the roll, pitch, and yaw rates, respectively. Equations (31–35) are integrated using a nonlinear integration routine to translate specified thruster and momentum wheel commands into angular velocity, attitude, velocity, and position states. These states are referred to as the "truth" for the simulation and used to generate measurement errors.

The vehicle configuration used in the simulations has 16 thrusters and 3 momentum wheels. Each thruster fires in the direction of a unit vector d_i. A lever arm L_i^{th} defines the location of the thruster relative to the center of mass. Thrusters cause both linear and rotational motion. The sensitivity matrix B_{Tt} representing the linear portion of the thruster input is defined in Eq. (36):

$$B_{Tt} = [\, B_{Tt1} \quad B_{Tt2} \quad \cdots \quad B_{Tt16} \,] = [\, d_1 \quad d_2 \quad \cdots \quad d_{16} \,] \tag{36}$$

The sensitivity matrix for the rotational portion of the thruster input B_{Mt} is defined in Eq. (37):

$$\begin{aligned} B_{Tr} &= [\, B_{Tr1} \quad B_{Tr2} \quad \cdots \quad B_{Tr16} \,] \\ &= [\, d_1 \times L_1^{th} \quad d_2 \times L_2^{th} \quad \cdots \quad d_{16} \times L_{16}^{th} \,] \end{aligned} \tag{37}$$

Momentum wheels provide pure moment inputs. Assuming that each momentum wheel provides input along a principle axis of the body, then the sensitivity matrix B_M for the rotational input of the three momentum wheels is defined in Eq. (38):

$$B_M = [\, B_{M1} \quad B_{M2} \quad B_{M3} \,] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{38}$$

Figure 2 shows a diagram of the thruster arrangement, and Table 1 provides the vehicle mass and moment of inertia values for a single TPF vehicle used in the simulation.

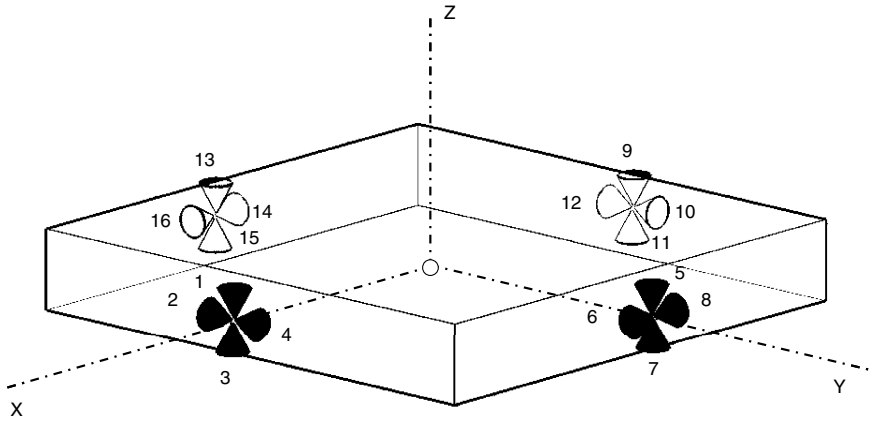

Fig. 2 Vehicle thruster configuration.

Note that the dynamics provided are only applicable to the simplified low-gravity case at the libation point. These dynamics may be adapted for an orbital parameter set, if required.

IV. Measurement Models

Four types of measurements are examined in this study. The star trackers estimate the attitude of the vehicle. Angular rate gyros and accelerometers measure inertial quantities. The DSN provides a simplified position measurement that is available periodically. Alternatively, this method could be applied toward low-Earth-orbit satellites using a GPS input in which a GPS receiver provides a position solution to the navigation system. Error models for the thrusters and momentum wheel estimates are also described.

The measurement error model used for the rate gyros is nonlinear and is given by Eq. (39), where the measured angular velocity $\tilde{\omega}^B_{IB}$ is a function of the true inertial angular velocity ω^B_{IB}, but rotated by the error in the assumed vehicle attitude relative to the inertial frame $C^{\bar{B}}_B$ with additive sensor noise:

$$\tilde{\omega}^{\,B}_{IB} = C^{\bar{B}}_B \omega^B_{IB} + \upsilon_\omega \tag{39}$$

The error in the true rotation from the inertial to the body frame C^I_B is defined using a nonlinear rotation of the a priori rotation matrix $C^I_{\bar{B}}$ and the error in the rotation $C^{\bar{B}}_B$, which is linearized in terms of the error in the quaternion δq, as shown in Eq. (40):

$$C^I_B = C^I_{\bar{B}} C^{\bar{B}}_B \approx C^I_{\bar{B}}(I + 2[\delta q\times]) \tag{40}$$

In this case, the error in the attitude δq is a 3×1 vector of the linearized error in the quaternion. When δq is estimated in the filter structure, then the full error quaternion $Q^{\bar{B}}_B$ can be reconstructed using the following step:

$$Q^{\bar{B}}_B = \frac{1}{\sqrt{1.0 + \delta q^T \delta q}} \begin{bmatrix} 1 \\ \delta q \end{bmatrix} \tag{41}$$

The star tracker measures the attitude of the vehicle body relative to the inertial, which is a function of the true attitude corrupted by a rotation error. The measurement equation for the star tracker is in Eq. (42):

$$\tilde{C}^{\,B}_I = C^B_{\bar{B}} C^{\bar{B}}_I + C^B_{\bar{B}} C^{\bar{B}}_I [\upsilon_q\times] \tag{42}$$

Table 1 TPF mass properties

Property	Value	Units
Mass	879	kg
I_{xx}	2787	$\text{kg} \cdot \text{m}^2$
I_{yy}	2836	$\text{kg} \cdot \text{m}^2$
I_{zz}	2266	$\text{kg} \cdot \text{m}^2$

Substituting Eq. (40) into this, the star tracker measurement equation yields

$$\tilde{C}^{\,B}_I = (I - 2[\delta q\times])C^{\bar{B}}_I + (I - 2[\delta q\times])C^{\bar{B}}_I [\upsilon_q\times] \tag{43}$$

Let

$$\tilde{C}^{\,B}_I C^I_{\bar{B}} = C^{\bar{B}}_{\bar{B}} = I - [\delta\theta\times] \tag{44}$$

Distributing terms, neglecting higher-order terms, and using the identity in Eq. (44), we can derive the attitude error equation:

$$[\delta\theta\times] = 2[\delta q\times] + C^{\bar{B}}_I [\upsilon_q\times] C^I_{\bar{B}} \tag{45}$$

A simplified accelerometer measurement is used in which the acceleration measurement is simply the truth corrupted by additive noise υ_a:

$$\tilde{a} = \bar{a} + \upsilon_a \tag{46}$$

Similarly, for the DSN, the measured position taken to be the true position plus the position error plus noise υ_P, as shown in Eq. (47). This position measurement is overly simplified, but applicable to both the DSN and to GPS in LEO types of orbits:

$$\tilde{P} = P + \upsilon_P \tag{47}$$

The estimate of the thruster output $\bar{u}^B_{Ti}$ is assumed to be equal to the true output u^B_{Ti} corrupted by noise of the following form:

$$\bar{u}^{\,B}_{Ti} = u^B_{Ti} + w_{Ti} \tag{48}$$

Likewise, momentum wheel command estimates $\bar{u}^B_{Mj}$ are defined as the sum of the true momentum wheel command u^B_{Mi} with additive noise:

$$\bar{u}^{\,B}_{Mj} = u^B_{Mj} + w_{Mj} \tag{49}$$

The noise terms representing the uncertainty in thrust and moment output w_{Ti} and w_{Mj} are assumed to be Gaussian with zero mean and are independent of all other instruments or controls. These noise terms represent uncertainty in the manufacture and output of the actual hardware device. If the control system commands a particular thrust, the noise term represents the difference between the commanded and actual thrust, the difference being caused by manufacturing defects or limits of precision.

V. Extended Kalman Filter for Navigation

An extended Kalman filter is generated by linearizing the dynamics presented in Sec. III around a nominal state estimate and estimating the error in the state based on the measurement models

presented in Sec. IV. The filter structure forms the basis for the fault detection filter structures presented later in this paper.

