AIAA GN&C Conference, August 2007, Hilton Head, SC

Early Detection of Solid Rocket Motor Case-Breach Failure*

Dimitry Gorinevsky[†]

Mitek Analytics LLC, Palo Alto, CA 94306 and Stanford University, Stanford, CA 94305

Robert W. Mah[‡] and Dogan Timucin[§]

NASA Ames Research Center, Moffett Field, CA 94035, USA

This paper describes work on early detection of space launch vehicle failures that could lead to loss of control during ascent. The particular focus is on the Solid Rocket Motor (SRM) comprising First Stage of the Crew Launch Vehicle (CLV). Some of the most important failure modes leading to loss of vehicle control are caused by hot gases ejected from the SRM combustion chamber through a case breach, through a joint between SRM segments, through a nozzle joint, or through an igniter seal. These hot gases create lateral thrust augmentation, with the resulting tilting moment possibly leading to loss of control. By using Thrust Vector Control (TVC) gimballing of the main nozzle, the flight control system tries to counter any attitude disturbance, which could mask the moment augmentation till it overwhelms the TVC and the control authority is lost. The approach proposed and demonstrated in a medium-fidelity simulation of the rocket dynamics employs continuous model-based multivariable estimation and monitoring of the augmentation forces and moments for early detection of the failure. The computations are performed in two stages. First, a 6-degree-of-freedom model of vehicle dynamics is used to compute the predicted vehicle accelerations. Second, the differences with the observed accelerations – model prediction residuals – are used in optimal multivariable algorithms to estimate the thrust augmentation. A case breach failure may thus be detected well before its effect on trajectory becomes noticeable in the raw data.

[*] This work was supported by NASA Exploration Systems Mission Directorate Exploration Technology Development Program, Integrated System Health Management project, through Mitek Analytics research subaward #8020-011, USRA prime contract #NCC2-1426.

[†] Principal, Mitek Analytics LLC, 281 El Verano Ave., Palo Alto, CA 94306, USA, email: dimitry@mitekan.com; Consulting Professor of Electrical Engineering, Stanford University, Stanford, CA 94305, email: gorin@stanford.edu

[‡] Senior Scientist, NASA Ames Research Center, Moffett Field, CA 94035

[§] Senior Scientist, Modeling, Learning, and Control Group Lead, NASA Ames Research Center, Moffett Field, CA 94035

Nomenclature

T_A	Thrust augmentation force, N
d_A	Axial (z-axis) distance from the CG to the breach location, m
$\bar{v}$	Instanteneous linear velocity in the attached axes, m/s
$\bar{\omega}$	Instantaneous angular rate in the attached axes, rad/s
p, r, q	Instantaneous angular rate vector components (roll rate, yaw rate, pitch rate)
m	Vehicle mass, kg
I	Vehicle tensor of intertia, kg m^2
D_{CG}	Vector pointing from the vehicle CG to the TVC gimbal center
$\bar{F}_{THRUST}$	Vector of the propulsion thrust force assuming normal operation, N
$\bar{F}_{AERO}$	Aerodynamic force vector, N
$\bar{F}_{AUGMENT}$	Thrust augmentatation force vector (due to a fault), N
$\bar{M}_{THRUST}$	Nominal vector of the propulsion thrust moment, Nm
$\bar{M}_{AERO}$	Vector of aerodynamic moment, Nm
$\bar{M}_{AUGMENT}$	Moment vector created by the thrust augmentatation force, N
T	Nominal propulsion thrust value, N
M_z	Pitching moment, Nm
M_y	Yawing moment, Nm
A	Vehicle cross-section area, m^2
D	Vehicle characteristic diameter
ρ	Air density, kg/m^3
V	Airspeed, m/s
c_k	Nondimensional aerodynamic force coefficients, indexed by $k = \{x, y, z\}$
$c_{\alpha,k}$	Nondimensional aerodynamic moment coefficients, indexed by $k = \{x, y, z\}$
$\bar{r}$	Prediction residual vector
X	Fault state vector
Y	Fault observation data vector (residuals)
S	Observations Jacobian (partials matrix) for the fault state influence on the residuals

I. Introduction

The NASA Constellation architecture selected a Crew Launch Vehicle (CLV), Ares I, that has an inline configuration with the first stage being a single solid rocket motor (SRM). The Crew Exploration Vehicle (CEV), also known as Orion, will be located on top of the stack and feature a launch abort system that allows it to be ejected in the event of a catastrophic failure. The escape mechanism could save the crew from many critical failure modes but must be activated early, before the vehicle is destroyed in the catastrophe.

The high-level goal of this work is to establish the fundamental ability for early detection of failure during controlled ascent from Earth using only standard sensors available on a launch vehicle. Many critical failures that threaten crew safety and/or mission success do so by causing a loss of control for the vehicle. We are interested in detecting such conditions using standard sensor data available to the vehicle GN&C system.

The particular focus is on the fault conditions leading to a breach of SRM case or one of its joints. The hot gases escaping through the breach create thrust augmentation that could be detected by observing the vehicle dynamics. Such was the case in the Challenger mishap. In Shuttle, the escaping hot jet might impinge on the main tank, which caused the Challenger explosion. The CLV has an inline configuration and the hot gases escaping through the breach will not directly threaten the vehicle. However, the reaction force of the escaping jet (thrust augmentation force) applied away from the vehicle center of gravity creates a lateral moment. If a sufficiently large breach develops, the moment caused by the thrust augmentation may overwhelm flight control. The rocket may start tumbling and eventually break up.

The objective of our study was to determine if applying optimal multivariable estimation methods can yield reliable estimates of (i) the severity of the evolving fault condition as manifested through thrust augmentation, and (ii) the location of the breach. The ability to estimate the position and track the development of a case breach condition would be a valuable tool for mission control. The multivariable estimation approaches developed in this work are relatively advanced and go beyond simpler monitoring methods such as “redlining” presently used in space launch vehicles. A part of our objective was to understand the trades involved in using the multivariable estimation versus more basic, flight proven, approaches.

Our analysis relies on medium-fidelity modeling and simulation of the CLV operation during the ascent flight. The dynamics are influenced by the control commands to Thrust Vector Control (TVC) actuators. The CLV dynamics simulation also includes the models of the faults, such as case breach faults. Different fault scenarios could be seeded to test and validate the Fault Detection and Diagnostics (FDD) algorithms. The GN&C Logic is a part of the simulation. The sensor measurement model and control effector models are included in the simulation.

The main contributions of this work, described in more detail below, include the following:

1. Developing a simple 6-DOFmodel of launch vehicle dynamics including models for SRM case breach faults.

2. Formulating and implementing a multivariable approach to estimating thrust augmentation conditions of the launch vehicle from the GN&C data and baselining the approach against redline fault detection.

3. Analyzing feasibility of multivariable estimation algorithms and performing an initial trade study where these algorithms are baselined against redline fault detection.

Multivariable estimation approaches have recently been applied to detecting faults and estimating fault parameters in aircraft flight actuators and aerodynamic surfaces from GN&C data, see.[7,19,20] Little published work seems to exist on estimating fault parameters from multivariable launch vehicle data. One exception is an earlier paper of an author of this paper and others[11] that explored a more basic model and an optimization-based approach with slow sampling. This paper provides a more mature and practical approach relying on the specific form of the rocket 6-DOF dynamics to achieve faster response and better clarity of the estimation algorithms.

We used a least-squares approach for estimating the fault state, which is closely related to Kalman filtering. Such approaches are well established in space systems applications. The flight dynamics are nonlinear, and we used an estimation analog of dynamic inversion approach to obtain a linear formulation of the estimation problem.

