# Health Monitoring and Fault Detection of Spacecraft Attitude Control Thrusters

Daniel Kolano
*Aeronautics & Astronautics*
*Stanford University*
Stanford, United States
dkolano@stanford.edu

## I. Introduction

As the cost of mass to orbit decreases and launch cadence increases thanks to improvements in launch vehicle technology, the relative expense of the payload itself becomes more significant to the mission designer. This often leads to cost-cutting measures at the system level as the monetary and timeline risk of mission failure are reduced. This design philosophy can be seen in several of the lunar landers designed by recipients of NASA's Commercial Lunar Payload Services contracts. Contractors are far more willing to reduce redundancy and rely on low Technology Readiness Level (TRL) thrusters, actuators, and processes than has generally been the case for large-budget, long-timeline state funded missions.

These trades generally result in an increase in uncertainty of the performance of spacecraft components, reduced fault tolerance, and a reduced number of available sensors for use in component health monitoring and fault detection. Specifically regarding spacecraft thruster systems, the system designer may remove redundant valves or other components, resulting in a single point of failure for the system. They may also remove monitoring sensors such as chamber pressure transducers, which would normally give direct insight into engine performance.

With increased risk, higher uncertainty, and lower system observability, developing methods to efficiently estimate the system state and rapidly detect faults has the potential to increase mission success rates at minimal cost.

## II. Problem Statement

This work focuses on the sub-problem of health monitoring and fault detection within the spacecraft propulsion system. For simplicity, the spacecraft sensors, electronics, structure, etc. will be assumed to always operate nominally. We therefore model the full system dynamics using the Euler equation as:

$$J\dot{\omega} = \omega \times J\omega + \tau + Jr_{t,x} \tag{1}$$

Where $\tau$ is the net torque applied by the thrusters, $J$ is the inertia tensor, $\omega$ is the angular velocity of the spacecraft, and $r_{t,x} \sim \mathcal{GWN}(0, Q_x)$ is zero-mean Gaussian White Noise (GWN).

Additionally, our propagation equations depend on a parameter vector $\theta$, which includes an "efficiency" for each thruster, denoted $\theta_i$, corresponding to the percentage of the nominal thrust thruster $i$ is currently producing. The "dynamics" of these parameters are given by $\theta_{t+1} = \theta_t + r_{t,\theta}$ where $r_{t,x} \sim \mathcal{GWN}(0, Q_\theta)$. In other words, our parameters are not expected to change rapidly over time, which implies this model should be able to capture gradual change in performance over time as well as initial errors in parameter estimation (highly likely when using low-TRL thrusters with little to no spaceflight heritage).

However, in the case of a thruster failure, the change in parameters is likely to be nearly instantaneous as most failure processes have extremely short timescales (flow passage blockage, chamber wall burnthrough, actuator failure, etc.). Therefore we also consider the case in which there exists a jump discontinuity in $\theta$. Specifically in the case that $\theta_{i,t} = \theta_{i,t}$, and $\theta_{i,t+1} \approx 0$ for some $i$.

We can calculate the applied torque $\tau$ based on the thrust output of each thruster $T_1, \cdots, T_M$, thruster parameters $\theta_1, \cdots, \theta_M$ and the system geometry as:

$$\tau = \sum_{i=1}^{M} r_i \times \theta_i F_{\text{nom},i} = T(u \cdot \theta)$$

Where $r_i$ is the vector from the center of mass to the point of thrust application for thruster $i$, $F_{\text{nom},i}$ is the nominal thrust applied by thruster $i$, $u \in \mathbb{R}^M$ is the set of input commanded thrusts for each thruster and $T \in \mathbb{R}^{3\times M}$ is a matrix encompassing the geometry of the problem and the thruster output.