A nominal state estimate $\bar{x}$ with associated error covariance M is defined in Eq. (50) based on the true state in Eq. (30):

$$x = \begin{bmatrix} \bar{P}^I \\ \bar{V}^I \\ \bar{Q}_I^B \\ \bar{\omega}_{IB}^B \end{bmatrix} \tag{50}$$

The nominal state $\bar{x}$ is the a priori estimate used for the filter. The error in the a priori estimate is defined using linear perturbations for position, velocity, and angular velocity, but with nonlinear perturbations for attitude. The position perturbation δP and velocity perturbation δV are defined in Eqs. (51) and (52):

$$P^I = \bar{P}^I + \delta P \tag{51}$$

$$V^I = \bar{V}^I + \delta V \tag{52}$$

The attitude rotation perturbation is defined by first calculating the cosine rotation matrix $\bar{C}_I^B$ derived from the a priori attitude $\bar{Q}_I^B$. The quaternion error perturbation δq is a 3×1 vector and was first described in Eq. (40) and defined as the first-order perturbation of the rotation matrix $C_B^{\bar{B}}$, the matrix that describes the rotation from the estimated body frame to the true body frame. The true angular velocity is defined as a linear sum of the a priori with the perturbed angular velocity error:

$$\omega_{IB}^B = \bar{\omega}_{IB}^B + \delta\omega \tag{53}$$

Using the definitions presented and the dynamics presented for the satellite, it is possible to form the dynamics describing the propagation and growth of the error as a function of the thrust and momentum wheel inputs, as shown in Eq. (54). The perturbed state δx is defined in Eq. (55), with the dynamics matrix A defined without proof in Eq. (56). The control sensitivity matrix B is defined in Eq. (57) for the control vector u defined in Eq. (58). The noise matrix for this problem is equal to the sensitivity matrix ($B = \Gamma$). The stochastic, zero-mean, Gaussian noise variable w is defined in terms of the thruster and momentum wheel noise in Eq. (59):

$$\delta\dot{x} = A\delta x + Bu + \Gamma w \tag{54}$$

$$\delta x = \begin{bmatrix} \delta P \\ \delta V \\ \delta q \\ \delta\omega \end{bmatrix} \tag{55}$$

$$A = \begin{bmatrix} 0 & I & 0 & 0 \\ 0 & 0 & -2\frac{1}{m} C_{\bar{B}}^I \left[\left(\sum_{i=1}^N B_{Tti} u_{Ti}^B \right) \times \right] & 0 \\ 0 & 0 & -[\bar{\omega}_{IB}^B \times] & \frac{1}{2} I \\ 0 & 0 & 0 & I_B^{-1}([\bar{\omega}_{IB}^B \times] I_B \bar{\omega}_{IB}^B) \end{bmatrix} \tag{56}$$

$$B = \begin{bmatrix} 0 & 0 \\ \frac{1}{m} C_{\bar{B}}^I B_{Tti} & 0 \\ 0 & 0 \\ I_B^{-1} B_{Tri} & I_B^{-1} B_{Mj} \end{bmatrix} \tag{57}$$

$$u = \begin{bmatrix} \bar{u}_{Ti}^B \\ \bar{u}_{Mj}^B \end{bmatrix} \tag{58}$$

$$w = \begin{bmatrix} w_{Ti} \\ w_{Mj} \end{bmatrix} \tag{59}$$

This filter processes all of the measurements available including position, attitude from both star trackers, and angular velocity. The residual is defined in Eq. (60):

$$\tilde{r} = \begin{bmatrix} \tilde{P} - \bar{P} \\ \delta\theta_1 \\ \delta\theta_2 \\ \tilde{\omega} - \bar{\omega} \end{bmatrix} = \begin{bmatrix} I_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 2I_3 & 0_3 \\ 0_3 & 0_3 & 2I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \begin{bmatrix} \delta P^O \\ \delta V^O \\ \delta q \\ \delta\omega_{IB}^B \end{bmatrix} + \begin{bmatrix} v_p \\ C_{S1}^B v_{S1} C_B^{S1} \\ C_{S2}^B v_{S2} C_B^{S2} \\ v_\omega \end{bmatrix} \tag{60}$$

The associated error matrix C is defined in Eq. (61):

$$C = \begin{bmatrix} I_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 2I_3 & 0_3 \\ 0_3 & 0_3 & 2I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \tag{61}$$

The state error is corrected using Eq. (62):

$$\hat{x} = \bar{x} + K\delta\hat{x} \tag{62}$$

A square-root filter is used to update the state estimate and error covariance or, in this case, the square root of the error covariance described by Maybeck [12]. We can find the square root of the a priori error covariance M by taking the Cholesky decomposition defined in Eq. (63):

$$\bar{S} = \sqrt[C]{M} \tag{63}$$

We can use $\bar{S}$ to compute the filter gain K as follows:

$$\bar{A} = \bar{S}^T C^T; \qquad \Sigma_i = \sqrt[C]{\bar{A}^T \bar{A} + R}; \qquad K = \bar{S}\,\bar{A}[\Sigma_i^{-1}]^T \Sigma_i^{-1} \tag{64}$$

Note that $\bar{A}$ is not the same as the linearized dynamics, but a placeholder value. The variable R is the measurement covariance matrix. The error covariance matrix is updated by Eq. (65):

$$\hat{S} = \bar{S} - \bar{S}\,\bar{A}[\Sigma_i^{-1}]^T [\Sigma_i + \sqrt[C]{R}]^{-1} \bar{A} \tag{65}$$

Between updates, the state estimate and the error covariance are propagated forward in time. The state estimate is propagated in time using the a posteriori state estimate and control inputs. The updated covariance P is calculated from the updated square-root covariance $P = \hat{S}^T \hat{S}$.

The linear dynamics found in Eq. (56) are used to propagate the error covariance matrix. The linear equations given previously are written in continuous time and must be converted to discrete time for use in the filter:

$$\Phi = e^{A\Delta t}; \qquad \Gamma_D = \int_0^{\Delta t} e^{A\tau} \Gamma \mathrm{d}\tau \tag{66}$$

The error covariance is propagated using

$$M = \Phi P \Phi^T + \Gamma_D W \Gamma_D^T \tag{67}$$

where $P = \hat{S}\hat{S}^T$ and W is the process noise variance.

VI. Residual Generators for Star Tracker and Gyro Faults

Stochastic fault detection filters such as those defined in Eqs. (7) and (8) are chosen as a means of generating test residuals for

detecting star tracker and gyro faults. Because the star tracker measures essentially inertial attitude and because the gyro measures inertial angular velocity, a decoupled reduced-order filter can be generated to detect faults in the system. The filter structure is similar to ACS attitude estimation schemes using linearized extended Kalman filter structures, although the present case is modified for fault detection. The gyros are used as inputs to the dynamic system and the star trackers are used to correct the attitude state. A filter structure consisting only of the attitude and gyro bias error is constructed with dynamics defined in Eq. (68) and measurement defined in Eq. (69) with measurement sensitivity matrix $C_{\theta 1} = [2I \quad 0]$. The measurement residual $\tilde{r}_{\theta 1}$ assumes that only star tracker 1 is used of the two available. The system described is susceptible to faults in the gyros μ_ω or the first star tracker $\mu_{\theta 1}$ through fault directions F_ω and $E_{\theta 1}$, respectively. The gyro fault model is described by $F_\omega = [0 \quad I]^T$ for three-dimensional fault μ_ω, one fault for each axis of the gyro. This choice of the fault signal requires that $n = 1$ to generate the annihilator H_ω using the formulation in Eq. (10). Note that the inverse required to calculate the annihilator always exists.

Similarly, the star tracker fault model is described simply by the known constant rotation matrix $E_{\theta 1} = C_{S1}^B$ for three-dimensional fault $\mu_{\theta 1}$, one fault for each rotation axis:

$$\begin{bmatrix} \delta\dot{q} \\ \delta\dot{\omega} \end{bmatrix} = \begin{bmatrix} -[\bar{\omega}_{IB}^B\times] & \frac{1}{2}I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \delta q \\ \delta\omega \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix}\upsilon_\omega + F_\omega\mu_\omega \tag{68}$$

$$\tilde{r}_{\theta 1} = C_{\theta 1}\begin{bmatrix} \delta q \\ \delta\omega \end{bmatrix} + C_{S1}^B\upsilon_{S1}C_B^{S1} + E_{\theta 1}\mu_{\theta 1} \tag{69}$$

As discussed by Chung and Speyer [6], a measurement fault can be converted to a fault in the dynamics system such that the fault signal $\mu_{\theta 1}$ and its derivative appears to enter through the dynamics. A measurement fault in a single instrument is represented as a fault in the dynamic that has two dimensions. The star tracker fault model $F_{\theta 1}$ is constructed from the fault model $E_{\theta 1}$ in the measurement model by first defining $f_{\theta 1}$ using $E_{\theta 1} = C_{\theta 1}f_{\theta 1}$. Then $F_{\theta 1}$ is constructed and the result is shown in Eq. (70) to generate a two-dimensional fault $\bar{\mu}_{\theta 1}$, shown in Eq. (71). The choice of $f_{\theta 1}$ is somewhat arbitrary and left to the designer. One choice is $f_{\theta 1} = [0.5I \quad 0]^T$. However, Eq. (70) does not have full column rank and a deficiency exists. Further, the gyro fault output through the residual has the same signature as the star tracker fault, because $C_{\theta 1}A_\theta f_\omega = C_{\theta 1}f_{\theta 1}$, where $A_{\theta 1}$ is the dynamics matrix of Eq. (68):

$$F_{\theta 1} = [A_{\theta 1}f_{\theta 1} \quad f_{\theta 1}] \tag{70}$$

$$\bar{\mu}_{\theta 1} = \begin{bmatrix} \mu_{\theta 1} \\ \dot{\mu}_{\theta 1} \end{bmatrix} \tag{71}$$

The problem is essentially that the dynamics and measurement model of Eqs. (68) and (69) do not have unique fault signatures. One alternative would be to use the navigation filter defined in Eqs. (54–61) as a fault filter. This method would still require the conversion of the star tracker sensor fault to a plant fault. Two problems arise with this approach. First, the new fault model $F_{\theta 1}$ would corrupt the velocity estimates through the thruster dynamics. To maintain output separability relative to the gyro measurements, the position measurements must be included as part of the residual process. Because the DSN measurement updates are only available at very low intervals (on the order of one update every few hours), the ability to detect a fault in a reasonable time frame is very limited. Further, the unique direction of the fault would only be visible if the vehicle was maneuvering or rotating. Otherwise, $[\bar{\omega}_{IB}^B\times] = 0$ and the dynamics of Eq. (68) are reduced to simple integration. A fault in the star tracker is indistinguishable from a fault in the gyros. Similarly, the dynamics of Eq. (56) only generate a unique fault signature if the vehicle is thrusting. If all thrusters are set to zero output, then the star tracker fault model will not propagate through the velocity measurements and the same problem exists as with this reduced-order filter presented in this section. The result is that under conditions with no motion, a fault in the star tracker or gyros may be detected but cannot be uniquely identified.