II. Modeling and Simulation

This section provides an overview of the models used in the analysis and the simulation software implementing this model. The simulation was developed in Matlab/Simulink^{TM} environment from Mathworks. The simulation includes blocks for vehicle dynamics, flight control (attitude rate stabilization of the rocket), guidance logic (stabilization of prescribed attitude angles), and basic FDD algorithm logic. A 3-DOF animation block is used for visualization of the rocket state as the simulation evolves. Fault parameter evolution scenario is programmed in a separate block and can be flexibly changed.

The main part of the simulation is a model of launch vehicle dynamics. This model includes the following main components:

- 6-DOF rigid-body dynamics and kinematics of the rocket.
- Environmental models, such as atmosphere model and Earth gravity model.
- Basic physics of thrust generation in the SRB, including an empirical model of random thrust variation in longitudinal and lateral directions, Thrust Vector Control (TVC) physics, and dynamics of the TVC actuators.
- External aerodynamics of the rocket modeled using pre-set aerodynamic tables.
- Model of thrust augmentation caused by a case breach.

II.A. Fault modeling

The report[4] analyzed critical failure modes for the CLV First Stage in detail, based on past flight experience with similar vehicle components. The study showed that an overwhelming majority of the critical failure modes threatening the launch vehicle are related to loss of vehicle control. These failure modes leave the crew capsule a chance to escape before the rocket starts disintegrating.

The failure modes potentially leading to loss of vehicle control include:

- Loss of lateral control of the vehicle because of thrust augmentation caused by hot gases escaping through a breach in SRM. The breach could happen in a joint between SRM segments, in the nozzle attachment, in the igniter seal, or as a result of a case burst or burn-through.
- Loss of lateral control because of a TVC actuator failure.
- Loss of longitudinal control because of nozzle throat burn-through or clogging.
- Loss of roll control because of abnormal SRM combustion creating a roll moment, and RCS thrusters having insufficient control authority to compensate for it.

Our focus in this work is on the first of these failure modes, namely loss of lateral control because of thrust augmentation. A simple model of thrust augmentation used in this work is illustrated in Figure 1. The root cause and detailed characterization of a gas jet escaping through a breach in the SRM case are ignored. The failure is modeled empirically as a reaction force (thrust augmentation) applied at a fixed breach location in a lateral direction. The axial component of the thrust augmentation is masked by the much bigger force of engine thrust, and could be reasonably ignored.

A TVC actuator failure could be reliably detected at the actuator level using a redlining approach, and does not require resorting to multivariable estimation. The other two failure modes mentioned above were left out of the scope of our work at this stage. To a large degree, these failure modes are decoupled from the lateral dynamics of the vehicle.

II.B. Rigid-body dynamics

A rigid-body model was used to describe the launch vehicle dynamics. The CLV is a long and thin vehicle, and a rigid-body model is not entirely accurate. A more accurate model should include a few structural flexibility modes of the rocket and propellant slosh in the Upper Stage tanks. Yet, even in the presence of structural flexibility, the dynamics of the center of gravity (CG) and the rigid-body-mode rotation are the same as for the rigid-body model. The main difference is that flexible dynamics makes the measured

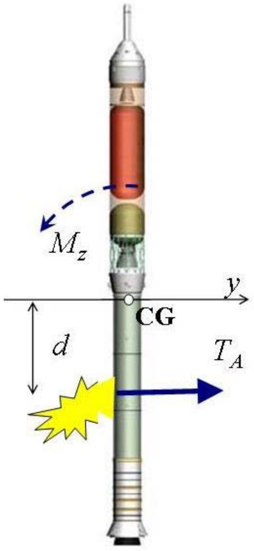

Figure 1. Thrust augmentation

variables different from the rigid-body states. For example, the actual CG acceleration would differ from what is registered by the IMU accelerometers on the top of the rocket; similarly, the actual thrust vectoring angle will include a flexible deflection and differ from what is measured at the gimbal. By using more detailed models, the actual CG acceleration and rigid-body rotational accelerations could be computed more accurately. The remainder of the discussed FDD approaches could still be applied with such modification.

Overall, the rigid-body dynamics model considered herein is used to establish feasibility of the proposed approach. The model needs to be enhanced as a part of the technology maturation process. This work considers the models at (or around) a fixed time in the flight. The mass, inertia, and thrust parameters of the rocket are considered fixed, and therefore the aerodynamics models used cover a relatively narrow range of flight conditions. Use of these algorithms onboard the CLV will require a look-up table to model the time dependencies of these parameters throughout the flight.

The models used in this work are formulated in a coordinate system illustrated in Figure 2. This is an inertial coordinate system aligned with the vehicle at a given fixed point in time. The coordinate origin is at the CG. The longitudinal coordinate axis x is a rotational symmetry axis for the vehicle, the y axis is in the pitch plane, and the z axis is in the yaw plane.

Since the vehicle is moving, a new coordinate system should be considered at each point in time. Integrating equations of motion requires considering transition between these systems. This is standard in aerospace dynamics modeling and is handled in our Simulink model through a quaternion formulation of the 6-DOF dynamics. Such a formulation is available as a standard block in Simulink Aerospace Blockset.

The main innovation in this work is the estimation of fault parameters from the vehicle dynamics. This is done by comparing observed vehicle accelerations against the model-based predictions. The attached coordinate system is convenient for formulating instantaneous accelerations of the vehicle and for use in the estimation algorithms. State variables used by the algorithms come directly from the GN&C system as either measurements or computed states. The fault estimation algorithms perform no integration of vehicle dynamics. Thus, the instantaneous aligned coordinate system can be used without any regard for the coordinate system transition from one instant to another.

Consider the instantaneous linear and rotational velocities of the vehicle in the attached coordinate system:

$$\bar{v} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}, \qquad \bar{\omega} = \begin{bmatrix} p \\ r \\ q \end{bmatrix} \tag{1}$$

The linear velocities $\bar{v}$ in (1) are obtained by projecting the CG velocity vector on the attached axes. The angular velocities $\bar{\omega}$ in (1) are obtained by projecting the instantaneous rigid-body angular velocity vector

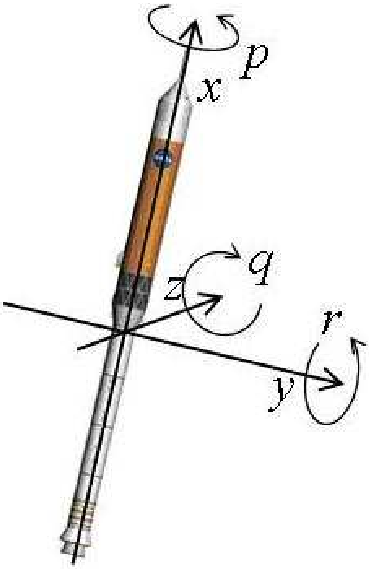

Figure 2. Attached coordinate system for rocket dynamics analysis

on the attached axes.

II.B.1. Translational dynamics

The translational dynamics of the vehicle (dynamics of the CG) can be formulated by using Newton's second law in the form

$$m\dot{\bar{v}} = \bar{F}_{THRUST} + \bar{F}_{AERO} + \bar{F}_{AUGMENT} \tag{2}$$

where the subscripts on the forces on the right-hand side of (2) respectively indicate

$THRUST$ – main propulsion thrust force;

$AERO$ – the total aerodynamic force;

$AUGMENT$ – thrust augmentation force caused by the gases escaping through a breach.