In this work we assume that there already exists a filter or sensor capable of estimating the full attitude state of the satellite (orientation and angular rates). This is taken as a discrete 7-element measurement vector $x_t = \begin{bmatrix} q_t^T & \omega_t^T \end{bmatrix}^T$ at each timestep with zero-mean GWN of constant distribution: $v_t \sim \mathcal{GWN}(0, R)$. In other words, our measurement model will be taken as follows:

$$\begin{aligned} y_t &= g(x_t) + v_t = Cx_t + v_t \\ C &= I_{6\times 6} \end{aligned}$$

## III. Literature Review

The basic spacecraft system dynamics model and Kalman filter applied to this model are fundamental to all the computations presented here. Reference [1] provides an ideal overview of the system dynamics, common discretizations, and measurements models. The UKF is also a common filter employed in attitude estimation problems, as explored in references [2], [3], and [4]. However, as usual the UKF is found not to be as computationally efficient at the EKF but does not require calculation of a Jacobian and displays increased robustness to disturbances.

Ref. [5] implements a Particle Filter (PF) based on the Bootstrap Filter to address the same attitude estimation problem. The primary advantage found with this filter was the extremely wide convergence region, with global convergence in initial attitude error. However, given the high computational time of the PF, it is generally avoided given the limited processing available on spacecraft.

With regard to parameter estimation, reference [6] lays out the fundamental optimization problem underlying the combined goal of optimally estimating a set of parameters in combination with the spacecraft state, given a set of inputs and observations with objective of minimizing the two-norm of the error between the estimated and true state vectors, as well as the two-norm of the estimated and true parameter vector. The most common methods of simultaneous state and parameter estimation are to either formulate the problem as a joint or dual estimation method.

The joint method concatenates the parameter vector to the state vector, and runs a single filter on the augmented system. This method is employed in [4] to estimate the alignment of an attitude rate sensor during operation. As the complete system is solved, this formulation allows modelling of direct coupling between parameters and system states, but increases computational complexity due to a larger matrix inversion. Additionally, the coupling of parameter and state spaces can introduce instabilities as the error is also closely coupled [2].

The dual method instead estimates the state and parameter vectors using two separate iterative filters which share limited information. This method avoids the increased computational cost of a larger matrix inversion as well as many instabilities introduced by coupling. However, it cannot capture coupling effects as the Joint filter can. Reference [2] investigates use of a Dual UKF to estimate both the spacecraft state and a set of parameters representing the moment of inertia vector of the spacecraft.

Finally, with respect to the problem of early detection of rapid-onset faults, reference [7] provides a strong overview of the state-of-the-art in fault detection methods. This paper describes the two most common methods of fault detection, isolation, and reconfiguration (FDIR) as hardware redundancy and analytical redundancy. Analytical redundancy relies on a software model of the system for estimation and fault-detection and software-defined responses to various system states (including component failures) to reduce overall system risk, which is the approach investigated here.

A common model for fault detection is based on an "Interacting Multiple Model" (IMM) as described in [8], in which the system is assumed to be in one of a set of modes, either faulted or nominal. Reference [8] uses this formulation to detect partial and total faults in reaction wheels in a spacecraft. Reference [9] extends this model to additionally detect faults in spacecraft sensors. This model is investigated in this work in combination with the Dual EKF, as expanded on below.

## IV. Linearized State Equations

Following from [10], we work with an exact linearization of the attitude dynamics equations through a nonlinear change of variables as follows:

First we define the attitude using a unit quaternion representation $\bar{q} = [q, q^0]^T$ where $[q, q^0] \in \mathbb{R}^3 \times \mathbb{R}$ and $q^T q + (q^0)^2 = 1$.

We then define the nonlinear forward transformation:

$$\begin{aligned}
\tilde{q} &= q \\
\tilde{\omega} &= \frac{1}{2}(q^0 J\omega + q \times (J\omega)) \\
\tilde{x} &= \begin{bmatrix} \tilde{q} \\ \tilde{w} \end{bmatrix} \\
\tilde{u} &= \frac{1}{2}(q^0 \dot{\omega} + q \times \dot{\omega}) - \frac{1}{4}\omega^T \omega q
\end{aligned}$$

$\dot{\omega}$ can be found from equation 1 as:

$$\dot{\omega} = J^{-1}(-\omega^\times J\omega + T(u \cdot \theta))$$

Where $a^\times$ denotes the skew-symmetric matrix of vector $a$

This results in the Linear Dynamics:

$$\dot{q} = \tilde{\omega} \qquad\qquad \dot{\tilde{\omega}} = \tilde{u}$$

or:

$$\dot{\tilde{x}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \tilde{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tilde{u} \tag{2}$$

$$\dot{\tilde{x}} = f(x) = A\tilde{x} + B\tilde{u} \tag{3}$$

The inverse transformation is then defined by:

$$q = \tilde{q} \tag{4}$$

$$\omega = \frac{2}{q^0}((q^0)^2 \tilde{w} + \tilde{q}\tilde{q}^T \tilde{w} + q^0 \tilde{q} \times \tilde{w}) \tag{5}$$