In this case, hardware redundancy is proposed as the solution. Because deep space applications require attitude determination capability, two star trackers are suggested. The loss of one will not result in the loss of mission capability. A parity relationship is devised between the two star trackers. As shown in Eq. (45), the residual for the star tracker takes the form in Eq. (72):

$$\begin{aligned} [\tilde{r}_{\theta 1}\times] &= C_I^{\bar{B}}\tilde{C}_{S1}^I C_B^{S1} - I = C_\theta\begin{bmatrix} \delta q \\ \delta\omega \end{bmatrix} + C_{S1}^B\upsilon_{S1}C_B^{S1} + E_{\theta 1}\mu_{\theta 1} \\ [\tilde{r}_{\theta 2}\times] &= C_I^{\bar{B}}\tilde{C}_{S2}^I C_B^{S2} - I = C_\theta\begin{bmatrix} \delta q \\ \delta\omega \end{bmatrix} + C_{S2}^B\upsilon_{S2}C_B^{S2} + E_{\theta 2}\mu_{\theta 2} \end{aligned} \tag{72}$$

The parity residual is described in Eq. (73), which is susceptible to both faults $\mu_{\theta 1}$ and $\mu_{\theta 2}$. Testing this residual will detect a fault in one star tracker:

$$\begin{aligned} [\tilde{r}_{\theta 12}\times] = [\tilde{r}_{\theta 1}\times] - [\tilde{r}_{\theta 2}\times] &= \tilde{C}_{S1}^I C_B^{S1} - \tilde{C}_{S2}^I C_B^{S2} \approx C_{S1}^B\upsilon_{S1}C_B^{S1} \\ &+ E_{\theta 1}\mu_{\theta 1} - C_{S2}^B\upsilon_{S2}C_B^{S2} - E_{\theta 2}\mu_{\theta 2} \end{aligned} \tag{73}$$

Then two fault detection filters are constructed using the dynamics defined in Eqs. (68) and (69). The first uses only measurements from the first star tracker and the second uses measurements from the second star tracker. At each time step with new gyro measurements, the state for each filter is propagated using Eq. (33). When star tracker measurements are made available, the fault detection filter structure, which assumes only gyro faults, is calculated using the dynamics and measurement models in Eqs. (68) and (69) using the methods described in Eqs. (7) and (8).

The test residual $\tilde{r}_{T\theta}$ to be combined with the other tests in the next section is constructed from the residuals of each filter and the parity relationship, as shown in Eq. (74). The associated covariance is defined in Eq. (75), where $M_{\theta 1}$ is the covariance of the first fault detection filter, $M_{\theta 2}$ is the covariance of the second filter, and $V_1 = C_{S1}^B E[\upsilon_{S1}\upsilon_{S1}^T]C_B^{S1}$ and $V_2 = C_{S2}^B E[\upsilon_{S2}\upsilon_{S2}^T]C_B^{S2}$. Several simplifying assumptions are made in Eq. (75) dealing with the correlation of the gyro measurements between the two fault detection filters for convenience. It is assumed that the noise in each star tracker is independent of the other:

$$\tilde{r}_{T\theta} = \begin{bmatrix} \tilde{r}_{\theta 1} \\ \tilde{r}_{\theta 2} \\ \tilde{r}_{\theta 12} \end{bmatrix} \tag{74}$$

$$\Lambda_{T\theta} = \begin{bmatrix} C_\theta M_{\theta 1}C_\theta^T + V_1 & 0 & V_1 \\ 0 & C_\theta M_{\theta 2}C_\theta^T + V_2 & -V_2 \\ V_1 & -V_2 & V_2 + V_1 \end{bmatrix} \tag{75}$$

The test residuals are calculated after each update and tested in the MHSSPRT to be described in a later section.

VII. Fault Detection Using Three Accelerometer Triads

At this point, methods for detecting faults in the star tracker and rate gyros have been defined. Fault detection filters could be developed for a fault in each of the thrusters and momentum wheel using the dynamics of the navigation filter described, in which one filter is created for each thruster fault or momentum wheel fault. An alternative scheme with lower computational power is developed in this section. Computational power is exchanged for additional hardware in the form of low-cost accelerometers precisely positioned on the vehicle. The result is a memoryless residual generator that detects faults in all of the accelerometer measurements, all of the thrusters, and all of the momentum wheels. The simplified filter structure and processing requirements may offset the cost of additional instruments, especially because the instruments are merely low-cost accelerometers.

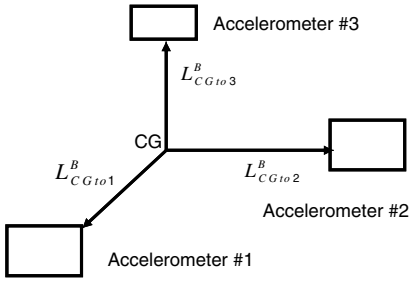

Fig. 3 Acceleration lever arms on the principle axes of the vehicle.

Figure 3 shows a general orientation of the accelerometers. The selection of orientation is not arbitrary, as will be discussed, but this orientation is not necessarily optimal. Each of the accelerometers is placed such that it will experience accelerations and moments generated by the thrusters and momentum wheels in a different manner. The lever arms are placed along the principle vehicle axes. Each lever arm is assumed to be 0.5 m long, half the length of the thruster lever arms. Three sensors are chosen as the minimum number allowable that is completely redundant, meaning that even faults in the accelerometers can be isolated by the fault detection techniques presented.

The acceleration experienced at a sensor triad located at a fixed distance from the center of gravity (CG) is given by Eq. (76), which assumes a rigid body:

$$A_1^{S1} = C_B^{S1}((A^B_{\mathrm{CG}} + ([\omega^B_{IB}\times][\omega^B_{IB}\times] + [\dot{\omega}^B_{IB_{\mathrm{CG}}}\times])L^B_{\mathrm{CG\,to\,1}})) \tag{76}$$

The acceleration at sensor 1 in the sensor frame is defined by A_1^{S1}. The known rotation from the body frame to the sensor frame is given by C_B^{S1} and accounts for a known installation relative to the body frame. The lever arm from the CG to the sensor is referred to as $L^B_{\mathrm{CG\,to\,1}}$ and is defined in the body frame. The linear acceleration at the CG is given by A^B_{CG}, and the angular acceleration at the CG is given by $\dot{\omega}^B_{IB_{\mathrm{CG}}}$. The angular velocity is constant across the rigid body and is described by ω^B_{IB}. From this equation, it is clear that the resulting acceleration is a function of the centripetal acceleration and angular acceleration operating through the lever arm.