In the absence of thrust vectoring, the main propulsion thrust force $\bar{F}_{THRUST}$ was assumed to be

$$\bar{F}_{THRUST} = \begin{bmatrix} T + W_x(t) \\ W_y(t) \\ W_z(t) \end{bmatrix}, \tag{3}$$

where $W_x(t)$ denotes the random variation of the thrust, modeling turbulence and nonuiformity of the exhaust flow, as well as the unmodeled "in-family" variation of the thrust time profile between different SRMs. In our simulations, this variation is modeled as a low-pass filtered white noise and has a magnitude of about 10% of the main thrust T. The random forces $W_y(t)$ and $W_z(t)$ model the effects of the engine exhaust turbulence. They are simulated similarly to $W_x(t)$ and their magnitude is about 1.4% of the main thrust T. Thrust vectoring was modeled as a rotation of the force $\bar{F}_{THRUST}$ about the gimbal center in the direction of the attached y and z axes. We used a second-order nonlinear model of the TVC actuator dynamics.

The aerodynamic force F_{AERO} could be computed from aerodynamic tables and depends on angle of attack, sideslip angle, air density (follows from altitude) and airspeed. In our simulations, the vehicle underwent pitching motion only. The sideslip was not present and was not included in the aerodynamics. This is a simple model; it could and should be improved in the future. In addition to sideslip, the model omits wind turbulence. This might yield an additional error early in the flight (while the rocket speed is still relatively small).

The aerodynamic force components were computed as

$$F_{AERO,k} = \frac{1}{2} c_k A \rho V^2, \tag{4}$$

where $k = x, y, z$. The meaning of the variables in (4) is as follows:

c_k is a nondimensional aerodynamic coefficient;

V is the vehicle airspeed (in the absence of wind $V = |\bar{v}|$);

ρ is the air density (can be computed from the altitude using a standard atmosphere model);

$A = \frac{1}{4}\pi D^2$, where D is the vehicle characteristic diameter.

We simulated CLV flight around the regime where it experiences the maximal aerodynamic pressure (max-Q). The aerodynamic coefficients were provided by Dr. Shishir Pandya at NASA ARC. They have been obtained from 3-D Computational Fluid Dynamics (CFD) analysis of the CLV flight carried out at NASA ARC as a part of the Simulation-Assisted Risk Assessment study. The detailed CFD analysis was performed on Columbia supercomputer at NASA ARC. The aerodynamic coefficients obtained in this way were then tabulated in Simulink and accessed in the simulation run time. Accurate aerodynamics models for the CLV are required for GN&C analysis and design, and will be developed and improved during the course of on-going CLV development. The proposed FDD logic will re-use the same updated aerodynamic tables.

Thrust augmentation force $\bar{F}_{AUGMENT}$ is generated by the hot gases escaping through a breach in SRM joint, seal, or case. It is assumed that the jet escapes the breach radially (in the lateral direction) and the reaction force is aimed towards the SRM axis.

In addition to the side reaction force, case breach causes a loss of axial thrust. Let F_A be the magnitude of the lateral thrust augmentation force (fault intensity). We assume that the axial thrust loss should be proportional to F_A since the combustion products escaping through the breach are missed at the nozzle. The mass rate of the breach jet is subtracted from the mass rate at the main nozzle. (We assume that the overall SRM combustion rate is not affected.) The proportional loss of the axial thrust can be expressed as

$$\Delta F_{AUGMENT,x} = -p_A F_A, \tag{5}$$

where p_A is fixed factor. It can be estimated as a ratio of the exhaust velocities at the nozzle and at the breach.

The overall change in the thrust force caused by a case breach fault was computed as

$$\bar{F}_{AUGMENT} = \begin{bmatrix} -p_A F_A \\ -F_A \cos(\beta_A) \\ -F_A \sin(\beta_A) \end{bmatrix}, \tag{6}$$

where β_A is the azimuthal angle of the breach location on the SRM surface.

II.B.2. Rotational dynamics

The attitude dynamics of the vehicle (rigid-body dynamics) can be formulated as Euler rotational dynamics equations in the form

$$I\dot{\bar{\omega}} \;\; = \;\; \bar{M}_G(\bar{\omega}) + \bar{M}_{THRUST} + \bar{M}_{AERO} + \bar{M}_{AUGMENT} \tag{7}$$

where I is the inertia matrix and the moment $\bar{M}_G(\bar{\omega})$ captures the cross-terms in the Euler equations,

$$I = \begin{bmatrix} I_{xx} & & \\ & I_{yy} & \\ & & I_{zz} \end{bmatrix}, \quad \bar{M}_G(\bar{\omega}) = \begin{bmatrix} (I_{yy} - I_{zz}) \cdot r\,q \\ (I_{zz} - I_{xx}) \cdot p\,q \\ (I_{xx} - I_{yy}) \cdot p\,r \end{bmatrix}, \tag{8}$$

The angular accelerations on the left-hand side of (7) are defined by the moments on the right-hand side. This is similar to (2); the meaning of the subscripts on the moments is the same as for the forces in (2).

The TVC moment can be computed as a cross product

$$\bar{M}_{THRUST} = \bar{F}_{THRUST} \times \bar{D}_{CG}, \tag{9}$$

where $\bar{D}_{CG} = [-D_{CG} \;\; 0 \;\; 0]^T$ is a vector pointing from the vehicle CG to the TVC gimbal center.

Similarly to (4), the components of the aerodynamic moment vector $\bar{M}_{AERO}$ can be expressed through nondimensional aero moment coefficients:

$$M_{AERO,k} = \frac{1}{2} c_{a,k} \frac{\pi D^2}{4} D \rho V^2, \tag{10}$$

where, again, $k = x, y, z$.

The augmentation moment can be computed as a cross product

$$\bar{M}_{AUGMENT} = \bar{F}_{AUGMENT} \times \bar{d}_A, \tag{11}$$

where $\bar{d}_A = [-d_A \;\; 0 \;\; 0]^T$ is a vector pointing from vehicle CG to the breach location. (We assume that the breach, as well as the main part of the SRM, is located aft of the CG).

II.C. Guidance, navigation, and control

A trivial model of the navigation system and other sensors was used in this work. We assumed that the measured state variables are available to the algorithms directly, undistorted by sensor noise or sensor dynamics.

The closed-loop flight control of the rocket has a bandwidth of a fraction of a Hertz. In practice, this bandwidth is limited by necessity to keep it below the main flexible mode of the rocket. Though only rigid-body dynamics were considered, the flight control loop was designed to have a realistic bandwidth. On-board avionics of a space launch vehicle would allow flight control sampling rate of several tens of Hertz, more than an order of magnitude above the loop bandwidth. This means that a continuous-time model is sufficiently accurate. A guidance loop subsuming the flight control loop has an even smaller bandwidth. A continuous-time model of flight control and guidance was used in the simulation.

Simulating flight control and guidance feedback is important for validation of the FDD algorithms. When a case breach causes thrust augmentation, it is countered by flight control action and the rocket is stabilized. The FDD algorithms must estimate the thrust augmentation from the closed-loop data.

Below, we consider pitch control of the rocket. Since the TVC gimbal angle is relative small (less than 0.1 rad), the yaw control is decoupled from pitch control and is designed identically. The PD control law is used to stabilize the vehicle pitching rate to a commanded value r_{CMD}. The controller computes the TVC gimbal angle demand as

$$\theta = -k_D r - k_P \int (r - r_{CMD}) dt, \tag{12}$$

where k_D and k_P are the derivative and the proportional feedback gains, respectively, and $r = \dot{\alpha}$ is the rocket pitching rate. The gimbal demand θ is provided to the TVC actuator servos. A standard anti-windup action was added to the control (12) to handle the TVC actuator saturation. The gains k_D and k_P were chosen to provide a closed-loop bandwidth of about 0.5 Hz with the damping ratio around 0.7.

The guidance loop is cascaded with the flight control and computes the rate demand for the inner loop using a PI controller. It feeds back the pitch angle deviation from the commanded attitude in the absolute coordinate system. Only a brief segment of the flight trajectory was simulated and the commanded attitude was constant. The feedback gains of the guidance loop were chosen such that the loop bandwidth is about 0.25 Hz.