Here we note that the coefficient on the inverse transformation for $\omega$ includes $1/q^0$, creating a singularity at $q^0 = 0$. In other words, this transformation is only valid as long as the domain on which the dynamics are evaluated restrict the quaternion to a single half-plane. However, we can avoid the singularity simply by taking our initial orientation at any timestep to have $q^0 = 1$ through a shift of reference frame. By then maintaining a record of the relative quaternion between our current orientation and the inertial frame, we can retrieve our orientation in the inertial frame at any time. Therefore, as long as we we restrict our system and timestep to angular velocities low enough that our trajectory will not cross the half-plane during a single timestep, this singularity will not be encountered.

For this work, a first-order Euler discretization of these equations is used for both simulation and filtering.

## V. The Dual Extended Kalman Filter

We would not expect to see significant coupling between the system dynamics and our engine parameters given the significantly different timescales on which these processes occur. Therefore for improved computational simplicity and model stability we choose to implement a Dual Extended Kalman Filter (DEKF) for simultaneous state estimation and parameter estimation. As described in [11], the DEKF incorporates two Kalman Filters operating in parallel. The first estimates the system state while the second estimates the parameter vector. We will refer to these as the state filter and parameter filter respectively.

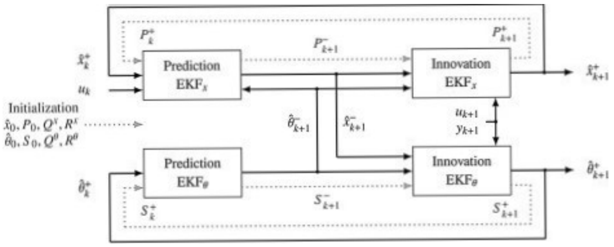

Fig. 1. Diagram of DEKF operation, with the state filter operating above and the parameter filter below [12].

These filters share information at two points. First, the predicted parameters $\theta_{t+1|t}$ ($\hat{\theta}_{k+1}^-$ in the diagram) are used in the prediction step of the state filter (and the update if relevant, but in our application the update is independent of $\theta$). Second, the predicted state $x_{t+1|t}$ ($\hat{x}_{k+1}^-$ in the diagram) is used in the update step of the parameter filter. This means our state is propagated using our best guess of our parameters given our past measurements, inputs, and state estimates, and our parameter vector is updated using our most recent state estimate given our past measurements, inputs, and parameter estimates. Therefore we see that there is an interdependency created each timestep between both the state estimates and parameter estimates. However, as we are assuming both the state and parameter vectors evolve as Markovian processes, our overall estimate ends up evolving only as a function of our initial state estimate, our measurements $y$, and inputs $u$.

### A. Filter Equations

The Dual EKF Predict and Update steps can be found in Section III of [11]. For the most part they are equivalent to the traditional EKF equations for both state and parameter filters. The primary differences in our application can be found in the state mean prediction and parameter update equations. The state mean predict equation is simply changed from $x_{t+1|t} = f(x_{t|t})$ to $x_{t+1|t} = f(x_{t|t}, \theta_{t+1|t})$. Note there is no change in the state update equations in our application as the measurement function is independent of the parameter vector, making $\frac{\partial g(x,\theta_{t+1|t})}{\partial x} = \frac{\partial g(x)}{\partial x}$ as in the traditional EKF.