The acceleration at the sensor may be predicted using a combination of existing sensors and knowledge of the thrust and momentum wheel commands. An estimate of the acceleration at a particular sensor $\bar{A}_1^{S1}$ is given by Eq. (77). The measured angular velocity from the gyros $\tilde{\omega}^B_{IB}$ is directly introduced to provide the centripetal acceleration term. The term w_1^{S1} represents the combined noise, which will be defined later:

$$\bar{A}_1^{S1} = C_B^{S1}(\bar{A}^B_{\mathrm{CG}} + ([\tilde{\omega}^B_{IB}\times][\tilde{\omega}^B_{IB}\times] + [\dot{\tilde{\omega}}^B_{IB_{\mathrm{CG}}}\times])L^B_{\mathrm{CG\,to\,1}}) + w_1^{S1} \tag{77}$$

The acceleration at the CG $\bar{A}^B_{\mathrm{CG}}$ is determined by the sum of the thruster inputs u^B_{Ti}, as shown in Eq. (78):

$$\bar{A}^{\,B}_{\mathrm{CG}} = \frac{1}{m}\sum_{i=1}^{N} B_{Ti}\bar{u}^B_{Ti} \tag{78}$$

The input moment estimates $\dot{\bar{\omega}}^B_{IB_{\mathrm{CG}}}$ are calculated as the sum of the thruster and momentum wheel command inputs:

$$\dot{\bar{\omega}}^{\,B}_{IB_{\mathrm{CG}}} = I_B^{-1}\left(\sum_{i=1}^{N} B_{Tri}\bar{u}^B_{Ti} + \sum_{j=1}^{S} B_{Mj}\bar{u}^B_{Mj}\right) \tag{79}$$

To assess the combined uncertainty in $\bar{A}_1^{S1}$, the measurement models from Eqs. (47–49) are substituted into Eq. (77). A first-order perturbation technique is taken and noise terms of higher-order than one are discarded. The result is given in Eq. (80):

$$\begin{aligned} w_1^{S1} = \upsilon_{S1} + C_B^{S1}\Bigg(&\frac{1}{m}\sum_{i=1}^{N} B_{Ti}w_{Ti} - [L^B_{\mathrm{CG\,to\,1}}\times]I_B^{-1} \\ &\times\left(\sum_{i=1}^{N} B_{Tri}w_{Ti} + \sum_{j=1}^{S} B_{Mj}w_{Mj}\right) - [\tilde{\omega}^B_{IB}\times][L^B_{\mathrm{CG\,to\,1}}\times]\upsilon_\omega \\ &- [(\tilde{\omega}^B_{IB}\times L^B_{\mathrm{CG\,to\,1}})\times]\upsilon_\omega\Bigg) \end{aligned} \tag{80}$$

A residual is formed at each accelerometer between the predicted and measured accelerations. This test residual will be used to identify the faults. The residual is composed of nine measurements:

$$\tilde{r}_A = \begin{bmatrix} \tilde{A}_1^{S1} - \bar{A}_1^{S1} \\ \tilde{A}_2^{S2} - \bar{A}_2^{S2} \\ \tilde{A}_3^{S3} - \bar{A}_3^{S3} \end{bmatrix} = \begin{bmatrix} \upsilon_{a1} - w_1^{S1} \\ \upsilon_{a2} - w_2^{S2} \\ \upsilon_{a3} - w_3^{S3} \end{bmatrix} \tag{81}$$

The residual has the following anticipated statistics, assuming that no fault exists:

$$E[\tilde{r}_A] = 0 \tag{82}$$

$$E[\tilde{r}_A\tilde{r}_A^T] = R_A \tag{83}$$

The matrix R_A is defined as follows with the subsequent definitions on the covariance of the input noise terms:

$$R_A = \begin{bmatrix} V_{A1} + W_{S1} & -W_{S1S2} & -W_{S1S3} \\ -W_{S1S2} & V_{A2} + W_{S2} & -W_{S2S3} \\ -W_{S1S3} & -W_{S2S3} & V_{A3} + W_{S3} \end{bmatrix} \tag{84}$$

$$\begin{aligned} E[\upsilon_{a1}\upsilon_{a1}^T] = V_{A1}; \quad E[\upsilon_{a2}\upsilon_{a2}^T] = V_{A2}; \quad E[\upsilon_{a3}\upsilon_{a3}^T] = V_{A3} \\ E[w_1^{S1}(w_1^{S1})^T] = W_{S1}; \quad E[w_2^{S2}(w_2^{S2})^T] = W_{S2} \\ E[w_3^{S3}(w_3^{S3})^T] = W_{S3}; \quad E[w_1^{S1}(w_3^{S3})^T] = W_{S1S3} \\ E[w_1^{S1}(w_2^{S2})^T] = W_{S1S2}; \quad E[w_3^{S3}(w_2^{S2})^T] = W_{S3S2} \end{aligned} \tag{85}$$

A. Thruster Fault Model

If a thruster fault occurs, then the following thruster fault model is assumed:

$$\bar{u}^{\,B}_{Ti} = u^B_{Ti} + w_{Ti} + \mu_{Ti} \tag{86}$$

In this case, the unknown fault signal μ_{Ti} is introduced into the ith fault. The unknown fault signal is propagated through the prediction of the acceleration at the sensor. A fault direction matrix F_{Ti} is defined for how this particular fault in this particular thruster affects the residual prediction. After substituting the fault model of Eq. (86) into Eq. (77), the fault model at accelerometer 1 is given in Eq. (87), in which the lever arm to the accelerometer is cross-multiplied by the direction and momentum directions. Note that the sign of the fault direction matrix is relevant because the construction of the mapping function relative to the unknown fault signal assumes that μ_{Ti} may take on positive or negative values:

$$F_{Ti1} = C_B^{S1}[L^B_{\mathrm{CG\,to\,1}}\times]\left(\frac{1}{m}B_{Ti} + (I^B_{BB})^{-1}(B_{Tri})\right) \tag{87}$$

The complete fault direction F_{Ti} using all three accelerometers is the combination of each of the fault directions at each sensor, as shown in Eq. (88):

$$F_{Ti} = \begin{bmatrix} F_{Ti1} \\ F_{Ti2} \\ F_{Ti3} \end{bmatrix} \tag{88}$$

The residual of Eq. (81) is modified to include the fault in this particular thruster:

$$\tilde{r} = \begin{bmatrix} \tilde{A}_1^{S1} - \bar{A}_1^{S1} \\ \tilde{A}_2^{S2} - \bar{A}_2^{S2} \\ \tilde{A}_3^{S3} - \bar{A}_3^{S3} \end{bmatrix} = \begin{bmatrix} \upsilon_{a1} - w_1^{S1} \\ \upsilon_{a2} - w_2^{S2} \\ \upsilon_{a3} - w_3^{S3} \end{bmatrix} + F_{Ti}\mu_{Ti} \tag{89}$$

A projector H_{Ti} can be generated such that $H_{Ti}F_{Ti} = 0$ using a standard annihilator generation process, as defined in Eq. (90). This annihilator for a single fault exists for the rank 1 fault direction. It is unique relative to other thruster faults provided that the fault direction is unique, which is defined by the unique lever arm in Eq. (87):

$$H_{Ti} = F_{Ti} - F_{Ti}(F_{Ti}^T F_{Ti})^{-1} F_{Ti}^T \tag{90}$$

The projected residual will not see a fault in this particular direction and will be zero-mean with standard deviation defined by the noise processes:

$$\tilde{r}_{Ti} = H_{Ti}\tilde{r} = H_{Ti}\left(\begin{bmatrix} \upsilon_{a1} - w_1^{S1} \\ \upsilon_{a2} - w_2^{S2} \\ \upsilon_{a3} - w_3^{S3} \end{bmatrix} + F_{Ti}\mu_{Ti}\right)$$
$$= H_{Ti}\begin{bmatrix} \upsilon_{a1} - w_1^{S1} \\ \upsilon_{a2} - w_2^{S2} \\ \upsilon_{a3} - w_3^{S3} \end{bmatrix} \tag{91}$$

No projector is required because all of the fault directions are unique. Instead of Eq. (91), the full residual $\tilde{r}$ will be used and the adaptive bias will be estimated according to the unique fault direction F_{Ti} for each thruster.

It is noted that only 8 of the 16 thrusters may be tested using this method. Every thruster in the configuration of Fig. 2 has a partner thruster that provides force and moment in exactly the opposite direction. This test cannot distinguish between faults in either thruster, only in the thruster pair. Once the thruster pair is identified, the particular fault can be easily identified by simply examining the sign of the residual error along the direction of the thruster. Table 2 lists the thruster pair combinations for the configuration in Fig. 2.

For an arbitrary configuration of thrusters, the resulting fault directions can be generated. A rank test determines if sufficient information exists to isolate the faults. The rank test is described in Eq. (92), where n is the total number of thrusters and the matrix F_T is tested for row rank:

$$\text{rank}\,(F_T) > n \qquad F_T = [\,F_{T1} \quad F_{T2} \quad \cdots \quad F_{Tn}\,] \tag{92}$$

If more thrusters are required for implementation, then more accelerometers at different locations may be required to correctly identify thruster faults.