An important real-time part of the simulated FDD function is data collection. In our simulations, all FDD data were low-pass filtered with a first-order filter having 20Hz bandwidth. The data were then sampled at 10Hz – the Nyquist frequency of the continuous low-pass filter. The FDD algorithms were simulated to run at 10Hz frequency – slower than flight control, but much faster than the bandwidth of the flight control loop. Accurate modeling of FDD data filtering and sampling is important for the validation of the fault detection and estimation algorithms. Because of the simulated high-frequency variation of the rocket thrust, the acceleration data are very noisy. The fundamental ability to estimate the fault signal in the noisy data depends on the signal-to-noise ratio, which is influenced by filtering and sampling.

III. Fault Detection and Diagnostics Problem

This work pursues a model-based approach to Fault Detection and Diagnostics. We use models at two levels. First, this section describes a prediction model that describes the nominal dynamics in the absence of any faults. The prediction model can be used to discern a deviation from a nominal dynamics by observing vehicle GN&C data and computing a deviation from the prediction (residuals). To illustrate the utility of such a model, we consider how the prediction residuals compare with raw sensor data observed in the standard redlining approach. The model at the second level describes how the signatures of the faults appear in the residuals. These fault signature models are used in the multivariable algorithms of the next section.

III.A. Redlining – baseline approach

The following fault scenario was used in the simulation to evaluate the fault estimation approaches. We assumed that a case breach occurred at the coordinate $d_A = -32.3$m (aft of the rocket CG, about 15m forward of the SRM gimbal). The magnitude of the lateral thrust augmentation force ramps up linearly over 10 seconds from zero to 7% of the main thrust and then remains at that level. The magnitude ramp models the growth of the breach as the escaping hot gases burn an opening of increasing size in the SRM case. The augmentation force and moment are as described by (6), (11). In accordance with (5), having an augmented thrust of this magnitude means that 42% of the main engine thrust is lost because of combustion products escaping through the breach. The fully developed thrust augmentation of this magnitude still does not overwhelm the control authority of the TVC, and the rocket continues its controlled ascent through the fault scenario simulation.

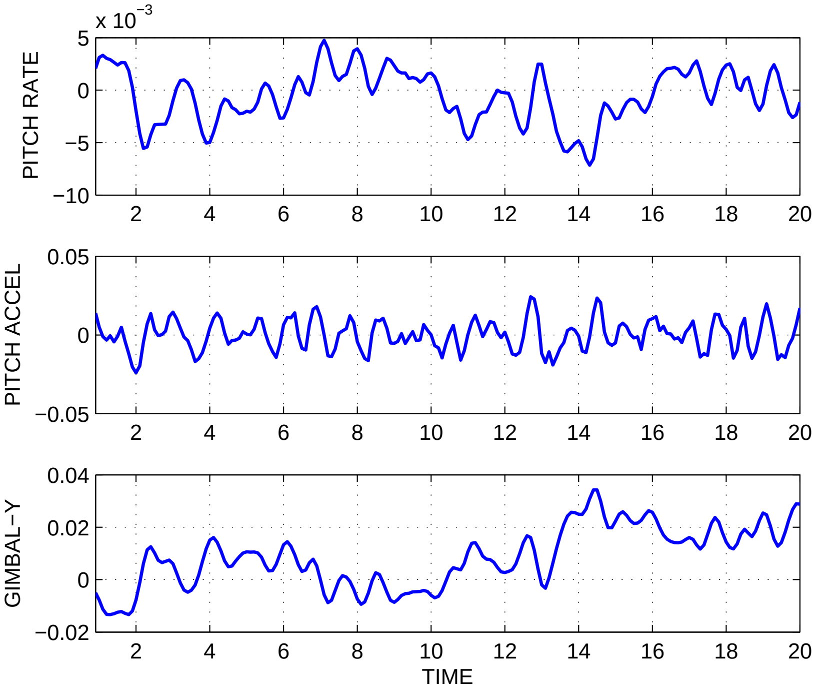

Figure 3. Redlining data from simulation run

The basic approach to detecting an incipient loss of control is to monitor a selected set of sensor channels for large deviations. If the measurement crosses a red line, a fault is declared. The redlining is the main approach used in practice, and is considered as a baseline in this work. Figure 3 shows plots of three 'raw' signals logged in our fault simulation. The three signals are the ones commonly used for redline detection of the lateral control loss in a rocket. The case breach in the simulated scenario causes the rocket to pitch. The signals on the plots in Figure 3 are

- Pitch rate of the rocket – the upper plot. The fault is not noticeable on this plot.

- Pitch acceleration – the middle plot. The fault is not noticeable on this plot.
- Gimbal deflection for pitch control – the lower plot. The fault leads to appearance of a visible offset in this plot. Yet this offset is less than 2σ (twice the mean variation of the data). Thus, using the gimbal deflection for fault detection would lead to an excessive probability of false alarms if the red line threshold is selected low. For a high threshold, the alarm would not be triggered by the fault.

III.B. Model-based compensation of nonlinearity

The multivariable analysis will be formulated through a 6-DOF velocity vector w combining the linear and angular rates (1),

$$w = \begin{bmatrix} \bar{\omega} \\ \bar{v} \end{bmatrix} \tag{13}$$

Dynamic inversion

One of the fundamental parts of the proposed approach to multivariable fault estimation is related in spirit to the dynamic inversion approach in multivariable flight control design. The dynamic inversion discussed in[2,3] has been used at Honeywell and Boeing for advanced control design in several high-performance aircraft (including the descent and landing control for the X-33 RLV orbiter).

In systems theory, estimation is dual to control. Hence, the same ideas can be applied to advanced mathematical algorithm design in both cases. A standard example is duality of Kalman Filtering and LQR control. We use an estimation dual of dynamic inversion.

Let us very briefly consider the dynamic inversion setup for flight control design. By using the notation (13), 3-DOF attitude dynamics of the vehicle can be presented in the form.

$$\dot{\omega} = f(\bar{\omega}, q) + B(q)u \tag{14}$$

where vector q collects the vehicle states (which, along with ω, are assumed to be directly observed) and vector u collects commands to the vehicle flight control surfaces generating the angular moments. The nonlinear map $f(\cdot,\cdot)$ describes vehicle dynamics and $B(q)$ is the control gain matrix.

The dynamic inversion approach is based on computing the control input u to satisfy the given rate demand

$$r_{des} = f(\bar{\omega}, q) + B(q)u, \tag{15}$$

The control u solving (15) can be computed in the form,

$$u = [B(q)]^{-1} \left(r_{des} - f(\bar{\omega}, q) \right).$$

The control computations use an embedded model of the dynamics in the form of the map $f(\cdot,\cdot)$. The dynamic inversion cancels the nonlinearity. Applying a linear time-invariant feedback law of the form $r_{des} = G(\bar{\omega}_* - \bar{\omega})$, where G is the feedback gain matrix, then leads to a linear dynamics

$$\dot{\bar{\omega}} = G(\bar{\omega}_* - \bar{\omega})$$

that can be designed and analyzed using standard linear systems control tools.

Prediction residuals for estimation

Using prediction residuals for estimation could be considered a dual of the dynamic inversion control approach. Similarly to (14), the 6-DOF dynamics of the launch vehicle can be presented in the form

$$\dot{w} = g(w, q; X) \tag{16}$$

where w is the 6-DOF velocity state vector (13), vector q collects the launch vehicle states (which are assumed observable), $g(\cdot,\cdot)$ is the nonlinear dynamics map, and X is the vector of fault parameters.