The largest change from the vanilla EKF comes in the parameter update equation, where what would normally be $\left.\frac{\partial g(\theta)}{\partial \theta}\right|_{\theta=\theta_{t+1|t}}$ becomes $\left.\frac{dg(\tilde{x}_{t+1|t},\theta)}{d\theta}\right|_{\theta=\theta_{t+1|t}}$. This expression must then be expanded as:

$$\frac{dg(\tilde{x}_{t+1|t},\theta)}{d\theta} = \frac{\partial g(\tilde{x}_{t+1|t},\theta)}{\partial \theta} + \frac{\partial g(\tilde{x}_{t+1|t},\theta)}{\partial \tilde{x}_{t+1|t}} \frac{d\tilde{x}_{t+1|t}}{d\theta}$$

$$\frac{d\tilde{x}_{t+1|t}}{d\theta} = \frac{\partial f(\tilde{x}_{t|t},\theta)}{\partial \theta} + \frac{\partial f(\tilde{x}_{t|t},\theta)}{\partial \tilde{x}_{t|t}} \frac{d\tilde{x}_{t|t}}{d\theta}$$

$$\frac{d\tilde{x}_{t|t}}{d\theta} = \frac{d\tilde{x}_{t|t-1}}{d\theta} - K_{x,t-1} \frac{dg(\tilde{x}_{t|t-1},\theta)}{d\theta}$$

Where $K_{x,t-1}$ is the state Kalman gain from the previous timestep. We can in fact use stored values from the previous timestep to represent all of $\frac{dx_{t|t-1}}{d\theta_{t-1}}$, $K_{x,t-1}$, and $\frac{dg(x_{t|t-1},\theta_{t-1})}{d\theta_{t-1}}$, and initialize each to 0 at $t = 0$.

In the context of our dynamics, this results in:

$$\frac{\partial \tilde{u}}{\partial \theta} = \frac{1}{2}(q^0 I_{3\times 3} + q^\times) J^{-1} T u$$

$$\frac{d\tilde{x}_{t|t}}{d\theta} = R_k^{k-1} \left( \frac{d\tilde{x}_{t|t-1}}{d\theta} - K_{s,k-1} \frac{dg(\tilde{x}_{t|t-1},\theta)}{d\theta} \right)$$

$R_k^{k-1}$ is the inverse of the previous step's rotation update

$$\left.\frac{dg(\tilde{x}_{t+1|t},\theta)}{d\theta}\right|_{\theta=\theta_{t+1|t}} = \Delta t C \left( A \frac{d\tilde{x}_{t|t}}{d\theta} + B \frac{\partial \tilde{u}}{\partial \theta} \right)$$

This value is then used in place of our traditional measurement Jacobian in the parameter update step.

## VI. The Interacting Multiple Model Filter

As mentioned above, the DEKF is not designed to capture rapid changes in parameter values, such as a step change in engine "efficiency" caused by an engine failure. Thus, if we would like to rapidly identify and compensate for thruster failures, a different approach is required. In this work we employ an Interacting Multiple Model (IMM) filter for this purpose.

The IMM filter simultaneously maintains a bank of filters, each of which assumes the system is operating in a given mode at any timestep, for a total of $N$ modes. The first mode is taken to be nominal operation, as filtered by the DEKF above, while the remainder assume that some fault condition has occurred. In this work we generally limit these fault conditions to that of any single thruster failure at a given time, corresponding to some $\theta_i = 0$, in our model to improve readability of plots and reduce runtime. However, this can be extended to include any fault condition or combination of conditions (eg. all combinations of thruster failures, partial thruster failures, etc.). All modes other than the nominal mode are maintained as traditional EKFs, although their parameters other than that of the faulty engine are updated with the DEKF values when the probability of this mode is above a threshold $\alpha = 0.99$.

The IMM assumes that the true system can be modelled by a Jump Markov Model, meaning that given the system is in mode $i$ at $t$, there is a probability $\pi_{ij}$ that the system will jump to mode $j$ at $t + 1$, where $\pi_{ij} \geq 0 \ \forall \ i, j, \ \sum_j \pi_{ij} = 1$. The model then proceeds to maintain a probability distribution over all of the modes corresponding to the estimated likelihood of

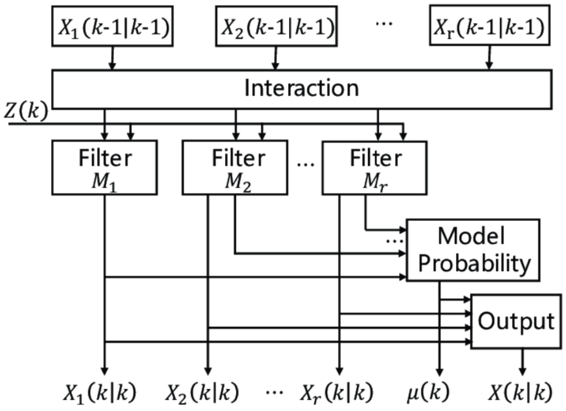

Fig. 2. Diagram of IMM operation [13].

the system being in a given mode at a given time. The full formulation of the filter as used here can be found in [8].