B. Momentum Wheel Faults

The present configuration of accelerometers is interesting because it is possible to simultaneously detect and isolate momentum wheel faults from thruster faults. For a momentum wheel fault, the fault model is defined in Eq. (93), where the fault signal μ_{Mj} is unknown and represents a fault in the momentum wheel:

$$\tilde{u}_{Mj}^B = u_{Mj}^B + w_{Mj} + \mu_{Mj} \tag{93}$$

Table 2 Thruster pairings in Fig. 1

Thruster ID	Opposing thruster
1	3
2	4
5	7
6	8
9	11
10	12
13	15
14	16

Substituting Eq. (93) into Eq. (77), a fault direction matrix F_{Mj} may be constructed that describes how μ_{Mj} affects the measurements. In this case, the fault direction is defined as follows:

$$F_{Mj1} = -[L_{\text{CG to 1}}^B \times] I_B^{-1} B_{Mj} \tag{94}$$

The full fault matrix is formed by combining the fault matrix from each accelerometer:

$$F_{Mj} = \begin{bmatrix} F_{Mj1} \\ F_{Mj2} \\ F_{Mj3} \end{bmatrix} \tag{95}$$

As with the thruster module, the effect of the fault model and direction influences the residual:

$$\tilde{r} = \begin{bmatrix} \tilde{A}_1^{S1} - \bar{A}_1^{S1} \\ \tilde{A}_2^{S2} - \bar{A}_2^{S2} \\ \tilde{A}_3^{S3} - \bar{A}_3^{S3} \end{bmatrix} = \begin{bmatrix} \upsilon_{a1} - w_1^{S1} \\ \upsilon_{a2} - w_2^{S2} \\ \upsilon_{a3} - w_3^{S3} \end{bmatrix} + F_{Mj}\mu_{Mj} \tag{96}$$

A projector is constructed to annihilate the effect of this direction on the residual such that $H_{Mj}F_{Mj} = 0$:

$$H_{Mj} = F_{Mj} - F_{Mj}(F_{Mj}^T F_{Mj})^{-1} F_{Mj}^T \tag{97}$$

The projected residual will be immune to the effect of this fault and remain zero mean:

$$\tilde{r}_{Mj} = H_{Mj}\tilde{r} = H_{Mj}\left(\begin{bmatrix} \upsilon_{a1} - w_1^{S1} \\ \upsilon_{a2} - w_2^{S2} \\ \upsilon_{a3} - w_3^{S3} \end{bmatrix} + F_{Mj}\mu_{Mj}\right)$$
$$= H_{Mj}\begin{bmatrix} \upsilon_{a1} - w_1^{S1} \\ \upsilon_{a2} - w_2^{S2} \\ \upsilon_{a3} - w_3^{S3} \end{bmatrix} \tag{98}$$

The fault direction F_{Mj} is stored and will be used to initialize the adaptive bias scheme.

C. Accelerometer Fault Models

Of course, one of the accelerometer measurements may fail. In this case, a fault in one sensor in one accelerometer triad is hypothesized. The fault model is described in Eq. (99), where μ_{Ak} is a scalar and represents the unknown fault signal in the accelerometer triad for the kth accelerometer triad for each axis $l = 1, 2, 3$ for the x, y, and z axes:

$$\tilde{a}_k = a_k + \upsilon_k + F_{Akl}\mu_{Akl} \tag{99}$$

The fault direction F_{Akl} is defined as simply a 3×1 vector with a 1 on the axis that has failed. A projector may be constructed to remove the one residual affected by the fault, although the simplified fault mode of these accelerometers does not require annihilation. Instead, only the fault direction F_{Akl} is stored and the adaptive bias scheme will estimate the probability of a fault, assuming that the fault model F_{Akl} is used.

D. Gyro Faults

Gyro measurements are required as part of this test. In this case, the fault model is defined as in Eq. (100), where the index l denotes an index on the particular axis that failed:

$$\tilde{\omega}_{IB}^B = \omega_{IB}^B + \upsilon_\omega + F_{\omega l}\mu_{\omega l} \tag{100}$$

Substituting Eq. (100) into Eq. (77) to determine the fault matrix produces a complex result. The total fault direction $F_{A\omega l}$ is defined in Eq. (101):

$$F_{A\omega l} = [\tilde{\omega}_{IB}^B \times][L_{\text{CG to 1}}^B \times] - [(\tilde{\omega}_{IB}^B \times L_{\text{CG to 1}}^B)\times] \tag{101}$$

Note that the fault model includes terms depending on the possibly faulty measurement. The existence of this type of fault mode in which the fault direction is time-varying and dependent upon the faulted measurement is difficult to use. Therefore, this test is not recommended to find gyro faults and instead it is recommended to use the fault detection filter of Eqs. (68) and (69) designed for gyro faults. Although gyros are essential to detecting faults in the thrusters, momentum wheels, and accelerometers, the gyro faults are rapidly absorbed into the test residual and cannot be detected using these test residuals. Interestingly, for small gyro faults, the test residuals generated are largely immune to the fault and no dependence on the gyro faults need be modeled.

E. Combined Faults and False Alarm

In actual implementation, it is possible to combine all of the residuals for the accelerometer faults, thruster faults, and momentum wheel faults into a single MHSSPRT. This method actually works better than separating out the tests, simply because the correct fault mode is in the combined test and is easily predicted.

On the surface, it may seem that this process may not work, because essentially nine measurements of acceleration are used to correctly detect and identify up to 8 thruster faults combined with 3 momentum wheel faults and 9 acceleration faults. For each hypothesized fault, the same scalar fault shows up in all 9 sensors through a very unique direction defined by the lever arms. Only the residual assuming the hypothesis that completely annihilates this direction will remain zero mean, enabling easy identification, as shown in the next section.

VIII. MHSSPRT with Adaptive Bias Estimation

The previous sections have defined the test residuals and associated statistics. A total of 37 different faults have been identified: 6 star tracker faults, 3 gyro faults, 16 thruster faults, 9 accelerometer faults, and 3 momentum wheel faults, although only 8 thrusters are modeled because the other 8 represent the same force and moment with opposite sign. The test residual to be used for fault detection is defined in Eq. (102) using the two fault detection filter residuals $\tilde{r}_{\theta 1}$ and $\tilde{r}_{\theta 2}$, the star tracker parity residual $\tilde{r}_{\theta 12}$, with all three defined in Eq. (74). The 9×1 test residual of three accelerometer measurements $\tilde{r}_A$ defined in Eq. (81). The total test residual size is 18×1. The associated covariance Λ is defined in Eq. (103). It is assumed that the state estimates of the fault detection filters are uncorrelated with the accelerometer residuals that used gyro measurements for convenience and simplicity:

$$\tilde{r}_T = \begin{bmatrix} \tilde{r}_{\theta 1} \\ \tilde{r}_{\theta 2} \\ \tilde{r}_{\theta 12} \\ \tilde{r}_A \end{bmatrix} \tag{102}$$

$$\Lambda = \begin{bmatrix} C_\theta M_{\theta 1} C_\theta^T + V & 0 & V_1 & 0 \\ 0 & C_\theta M_{\theta 2} C_\theta^T + V_2 & -V_2 & 0 \\ V_1 & -V_2 & V_2 + V & 0 \\ 0 & 0 & 0 & R_A \end{bmatrix} \tag{103}$$

The mapping function κ is defined using the associated fault directions and test residuals, as shown in Eq. (104), in which submatrices are defined for each fault signal. The map for the first star tracker is defined in Eq. (105) and the map for the second star tracker and gyro are in Eqs. (106) and (107), respectively. Note that the gyro fault impacts both star tracker fault detection filter residuals, providing a unique fault matrix:

$$\kappa = [\kappa_{\theta 1} \quad \kappa_{\theta 2} \quad \kappa_\omega \quad \kappa_A \quad \kappa_{th} \quad \kappa_{MW}] \tag{104}$$

$$\kappa_{\theta 1} = \begin{bmatrix} C_\theta f_{\theta 1} \\ 0_{3\times 3} \\ C_{S1}^B \\ 0_{9\times 3} \end{bmatrix} \tag{105}$$

$$\kappa_{\theta 2} = \begin{bmatrix} 0_{3\times 3} \\ C_\theta f_{\theta 2} \\ C_{S2}^B \\ 0_{9\times 3} \end{bmatrix} \tag{106}$$

$$\kappa_\omega = \begin{bmatrix} C_\theta f_\omega \\ C_\theta f_\omega \\ 0_{3\times 3} \\ 0_{9\times 3} \end{bmatrix} \tag{107}$$

$$\kappa_A = \begin{bmatrix} 0_{9\times 9} \\ I_{9\times 9} \end{bmatrix} \tag{108}$$

$$\kappa_{\text{th}} = \begin{bmatrix} 0_{9\times 1} & 0_{9\times 1} & \cdots & 0_{9\times 1} \\ F_{T1} & F_{T2} & \cdots & F_{T8} \end{bmatrix} \tag{109}$$

$$\kappa_{MW} = \begin{bmatrix} 0_{9\times 1} & 0_{9\times 1} & 0_{9\times 1} \\ F_{M1} & F_{M2} & F_{M3} \end{bmatrix} \tag{110}$$

The fault map for the accelerometers is simple and given in Eq. (108). The map for the 8 unique thruster directions is given in Eq. (109).

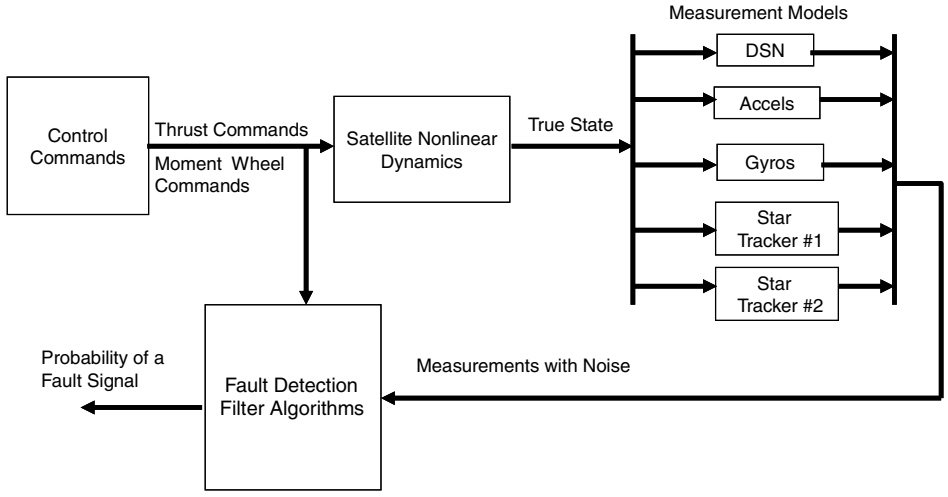

Fig. 4 Simulation block diagram.