The prediction residuals for the six accelerations $\dot{w}$ could be computed similarly to (15). Assuming that there is no fault, $X = 0$, yields

$$\bar{r} = \dot{w} - g(w, q; X = 0), \tag{17}$$

The nonlinear dynamic map $g(w, q; X = 0)$ in (16)-(17) is defined as

$$g(w, q; X = 0) = \begin{bmatrix} I^{-1}(\bar{M}_G + \bar{M}_{THRUST} + \bar{M}_{AERO}) \\ m^{-1}(\bar{F}_{THRUST} + \bar{F}_{AERO}) \end{bmatrix}, \tag{18}$$

where I and $\bar{M}_G$ are given by (8), $\bar{F}_{THRUST}$, $\bar{F}_{AERO}$, $\bar{M}_{THRUST}$, $\bar{M}_{AERO}$ are given by (3), (4), (9), and (10). By substituting accelerations $\dot{w}$ from (13), (2), (7) and the map $g(w, q; X = 0)$ from (18) into (17) we obtain the residuals to be

$$\bar{r} = \begin{bmatrix} I^{-1}\bar{M}_{AUGMENT} \\ m^{-1}\bar{F}_{AUGMENT} \end{bmatrix}. \tag{19}$$

Computing the residuals cancels nonlinearity of the dynamics. The residuals are linearly related to the thrust augmentation forces and the moments. Advanced estimation algorithms discussed in this report compute the residuals first, and then use the residuals for estimating the fault states.

III.C. Prediction residuals

The expression (19) for the residuals was obtained by manipulating equations of motion. In practice, the residuals (18) can be computed on line from the data available in the GN&C system of the launch vehicle. The flow of the computations is illustrated in Figure 4. One input to the computations is the vector of six accelerations. Instantaneous values of linear and angular accelerations are measured by accelerometers in the IMU (inertial measurement unit) of the vehicle. Another input collects the sensor channels describing the dynamical states of the vehicle rigid-body model. A list of these state measurements can be found at the end of Section III.

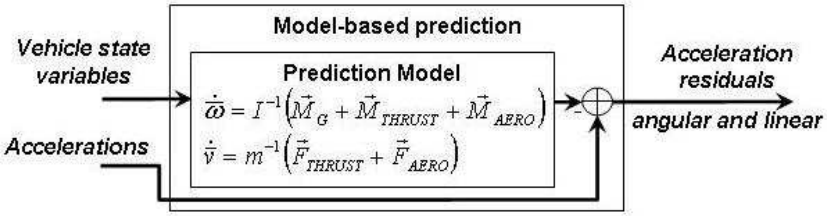
Model-based prediction
Vehicle state variables
Prediction Model
$\dot{\vec{\omega}} = I^{-1}\left(\vec{M}_G + \vec{M}_{THRUST} + \vec{M}_{AERO}\right)$
$\dot{\vec{v}} = m^{-1}\left(\vec{F}_{THRUST} + \vec{F}_{AERO}\right)$
Accelerations
-
Acceleration residuals
angular and linear

Figure 4. Data flow diagram for computing residuals using the prediction model

Ideally, the six acceleration residuals shown as the output in Figure 4 will be related to the thrust augmentation forces and moments by (19). Of course, this is only true if the models used for computing the residuals describe the actual vehicle dynamics perfectly. This is never the case in practice, and the computed residuals will reflect modeling errors and noises in addition to the thrust augmentation (19).

The data used in our algorithm validation study are obtained through simulation, as described in Section II. The prediction model used with this data is based on the same parameters and has the same structure as the 'truth' simulation model. A random thrust variation (3) is included in the simulation model to mimic realistic modeling errors. The main (deterministic) part of the thrust T is assumed to be available to the prediction model (the time-profile of SRM thrust is well-known and highly repeatable). At the same time, the disturbance inputs W_x, W_y, and W_z in (3) are unavailable to the prediction model. By propagating through the closed-loop dynamics of the controlled rocket flight, the thrust variation noise impacts the residuals. Thus, the residuals are nonzero even in the absence of a fault.

Figure 5 shows the acceleration residuals (solid lines) obtained in the simulated fault scenario, along with the acceleration signals (dashed lines) used in the residual computations. These accelerations are low-pass filtered and sampled, as discussed at the end of Section II. As one can see from Figure 5, the onset of failure at time $t = 10$ is not easily discernable from the raw acceleration signals. At the same time, the residuals clearly show growing offsets as the fault evolves (thrust augmentation force ramps up).

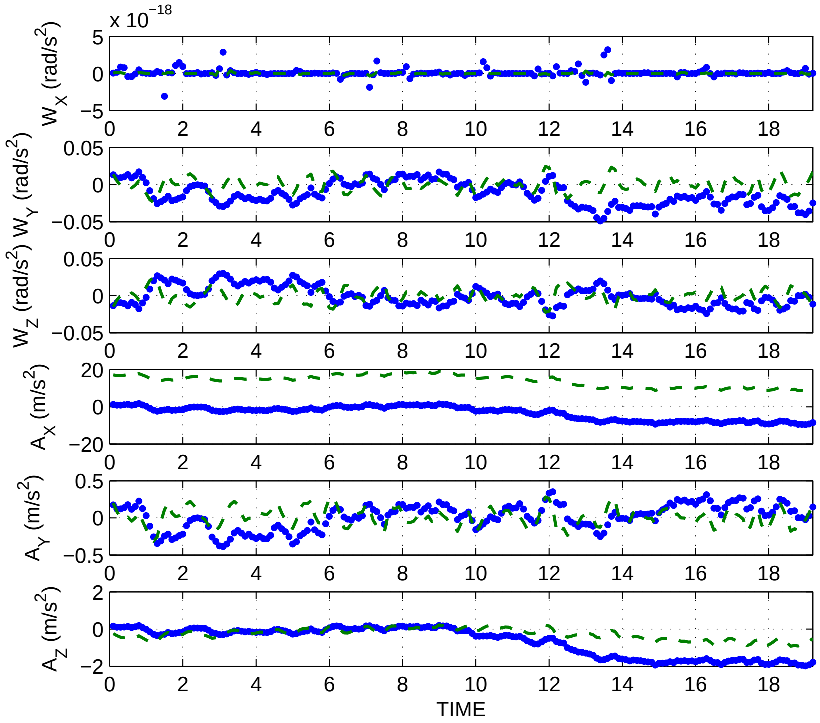

Figure 5. Accelerations (dashed) and acceleration residuals (solid) in the simulation run

IV. Multivariable Estimation

The residuals contain the information about the fault state. This information is contaminated by noise. Reliable estimation and detection of the fault state from the noisy data require using statistical averaging. Such signal processing needs prior knowledge about the fault state (thrust augmentation) signal that could include

1. Knowledge of time-domain evolution of the fault. This knowledge might be stated in the frequency domain: e.g., high-frequency signal components correspond to noise and low-frequency components reflect the fault.

2. Knowledge that the residuals depend on a smaller number of underlying fault parameters. For example, one could reasonably assume that the augmentation forces and moments are caused by a single lateral force applied radially at the breach location.

This section considers approaches that can incorporate prior information of these types. The statistically optimal fault estimation algorithms are based on a formal model of the signal. In most general form, consider a time-dependent fault vector $X(t)$ that needs to be estimated from data $Y(t)$. The vector $X(t)$ describes thrust augmentation and the vector $Y(t)$ includes some (but possibly not all) components of the residual vector r at time step t). Since the faults are related to the thrust augmentation forces and we observe acceleration residuals, there is an instantaneous relationship between $X(t)$ and $Y(t)$. It can be expressed as a linear observation equation

$$Y(t) = S(t) \cdot X(t) + e(t), \qquad e(t) \sim N(0, Q), \tag{20}$$

where $S(t)$ is a Jacobian (sensitivity) matrix of an appropriate size, and $e(t)$ is the noise vector realization at time t. The columns of $S(t)$ represent signatures of the individual fault states (components of vector $X(t)$). The 'noise' $e(t)$ in (20) incorporates modeling errors, sensor noise, random variation of the thrust, and other sources of randomness. In the absence of detailed information about the noise $e(t)$, we model it as an i.i.d.