This probability distribution is updated at each time by running the predict and update steps of each filter using the most recent input and observation. A likelihood function for each mode $j$ is then computed for each model as given below:

$$L_{j,t+1} = \frac{1}{\sqrt{|(2\pi)S_{j,t+1}|}} \exp\left[-\frac{1}{2}\nu_{j,t+1}^T S_{j,t+1}^T \nu_{j,t+1}\right]$$

$$\nu_j = y_{t+1} - g(x_{j,t+1|t+1}) = \text{ Measurement Residual}$$

$$S_j = \left.\frac{\partial g(x)}{\partial x}\right|_{x_{j,t+1|t}} \Sigma_{j,t+1|t} \left(\left.\frac{\partial g(x)}{\partial x}\right|_{x_{j,t+1|t}}\right)^T + R$$
$$= \text{ Residual Covariance}$$

The posterior probability of each mode is then updated using Bayes' rule as:

$$\mu_{j,t+1} = \frac{\mu_{j,t+1|t} L_{j,t+1}}{\sum_1^N \mu_{j,t+1|t} L_{j,t+1}}$$

Where $\mu_{j,t+1|t} = \sum_{i=1}^N \pi_{ij}\mu_{i,t}$ is the prior probability given our jump Markov model of being in state $j$ at time $t+1$ and previous mode probabilities.

[9] proposes a modification to the likelihood function as:

$$L_{j,t+1} = \frac{1}{(2\pi)^{m/2}\sqrt{|S_{j,t+1}|}} \exp\left[-\beta\frac{1}{2}\nu_{j,t+1}^T S_{j,t+1}^T \nu_{j,t+1}\right]$$

Where $m = 6$ is the size of the attitude system state and $\beta$ is a free parameter which, when greater than 1 reduces the time to fault detection at the cost of increased false positives and noise, and when less than 1 increases the time to detection but reduces false positives.

## VII. Results and Discussion

To test each of these filters, they were run on a simulated system with relatively randomized but discrete (as would be used on pulsed attitude control thrusters) inputs to the thrusters as can be seen in Figure 3. The simulation parameters can be found in Table I. The true values of $\theta$ were not set to vary randomly according to $Q_\theta$ as this is not realistic, but followed some preset profiles as shown in each estimation figure.

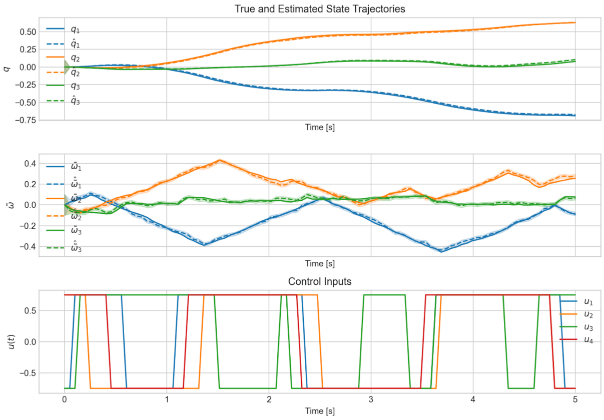

Fig. 3. True and estimated (with 95% confidence interval) state trajectories and control inputs for the test system over five seconds. This behavior was periodic with additive Gaussian noise in state and measurements.

TABLE I
Parameters used for simulation

| Parameter | Value |
|---|---|
| $J$ | $I_{3\times3}$ |
| $\Delta t$ | 0.005 s |
| $Q_x$ | $\begin{bmatrix}0 & 0\\ 0 & 0.02I_{3\times3}\end{bmatrix}$ |
| $Q_\theta$ | $10^{-7}I_{M\times M}$ |
| $R$ | $5\times10^{-3}I_{6\times6}$ |
| $\tilde{x}_{0\|0}$ | $\sim\mathcal{N}(\tilde{x}_0, 0.01I_{6\times6})$ |
| $\theta_{0\|0}$ | $\sim\mathcal{N}(\mathbf{1}, 0.1I_{M\times M})$ |

The spacecraft was configured with $M = 4$ thrusters such that each exerted unit torque at full thrust and efficiency around a principal spacecraft axis as given in Table II. Note that thrusters 1 and 3 exert identical torque under identical thrust.