Table 3 Sensor noise parameters (1σ)

Sensor	Value	Unit
DSN	1500	m
Star tracker transverse axis	4	arcsec
Star tracker boresight axis	24	arcsec
Rate gyros	0.15	deg/h
Accelerometers	1.00E − 07	m/s/s

Table 4 Actuator uncertainties (1σ)

Actuator	Value	Unit
Thrusters	1.00E − 06	N
Momentum wheels	1.50E − 05	Nm

Table 5 Initial uncertainty values

State	Value	Unit
Position	2000	m
Velocity	1	m/s
Attitude	1	deg
Angular rate	0.01	deg/s

Likewise, the map for the unique momentum wheel faults is given in Eq. (110).

Processing of the MHSSPRT proceeds as described previously. For a total of 29 unique faults, a set of 59 hypotheses is generated: 29 for a positive fault for each unique fault, 29 for a negative fault for each unique fault, and one healthy hypothesis. The positive and negative baseline (threshold) biases for each hypothesis are calculated using Eq. (23). A unique bias $\bar{b}_i$ is hypothesized for each fault mode. At each time step, all of the residuals are generated. The biases are updated using Eqs. (24) and (25) and checked against the threshold on an element-by-element basis. The density function for each hypothesis is calculated using Eq. (22). The probability of a fault is then calculated using the standard MHSSPRT update and propagation methods in Eqs. (16–18). The next section presents simulation results using the filters and processors developed.

IX. Simulation Results

The methods presented are implemented in simulation and tested. Figure 4 shows the simulation block diagram. The control law, which generated only simple maneuvers for this case, is used to generate the forces and moments acting on the vehicle, which are then integrated through the dynamics of the vehicle. The output state is then transformed into the measurements and corrupted by the noise models presented. The outputs of the instruments are passed to the fault detection and isolation software block. This block also receives the commanded momentum wheel and thruster commands. Tables 3 and 4 show the 1-sigma values used for the thruster and momentum wheel noises, respectively, and Table 5 shows the 1-sigma values used to generate the initial error covariance for each filter structure.

A. Test Maneuver

A short test is performed to demonstrate the effectiveness of this filter structure. The thrusters and momentum wheels specified for the TPF-type program used 1 mN thrusters to perform station keeping within satellite clusters. The simulation test includes a short thrust of thruster 1 of 0.001 N for 2 s. The thrust is turned off for 2 s. Then thruster 3 is fired with a force of 0.001 N for 2 s, essentially reversing the motion. After this burst, the X-axis momentum wheel is activated, providing a positive moment of 1.0e − 3 Nm for 2 s followed by 2 s of no momentum wheel commands. A final 2 s command of −1.0e − 3 Nm reverses the system. Note that the system does not achieve steady state and that an attitude and velocity change has occurred during the maneuver. All data are simulated using an integration step size of 100 Hz. All of the estimation algorithms and fault detection filters operate on measurements processed at 10 Hz. A 12 s simulation is generated. For each fault type described in the next section, the fault is introduced 2 s into the simulation. Because the accelerometer residuals are used to examine thruster, momentum wheel, and accelerometer faults, only the accelerometer residuals are shown for that type of fault. Likewise, only the star tracker residuals are shown when examining the star tracker and gyro faults, even though both are generated simultaneously. Similarly, only a subset of the MHSSPRT probabilities generated is displayed for any fault set even though all of the probabilities were calculated using a single MHSSPRT algorithm. A detection threshold of 99.9% is required to declare a fault, and a 99.9% certainty in the probability of a fault is required for isolation. A probability of 1.0e − 6 per epoch is injected into the MHSSPRT for each fault, and it is assumed the system starts in the healthy configuration.

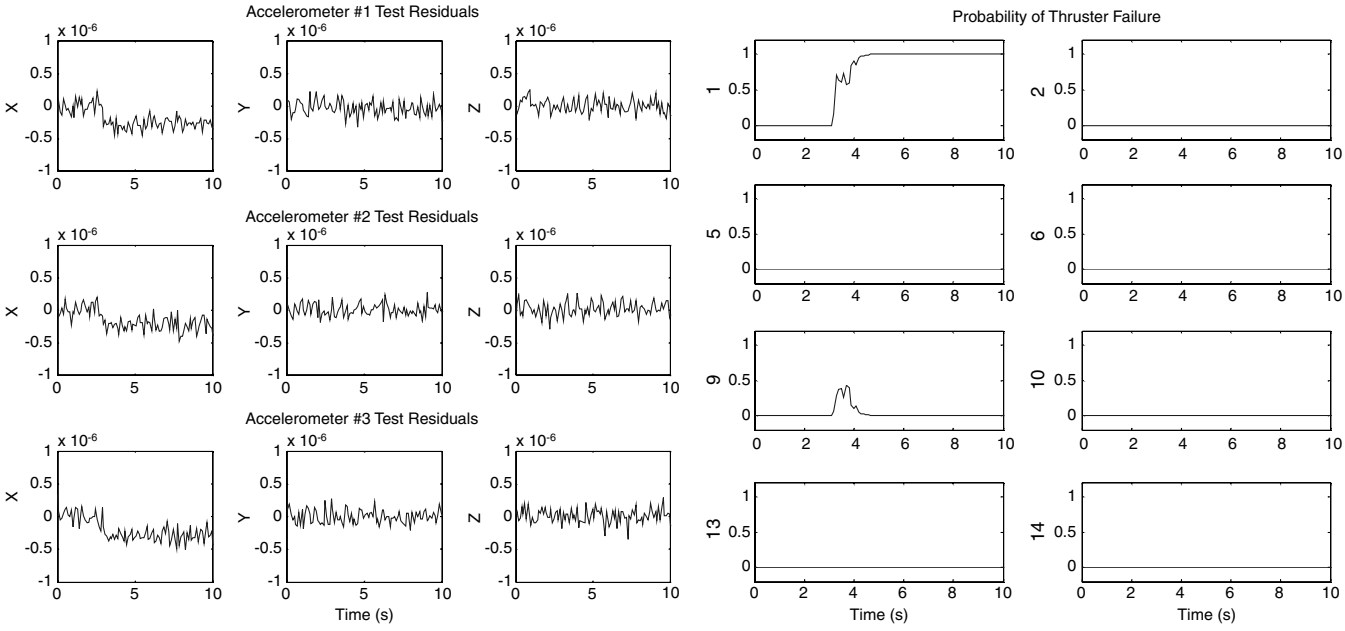

Fig. 5 Accelerometer residuals and MHSSPRT probabilities for a fault in thruster 1.

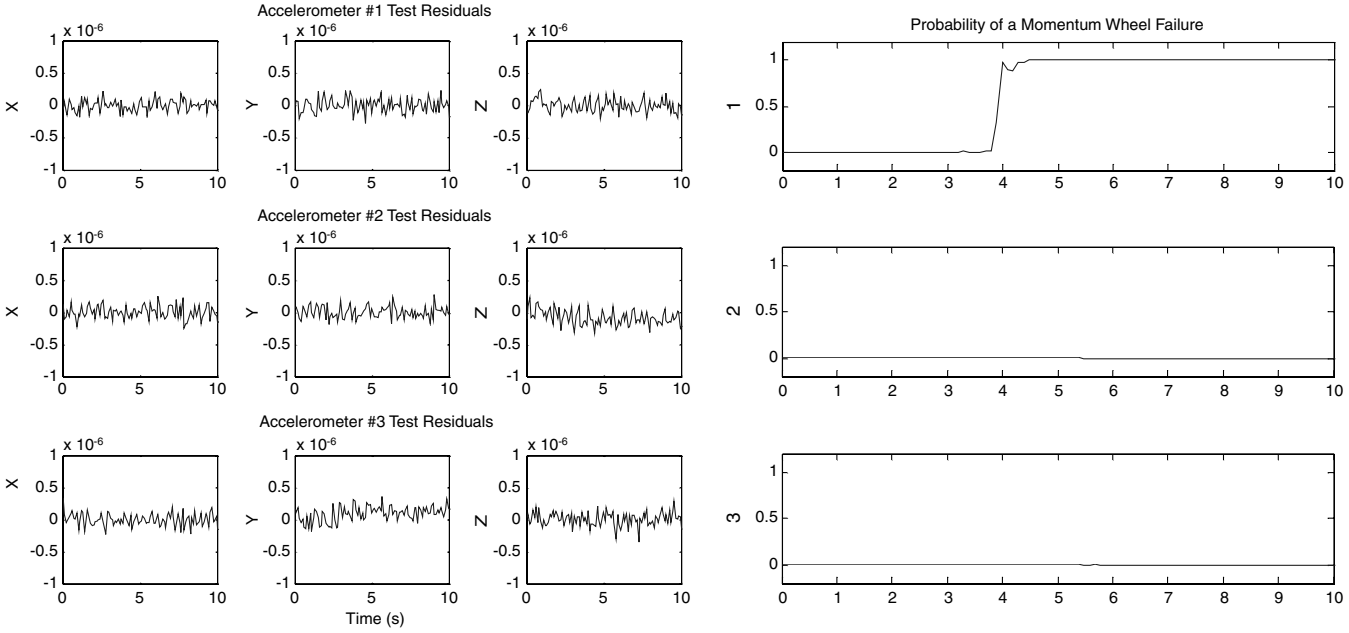

Fig. 6 Accelerometer residuals and MHSSPRT probabilities for a fault in the *X*- axis momentum wheel.