(independent, identically distributed) Gaussian white noise sequence with covariance matrix Q. This is a usual engineering assumption. The covariance Q can be empirically estimated from historical residual data obtained in nominal operation (without a fault present).

A linear model of the form (20) could be justified in more than one way. Firstly, the system physics are such that the residuals (19) linearly depend on the forces and moments caused by the thrust augmentation. Secondly, even with various nonlinearities present (e.g., aerodynamic changes caused by the jet escaping through the case breach) a linearized model of the form (19) is a reasonable engineering approximation. Finally, as demonstrated below, it might be possible to work around some of the nonlinearities by limiting the analysis to a subset of the data.

Derivation of optimal estimation algorithms requires the observation model (20) to be complemented by a prior model that describes available statistical knowledge of the fault state sequence $X(t)$. The prior model should hold in the absence of any observation information. We assume a Markov model of the fault evolution in a general state space form

$$X(t+1) = AX(t) + \xi(t), \tag{21}$$

where A is a fixed matrix, and $\xi(t)$ is an i.i.d. sequence of random variables.

More specifically, the results presented in this paper have been obtained assuming an independent random walk model for each component of vector X:

$$X_j(t+1) = X_j(t) + \xi_j(t), \tag{22}$$

where $\xi_j(t)$ is independent of $\xi_k(t)$ for $j \neq k$. The model (22) corresponds to A being an identity matrix. In the estimation approaches described below, we assume different types of distributions for the fault driving sequences $\xi_j(t)$.

In addition to the simple random walk model, a second-order random walk model was used and corresponding algorithms were tried out in simulations. This is a more complex model of the form (21) and its use for fault estimation is discussed in more detail in.[10] Using the second-order model makes the algorithm more complicated; at the same time we found that detection of the fault condition does not improve by much. Therefore, the results for the second-order random walk model are not discussed in this report. The second-order model allows predicting a linear trend in the fault evolution, and could be useful if such prediction is desirable.

IV.A. Linear Gaussian model

As a first and most straightforward version of (20), consider a linear model for the lateral dynamics residuals. Assume that the fault vector collects components of the lateral thrust augmentation forces and moments and has the form

$$X = \begin{bmatrix} \Delta F_{AUGMENT,y} \\ \Delta F_{AUGMENT,z} \\ \Delta M_{AUGMENT,y} \\ \Delta M_{AUGMENT,x} \end{bmatrix} \tag{23}$$

In fact, the four components of vector (23) are not completely independent as they are jointly defined by three parameters in (6), (11): thrust augmentation magnitude F_A, azimuthal location β_A of the case breach, and longitudinal coordinate of the breach location d_A.

We will use an observation vector Y including four residuals: two for the lateral accelerations and two for the lateral angular accelerations:

$$Y = \begin{bmatrix} \Delta \dot{r} \\ \Delta \dot{q} \\ \Delta \dot{v}_y \\ \Delta \dot{v}_z \end{bmatrix}. \tag{24}$$

The axial acceleration data are ignored because in accordance with (6), the loss of axial thrust is related to the lateral augmented thrust components in (23) as

$$\Delta F_{AUGMENT,x} = -p\sqrt{\Delta F^2_{AUGMENT,y} + \Delta F^2_{AUGMENT,z}}. \tag{25}$$

Considering axial thrust would make the model nonlinear in vector X.

Combining (2), (7), (6), (11), (23), and (24) yields a linear observation model of the form (20), $Y(t) = S(t)X(t)$, where the 4×4 sensitivity (fault signature) matrix has the form

$$S = \begin{bmatrix} 0 & -d_A/I_{zz} & 1/I_{zz} & 0 \\ d_A/I_{yy} & 0 & 0 & 1/I_{yy} \\ 1/m & 0 & 0 & 0 \\ 0 & 1/m & 0 & 0 \end{bmatrix}. \tag{26}$$

As the prior model, this subsection assumes a random walk model of the form (21), (22) for the components of the fault state vector X (23).

$$X(t+1) \quad = \quad X(t) + \xi(t), \quad \xi(t) \sim N(0, \Xi), \tag{27}$$

where $\xi(t)$ is a white Gaussian noise sequence with a covariance matrix Ξ.

IV.A.1. Kalman Filter estimation

The linear-Gaussian model (21), (22), (23), (24), (26), (27) allows optimal estimation of fault state in the recursive form of a Kalman Filter:

$$\hat{X}(t+1) = \hat{X}(t) + K(t)\left(Y(t) - S(t)\hat{X}(t)\right), \tag{28}$$

where the filter gain $K(t)$ is computed from a Riccati equation update. An exact form of the update can be found in most standard textbooks on estimation theory and signal processing. The Riccati equation includes the observation noise covariance matrix Q and the (fault) state driving noise covariance Ξ.

The Kalman filter was applied to the residual data in Figure 5. The observation noise covariance Q was empirically estimated from the residuals collected in a simulation without faults. This noise is in fact caused by the random variation of the thrust propagating through the rocket dynamics and the closed-loop flight control. When designing the filter, the state-driving noise covariance was assumed to have the form $\Xi = \xi_0 \cdot I$. The scalar parameter ξ_0 was used as a tuning parameter in the algorithm design.

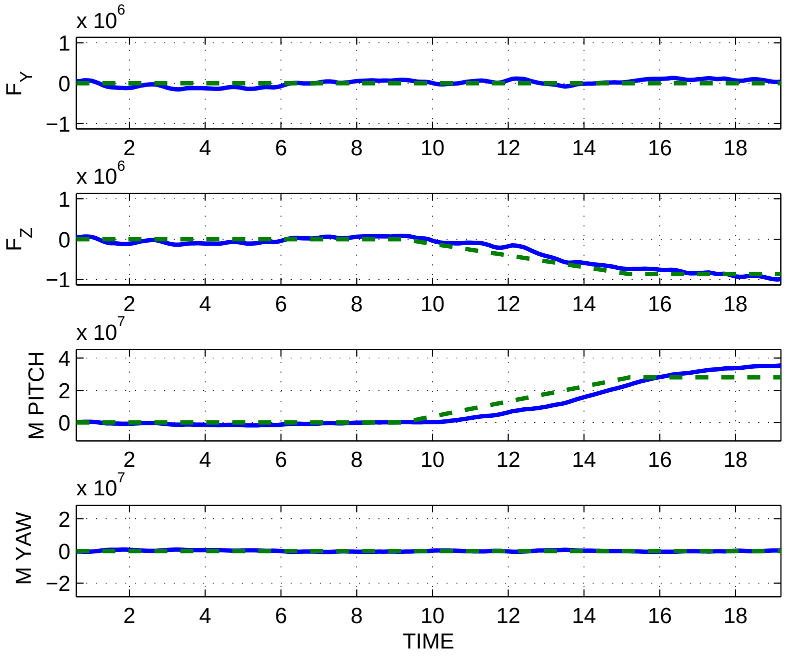

Figure 6. Estimation results for linear-Gaussian model (Kalman filter)

The estimation results (components of the vector (23)) are shown in Figure 6. The upper two plots show the estimated lateral force components, the lower two plots show the estimated lateral moments. The solid lines mark the obtained optimal estimates, and the dashed lines are the forces and the moments seeded in the simulation (the ground truth data). The latter are shown for comparison and algorithm validation purposes only. The estimation algorithm did not have access to the ground truth data.