TABLE II
Thruster torque vectors

| Thruster Number | Axis |
|---|---|
| **0** | +x |
| **1** | +z |
| **2** | +y |
| **3** | +z |

### A. Parameter Estimation with DEKF

The DEKF was tested with both constant and time varying efficiencies. In Figure 4, we see that in each case the estimate converges to closely follow the true parameter in less than ten seconds. We do see relatively significant noise in the mean estimate of each parameter, due to a combination of process and measurement noise, and note the relatively wide confidence interval for each estimate, despite the mean tracking quite closely. This demonstrates that as our parameter estimate is a tertiary measurement which is affected by any uncertainties

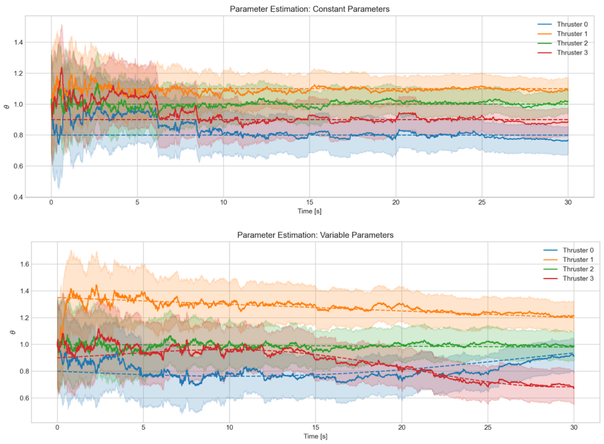

Fig. 4. True and estimated (with 95% confidence interval) parameter trajectories for the test system.

upstream in state and measurement, this estimate is highly sensitive to increases in upstream uncertainties.

A major limitation of this approach can be seen if the geometrically identical thrusters 1 and 3 are actuated on the same schedule as in Figure 5. In this case thrusters 1 and 3 are estimated to have identical efficiencies and follow an average path between the true efficiencies. In this case our parameters are not fully observable as the torque exerted in the $z$ direction is simply the sum of thrusters 1 and 3. We do however see that the variance is higher in each of these thrusters in this situation, meaning the true parameter is still captured at all times within the 95% confidence interval.

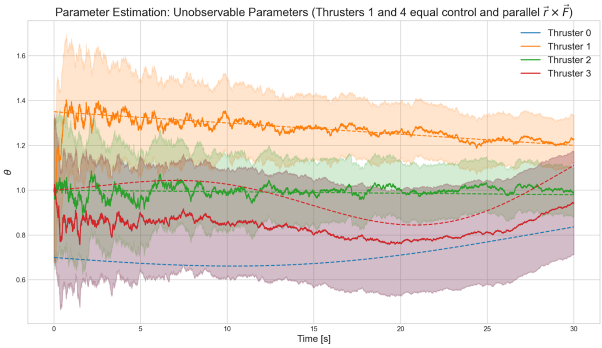

Fig. 5. True and estimated (with 95% confidence interval) parameter trajectories with identical control applied to thrusters 1 and 3.

The other major drawback to this method comes with a step change in parameters, as demonstrated in Figure 6. We see that it takes almost 10 seconds for the system to again determine the true parameter values with the faulted thruster, far longer than we'd like to take to react. We also see an error is propagated to thruster 3 due to its correlation with thruster 1.

### B. Fault Detection with IMM

The IMM was tested with similar failure cases to the DEKF, with a given thruster failing at 15s. The models were numbered as model 0 being the nominal DEKF, and each model $j = 1, 2, 3, 4$ corresponding to thruster $j - 1$ having failed. The transition model was given as $\pi_{ii} = 0.9995$, $\pi_{ij} = 0.0005 \ \forall \ i \neq j$ corresponding to a 99.95% chance of remaining in the current mode, and uniform probability of transitioning to any other mode otherwise. A value of $\beta = 0.25$ was used in all cases.