B. Thruster Faults

A 0.05 mN thruster fault is introduced at the 2 s mark into thruster 1. Essentially, thrust is commanded, but the vehicle does not respond. Figure 5 shows the 9 accelerometer test residuals as well as the MHSSPRT probabilities of a thruster fault for 8 of the thrusters. The fault jump is clearly seen in the residuals. The fault is detected in 1.1 s and identified in 2.0 s after the fault is injected. This fault size would have injected a 0.05 mN meter torque on the vehicle assuming a 1.0 m lever arm. Note that initially, the probability that a fault occurs in thruster 9 is indicated as well as thruster 1, but that the MHSSPRT detects and isolates correctly over time.

C. Momentum Wheel Faults

Using the same simulation, a momentum wheel fault of 7.5e − 4 Nm is introduced in the *x* axis momentum wheel at 2 s into the simulation. Again, the momentum wheel is commanded, but does not physically respond, indicating a fault. Figure 6 shows the residual from the three triaxial accelerometers and the resulting MHSSPRT output examining the momentum wheel faults. The step in the momentum wheel is barely observable in the third accelerometer in the *z* axis and in the second accelerometer *y* axis. The fault is detected in 1.2 s and isolated to the *x*-axis momentum wheel in 2.5 s after the fault is injected.

D. Accelerometer Faults

Finally, a 5-sigma accelerometer bias is introduced into the *X* axis of accelerometer 1. This size of the fault is 5 times greater than the standard deviation of noise used on the accelerometers. Figure 7 displays the MHSSPRT probability results. This plot shows that detection of the fault takes 1.0 s or 10 epochs to declare a fault and 1.2 s or 12 epochs to isolate.

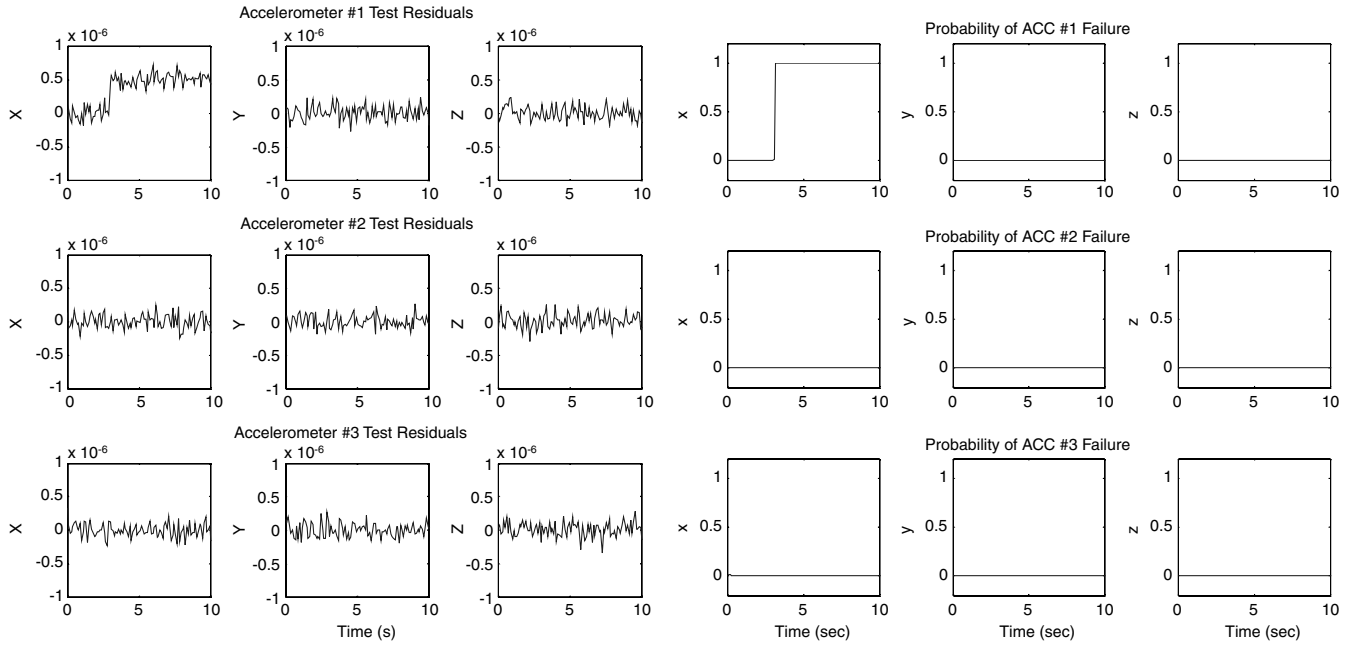

Fig. 7 Accelerometer residuals and MHSSPRT probabilities for a fault in the *X*-axis accelerometer 1.

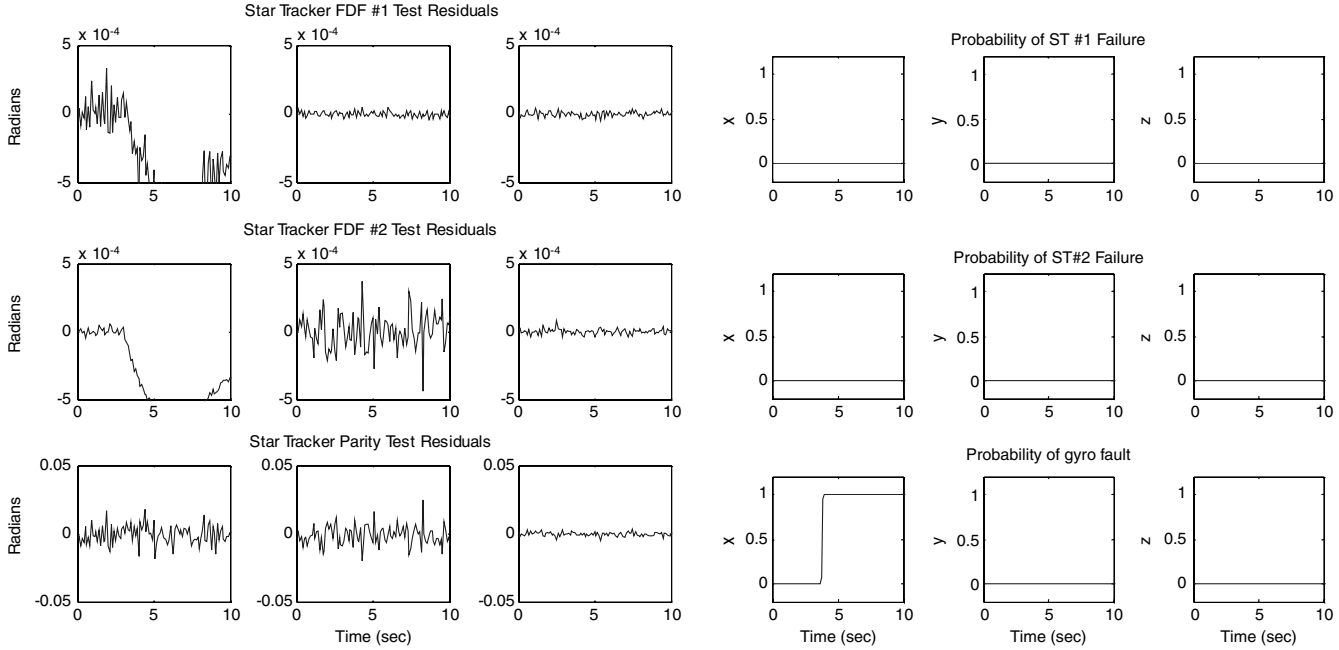

Fig. 8 Star tracker (ST) residuals and MHSSPRT probabilities for a fault in the *X* axis of the gyro.

E. Gyro Faults

The fault detection filters described were implemented in software and tested using the same scenario. A gyro fault consisting of a 5-sigma step function relative to the gyro measurement noise is introduced after 2 s. The fault enters through the *x*-axis gyro. Figure 8 shows the residuals from the two star tracker residuals as well as the star tracker parity test. The first star tracker residual tests only uses star tracker 1, with the gyros using the fault detection filter structure defined in Sec. VI. Two points are readily apparent. First, the gyro fault appears in the residual of both star tracker residuals as expected, but not in the star tracker parity test. Second, the noise level is different in the different residuals, due to the fact that one star tracker is orientated with a 90 deg offset from the other. The MHSSPRT probabilities detect the fault in 1.7 s and isolate the correct gyro fault 2.0 s after the fault is injected.