Overall, it is much easier to detect the fault onset from the estimates in Figure 6 compared to the raw data redlining in Figure 3, as can be seen from these plots. The force and moment in Figure 6 leave the 3-σ band at around $t = 15$s in the simulation. The TVC gimbal position in Figure 3 leaves the 2-σ, band but never gets out of the 3-σ band.

By comparing the estimates for lateral forces and moments makes it possible, at least in principle, to estimate the location of the case breach.

IV.B. Batch formulation of linear estimation

Let us now consider a batch solution of the optimal estimation problem based on the linear-Gaussian model (23), (24), (26), (27). The batch solution is equivalent to the Kalman filter but is presented in a different form. The linear-Gaussian derivation below is subsequently generalized for a non-Gaussian distribution in the next subsection, which introduces more efficient estimation algorithms.

Consider a set of the observed vectors $Y(t)$ (24) reflecting the underlying fault states (23)

$$\begin{aligned} Y_N &= \mathrm{col}\{\, Y(1), ..., Y(N)\, \}, & (29)\\ X_N &= \mathrm{col}\{\, X(1), ..., X(N)\, \} & (30) \end{aligned}$$

Consider maximum a posteriori (MAP) estimation for the model (20), (27). The MAP estimate for the trend X_N is obtained by maximizing the conditional probability of the trend given the time-series data Y_N. This can be expressed as $P(X_N|Y_N) \to \max$. Using Bayes' rule, the conditional probability can be expanded as

$$P(X_N|Y_N) = P(Y_N|X_N) \cdot P(X_N) \cdot C, \tag{31}$$

where $C = [P(Y_N)]^{-1}$ is a scaling factor independent of X_N. Since the measurement noise values $e(t)$ in (20) are independent for different t, we get

$$P(Y_N|X_N) = \prod_{t=1}^{N} p_e\left(Y(t) - S(t)X(t)\right). \tag{32}$$

By applying Bayes' rule to equations (27) for each t and using the fact that $e(t)$ in (20) are independent, we get

$$P(X_N) = p_0\left(X(1)\right) \cdot \prod_{t=2}^{N} p_\xi\left(X(t) - X(t-1)\right). \tag{33}$$

We assume that $e(t)$, $\xi(t)$ and $X(1)$ in (20), (27) are independent, identically (normally) distributed

$$p_e(\cdot) \sim N(0, Q), \qquad p_0(\cdot) \sim N(X_0, \Xi_0), \qquad p_\xi(\cdot) \sim N(0, r \cdot I), \tag{34}$$

where, as in the Kalman filter design, the covariance of the state-driving noise is assumed to be $\Xi = r \cdot I$.

The MAP estimation problem can be formulated as $L = -\log P(X_N|Y_N) \to \min$. Substituting (32)–(34) into (31) allows to write the MAP problem in the form

$$L = \frac{1}{2}\sum_{t=1}^{N} \|Y(t) - S(t)X(t)\|_Q^2 + \frac{1}{2}\|X(1) - X_0\|_{\Xi_0}^2 + \frac{1}{2} r \sum_{t=2}^{N} [X(t) - X(t-1)]^2 \to \min, \tag{35}$$

where $\|\cdot\|_W$ denotes a weighted Euclidean norm such that $\|Y\|_W^2 = Y^T W Y$. The expression (35) for the log-likelihood loss index L omits an additive constant and a positive multiplier, which do not influence the optimization result.

Problem (35) can be presented in matrix form as

$$L = \frac{1}{2}\|Y_N - A_N X_N\|^2_{Q_N} + \frac{1}{2} r\|D_N X_N\|^2 + \frac{1}{2}\|e_1^T X_N - X_0\|^2_{\Xi_0} \to \min, \tag{36}$$

where $A_N = \text{block diag}\{S(1),...,S(N)\}$, $Q_N = \text{block diag}\{Q,...,Q\}$, D_N is the first difference block matrix, and e_1 is the matrix selecting the first components of vector X_N such that $e_1^T X_N = X(1)$. The solution to (36) is

$$X_N = \left(A_N^T Q_N A_N + r D_N^T D_N + e_1^T \Xi_0 e_1\right)^{-1} \left(A_N^T Q_N Y_N + e_1 \Xi_0 X_0\right). \tag{37}$$

The matrices in (37) are sparse, and the solution can be computed extremely fast even for a large size N of the data set with the help of modern sparse-matrix arithmetic routines, such as those from LAPACK libraries used by Matlab.

The batch solution (37) can be re-computed at each time step as a new residual vector $Y(N)$ becomes available to yield a new and improved estimate of the entire trend X_N, including the last point $X(N)$. The results of such batch filtering (a series of estimates $X(N)$ obtained as N increases) are identical to the results of Kalman filtering displayed in Figure 6.

The batch filtering has to deal with the problem of increasingly large size as the horizon N increases. Eventually the problem size will grow beyond the available computational capabilities. The solution is to use a *Moving Horizon Estimation* (MHE) approach and retain only last N points (where, say, $N = 100$) dropping the old ones as the new data arrive. (There are some subtle details on setting the initial condition penalty, but they have little impact for large horizons.) The MHE approach is computationally more intensive compared with Kalman filtering. For this reason, it has not been extensively used in practice. This disadvantage disappears with the growth of available computing power.

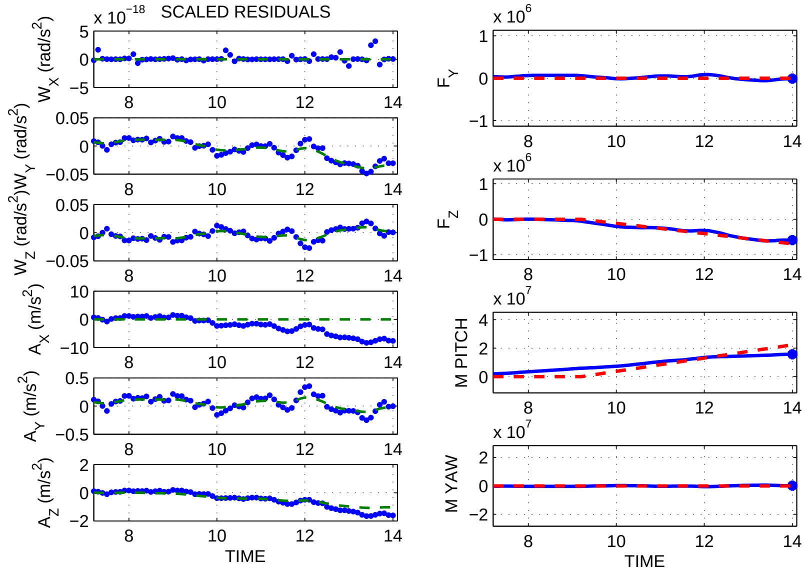

Figure 7. Results for linear-quadratic Moving Horizon Estimation. Acceleration residuals (left plots) and augmented force estimates (right plots)

The main advantage of the MHE approach is that at each time step, an estimate for the entire horizon is obtained. This is an important advantage if the estimation results must be displayed for the crew and/or mission control. The MHE results obtained in the same simulation are displayed in Figure 7. A Kalman filter estimate provides a filtering update at just the most recent time instant. Therefore, the plots in Figure 6, which show the history of the Kalman filter estimates, reproduce the ground truth data less accurately.

IV.C. Extensions

Linear-Gaussian models, such as (21), (22) and algorithms are the staple of practical signal processing and estimation in aerospace systems. In sume cases, for trending fault states, these algorithms can be improved upon by incorporating simple additional information on the faults that cannot be described within a linear-Gaussian framework. Examples of such information are as follows:

- For some fault states, there is a prior knowledge of the fault variable sign. For instance, the thrust augmentation force applied at the breach on the case surface is always directed towards the rocket center.
- For some fault states, it is known that fault intensity can only grow with time because the damage associated with the fault is irreversible. For instance, the thrust augmentation force magnitude is defined by the breach opening that that is burned larger with time by the escaping hot gases.