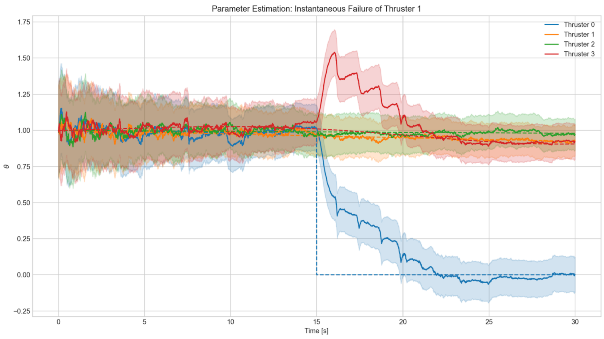

Fig. 6. True and estimated (with 95% confidence interval) parameter trajectories with thruster 1 failure at 15s.

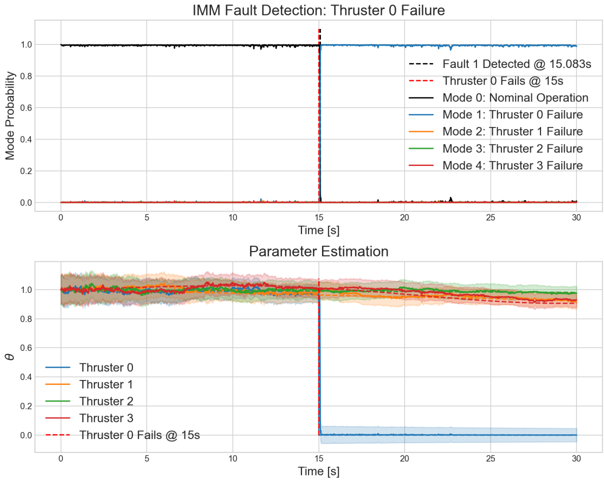

Fig. 7. Mode probabilities (above) and parameter estimates (below) for thruster 0 failure at 15s.

In Figure 7, the fault is detected and the system identifies it is most likely in mode 1 within 83 milliseconds of failure. Moreover, the parameters continue to be estimated accurately using the weighted estimate: $\hat{\theta}_i = \sum_{j=0}^{N} \mu_{j,t} \theta_{j,i,t|t}$.

In Figures 8 and 9 we see the parameter estimation and mode probabilities with a more drastic but still gradual variation in parameter vectors. Figure 8 shows the performance using our model with the DEKF, while Figure 9 uses a static EKF with nominal parameters as the nominal model. We see a drastic improvement in both the mode detection stability and parameter estimation with the DEKF model. This formulation can isolate gradual changes in parameters from step faults as each model maintains a relevant running estimate of all parameters not assumed to be faulted, thereby minimizing residuals due to gradual parameter shifts. The model without the DEKF very poorly estimates parameters as this is not a primary component of the system, and the large residuals due to parameter shifts cause significant noise an numerous false-positives as parameters gradually decrease.

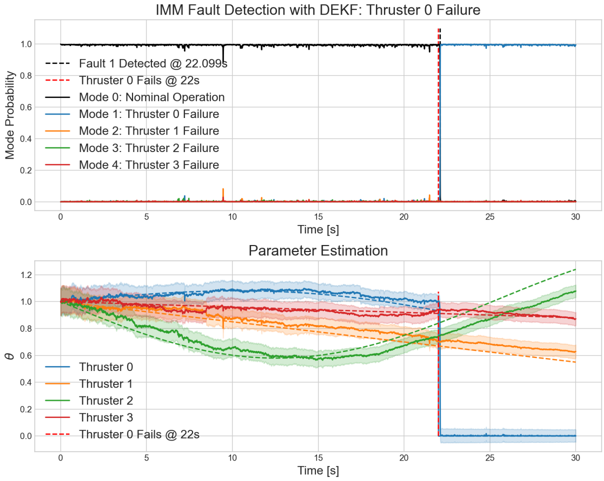

Fig. 8. Mode probabilities (above) and parameter estimates (below) for thruster 0 failure at 22s with strong parameter variation and our DEKF variable-parameter model for nominal operation.