F. Star Tracker Faults

A fault in the *x* axis of star tracker 1 is introduced at 2 s. The *x* axis represents the boresight axis of this star tracker, which has the least accuracy. A 5-sigma fault bias is introduced. Note that the second star tracker is orientated such that it has a 90 deg offset and the two star trackers do not have a common boresight. Figure 9 shows the processed residuals and the associated probabilities. The residual shows the fault in both the star tracker 1 residual and the parity relationship. Over time, the fault is absorbed into the star tracker 1 filter. The MHSSPRT reflects this fact, because it initially detects and isolates the fault to the correct sensor and axis even though the fault signal is transient. Over time, the fault detection filter can no longer be used to detect the star tracker bias and distinguish which star tracker has failed. The filter structure detects the fault in 1.1 s and isolates at 1.9 s after the fault is injected. The fault signal lasts 2.8 s

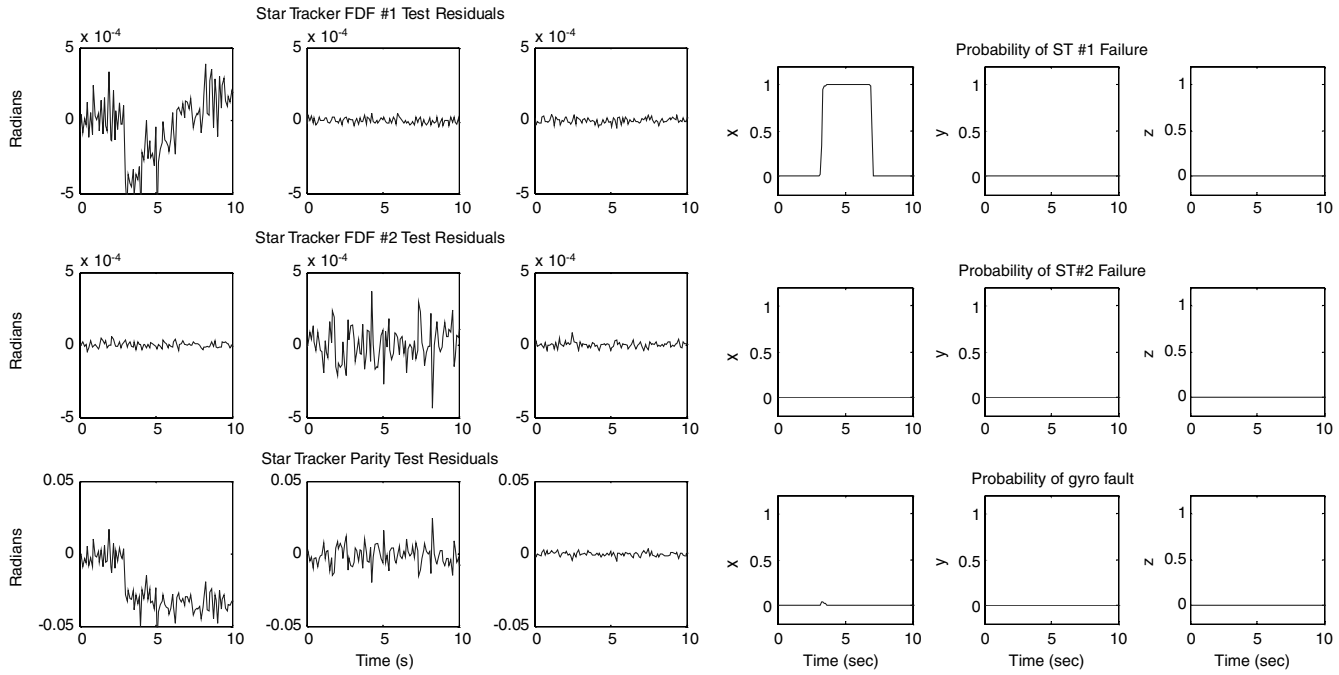

Fig. 9 Star tracker residuals and MHSSPRT probabilities for a fault in the *X* axis of star tracker 1.

before dissipating, which would be sufficient to correctly identify and generate an appropriate response.

X. Conclusions

A new method for fault detection of the attitude control system is presented. The system has shown the ability to detect faults in command of thrust and momentum rapidly and accurately, as well as faults in the star tracker, gyros, and additional accelerometers. The problem of fault detection is shown to decouple into two nearly independent problems. In the first, a set of fault detection filters is constructed to detect and isolate star tracker and gyro faults. The second system uses a set of accelerometers to detect momentum wheel, thruster, and accelerometer faults. Because the second system is dependent on rate gyro measurements, but unable to detect gyro faults, the fault detection method must be used to eliminate the possibility of gyro faults before the parity tests may be used. Using the method of combining a fault mapping along with the adaptive estimation of the fault signal enables a single MHSSPRT test to act as the decision-maker for the fault detection system. The MHSSPRT assumes that only one fault occurs at a time and detects the appropriate fault, given the fault map and adaptive estimation scheme.

The method presented here moves fault detection technology closer to an online, reconfigurable fault detection scheme based on stochastic uncertainty. Each sensor fault is explicitly mapped to an available residual through the mapping function. Because no assumptions on the fault signal are made, the mapping function is generic and shows how each fault affects all of the available residuals. If a sensor has a fault and it is detected, it may be removed and replaced by a different sensor of similar or possibly lesser quality. The resulting mapping function could be updated and the associated adaptive scheme could be restarted with new bias estimates and assumed uncertainties based on the assumed healthy performance of the sensor fusion processes using the new sensor. In this way, reconfiguration may be automated based on the latest calibrated information and continue to provide fault detection information on the remaining sensors.

References

[1] Martin, S. R., Scharf, D., Wirz, R., Lay, O., McKinstry, D., Mennesson, B., Purcell, G., Rodriguez, J., Scheerr, L., Smith, J., and Wayne, L., "TPF-Emma: Concept Study of a Planet Finding Space Interferometer," *Proceedings of SPIE: The International Society for Optical Engineering*, Vol. 6693, 2007, Paper 669309. doi:10.1117/12.734835

[2] Smith, R. S., and Hadaegh, F. Y., "Closed-Loop Dynamics of Cooperative Vehicle Formations with Parallel Estimators and Communications," *IEEE Transactions on Automatic Control*, Vol. 52, No. 8, Aug. 2007, pp. 1404–1414. doi:10.1109/TAC.2007.902735

[3] Scharf, D. P., Hadaegh, F. Y., Rahman, Z. H., Shields, J. F., Singh, G., and Wette, M. R., "An Overview of the Formation and Attitude Control System for the Terrestrial Planet Finder Formation Flying Interferometer," International Symposium on Formation Flying Missions and Technologies, Washington, D.C., Jet Propulsion Lab. Paper 40212, Sept. 2004, http://hdl.handle.net/2014/40212.

[4] Chen, R. H., Ng, H. K., Speyer, J. L., Guntur, L. S., and Carpenter, R., "Health Monitoring of a Satellite System," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 3, 2006, pp. 593–604. doi:10.2514/1.15012

[5] Lee, A., and Brown, M., "A Model-Based Thruster Leakage Monitor for the Cassini Spacecraft," *Proceedings of the American Control Conference*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, June 1998, pp. 902–904.

[6] Chung, W., and Speyer, J. L., "A Game Theoretic Fault Detection Filter," *IEEE Transactions on Automatic Control*, Vol. 43, No. 2, Feb. 1998, pp. 141–161.

[7] Malladi, D., and Speyer, J., "A Generalized Shiryayev Sequential Probability Ratio Test for Change Detection and Isolation," *IEEE Transactions on Automatic Control*, Vol. 44, No. 8, Aug. 1999, pp. 1522–1534. doi:10.1109/9.780416

[8] Chan, S., and Speyer, J. L., "A Sequential Probability Test for RAIM," *Proceedings of the 17th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2004)*, Inst. of Navigation, Manassas, VA, Sept. 2005, pp. 1798–1802.

[9] Mutuel, L. H., and Speyer, J. L., "Fault Tolerant Estimator Design for a UAV," 2000 AIAA Guidance, Navigation, and Control Conference, Denver, CO, AIAA Paper 2000-4464, Aug. 2000.

[10] Malladi, D., and Speyer, J. L., "A New Approach to Multiple Model Adaptive Estimation," *Proceedings of the 36th IEEE Conference on Decision and Control*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, 1997, pp. 3460–3467.

[11] Zipfel, P. H., *Modeling and Simulation of Aerospace Vehicle Dynamics*, AIAA Education Series, AIAA, Reston, VA, 2000.

[12] Maybeck, P. S., *Stochastic Models, Estimation, and Control*, Vol. 1, NavtechGPS, Springfield, VA, 1994.