These kinds of prior information can be stated as *linear constraints* on the fault state variables. The estimate can then be obtained by solving a constrained optimization problem. The recent work on optimal constrained estimation of faults using embedded convex optimization is reflected in.[8, 9, 11, 15, 16, 18, 19] At each time step, as a new residual vector $Y(N)$ becomes available, a Quadratic Programming (QP) problem can be solved to yield a new and improved estimate X_N. To make the approach computationally feasible, the MHE approach is used.

Explicitly taking into account the fact that the fault intensity grows monotonically can significantly improve estimation accuracy. This is discussed and demonstrated in several aerospace-related applications in.[9, 11, 17–19] Unfortunately, for the application in hand this approach cannot be applied as a direct extension of the linear models considered so far in this paper. The reason is that we do not know the sign of the components of forces and moments and whether they are decreasing or increasing in advance before the case breach location is known.

One possible approach that we explored in this work is modeling a thrust augmentation force component as a sum of two variables: one positive and increasing, another negative and decreasing. The reasoning is that if the thrust augmentation force is positive and growing, the solution will come out as the positive variable. If the thrust augmentation force is negative and growing in magnitude, the solution will come out as the negative variable.

In actual implementation of this model, the positive part and negative part might be simultaneously nonzero. The noisy data increase is resolved as increase in the positive component; the decrease yields the negative component. Thus, the filtering action is somewhat limited. The approach did not bring a substantial improvement over linear-Gaussian estimation (Kalman filtering) in our simulations. At the same time, it is much more intensive computationally.

Note that the sum of a positive exponentially distributed variable and a negative exponentially distributed variable is equivalent to a single random variable with a Laplacian distribution. Such a heavy-tailed noise model has certain advantages, e.g., for modeling data with large outliers (spikes or steps). It might prove useful for dealing with real rocket data. In our simulations, however, the driving noises were Gaussian. Possibly because of this, the improvement observed was insufficient to justify using substantially more complex algorithms.

V. Conclusions

We have considered the case breach fault of a crew launch vehicle powered by a solid rocket motor. Hot gases escaping through a breach in the rocket motor case create thrust augmentation that could lead to loss of vehicle control. The reported work is focused on early detection and estimation of the severity and location of the case breach. Using a 6-DOF model of the ascending launch vehicle flight dynamics, this report demonstrated feasibility and practical benefits of multivariable estimation of the augmented lateral forces and moments caused by the breach from the noisy GN&C data.

It has been demonstrated that a multivariable approach, such as Kalman Filtering, improves accuracy of the fault detection significantly over the currently used redlining approaches. The computational performance and modeling requirements of the approach make it an acceptable candidate for Ares I implementation.

The feasibility of the proposed approach was demonstrated on a simplified 6-DOF rigid-body model of the rocket. A number of model improvements is needed for further maturing the approach. These include more accurate modeling of aerodynamics with faults, modeling of the flex and slosh dynamics, and rotational dynamics of the rocket. There is also a need to improve the fidelity of modeling the GN&C system including the RCS actuators.

The simulation analysis described in this report allowed validating *application-level* functionality of the FDD approach. The detail associated with implementation of algorithms on a specific avionics platform were *not* modeled and analyzed in the work reported herein. Such issues include real-time computing requirements, sampling time, and distributed avionics implementation for the algorithms. The related issues of Fault Tolerance and Redundancy Management (FT–RM) were also left outside of this project scope. These issues are important, and need to be considered as a part of the technology maturation process.

VI. Acknowledgements

The authors are grateful to colleagues at NASA ARC: to Dr. Shishir Pandya for providing aerodynamic table data for the simulation model, and to Dr. Vadim Smelyanskiy for discussions of the SRM combustion dynamics.

References

[1] J. Chen and R. J. Patton. *Robust model-based fault diagnosis for dynamic systems.* Kluwer Academic Publishers, Norwell, MA, USA, 1999.

[2] Enns, D.F., Bugajski, D.J., Hendrick, R.C., and Stein, G., "Dynamic inversion: An evolving methodology for flight control design," *International Journal of Control*, Vol. 59, No. 1, 1994, pp. 71–91.

[3] Enns, D.F., "Robust nonlinear multivariable control applications and problems," *AFOSR Workshop on the Theory and Applications of Nonlinear Control*, St. Louis, MI, 15-16 August 1991

[4] J. R. Fragola, B. Putney, J. Minarick III, et al, "Reliability and Crew Safety Assessment for Solid Rocket Booster / J-2S Based Launch Vehicle," *Report SAICNY05-04-1F*, Science Applications International Corp., New York, NY, April 2005

[5] P. M. Frank. "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy–a survey and some new results," *Automatica*, 26(3):459–474, 1990.

[6] J. Gertler. "Survey of model-based failure detection and isolation in complex plants," *IEEE Control Systems Magazine*, 8(6):3–11, 1988.

[7] Glavaski, S. and Elgersma, M. "Active aircraft fault detection and isolation," *IEEE Systems Readiness Technology Conference AUTOTESTCON*, pages 692–705, August 2001.

[8] Goodwin, G. C., Seron, M. M., and De Dona, J. A., *Constrained Control and Estimation*, Springer Verlag, 2004.

[9] Gorinevsky, D., "Monotonic regression filters for trending gradual deterioration faults," *American Control Conference*, pp. 5394–5399, Boston, MA, June 2004.

[10] Gorinevsky, D., "Optimal estimate of monotonic trend with sparse jumps," *American Control Conference*, New York, NY, June 2007.

[11] Gorinevsky, D., Samar, S., Bain, J., and Aaseng, G., "Integrated diagnostics of rocket flight control," *IEEE Aerospace Conference*, Big Sky, MT, March 2005.

[12] R. Isermann. "Process fault detection based on modeling and estimation methods—a survey," *Automatica*, 20(4):387–404, 1984.

[13] Oppenheim, A. V., Schafer, R. W., and Buck, J. R., *Discrete-Time Signal Processing*, Prentice Hall, 1999.

[14] Paddock, E. J., Lin, A., Vetter, K., , Crues, E. Z., "TRICK: A simulation development toolkit," AIAA 2003-5809, *AIAA Modeling and Simulation Technologies Conference and Exhibit*, 11-14 August 2003, Austin, TX

[15] Tyle, M.L, Asano, K., and Morari, M. "Application of moving horizon estimation based fault detection to cold tandem steel mill," *Int. J. of Control*, Vol. 73, No. 5, 2000, pp. 427–438.

[16] Rao, C. V., Rawlings, J. B., and Mayne, D. Q. "Constrained state estimation for nonlinear discrete-time systems: stability and moving horizon approximations," *IEEE Trans. on Automatic Control*, Vol. 48, No 2, 2003, pp. 246–258.

[17] Samar, S. and Gorinevsky, D., "Model predictive estimation of evolving faults," *American Control Conf.*, Minneapolis, MN, June 2006.

[18] Samar, S., Gorinevsky, D., and Boyd, S., "Moving horizon filter for monotonic trends," *IEEE Conf. on Decision and Control*, Paradise Island, Bahamas, December 2004.

[19] Samar, S., Gorinevsky, D., and Boyd, S., "Embedded estimation of fault parameters in an unmanned aerial vehicle," *IEEE Conf. on Control Applications*, Munich, Germany, Oct. 2006.

[20] X. Zhang, M. Polycarpou, and T. Parisini. A robust detection and isolation scheme for abrupt and incipient faults in nonlinear systems. *IEEE Transactions on Automatic Control*, Vol. 47, No. 4, 2002, pp 576–593.