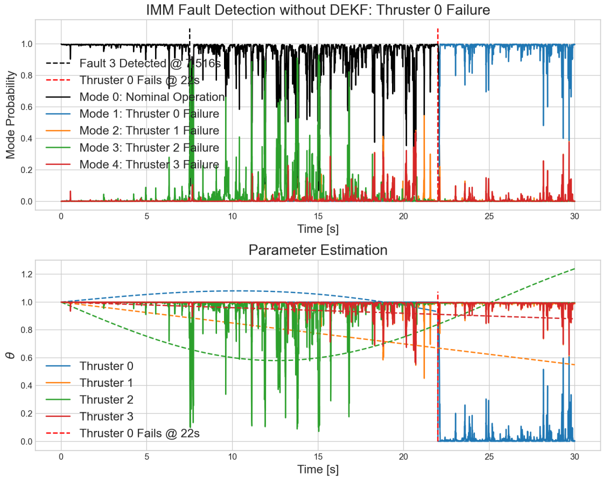

Fig. 9. Mode probabilities (above) and parameter estimates (below) for thruster 0 failure at 22s with strong parameter variation and an EKF constant-parameter model for nominal operation.

Finally, we investigate the case of multiple thruster failure, in which we add six more EKFs corresponding to any combination of two thrusters having failed. The result can be seen in Figure 10, where we quickly identify each fault, with slightly more delay on the second. This model does need 150% more EKFs than the single-failure model, requiring far more computation. This could be alleviated by assuming only one thruster fails at a time, and then instantiating multiple failure filters if a single thruster is determined to have failed (eg. if the mode probability of mode 1 exceeds a threshold, instantiate three EKFs for failure modes in combination with thruster 0).

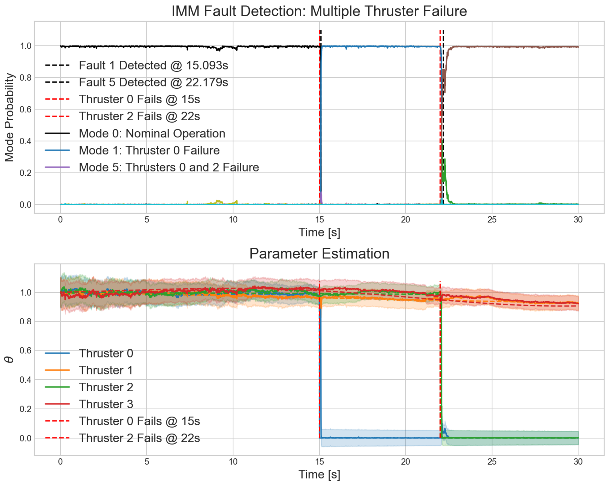

Fig. 10. Mode probabilities (above) and parameter estimates (below) for thruster 0 failure at 15s followed by thruster 2 failure at 22s.

## VIII. Conclusion and Future Work

This work developed a method to estimate spacecraft attitude control thruster efficiencies from input controls and attitude state measurements using a DEKF, which showed strong convergence to the true system parameters in the case of a fully observable system and slow changes in parameters. This estimation was then incorporated into an IMM filter to add the capability to detect step changes in parameters in the particular case of thruster failure. This method demonstrated the ability to detect thruster failures on a timescale 100 times faster than the DEKF model, and continuously maintain an accurate estimate of system parameters with minimal disturbance when entering the fault condition. Additionally, the incorporation of the DEKF in the IMM was shown to greatly increase stability in fault detection.

This work assumed that the estimation of the system attitude was an input measurement in this algorithm. However it is more likely this is a value computed by some other onboard filter. This work could be expanded to either incorporate this estimation directly in the DEKF, or make use of the covariance matrix output by the attitude estimation filter, rather than just sampling the mean.

Additional avenues for exploration include incorporating additional torques not provided by the thrusters (eg. reaction wheels and gravitational torque) and adding multiple layers to the IMM filters to increase sensitivity without increased noise.

# References

[1] E. Lefferts, F. Markley, and M. Shuster, "Kalman filtering for spacecraft attitude estimation," 20th Aerospace Sciences Meeting, 1982.

[2] S.-W. Kim, M. Abdelrahman, S.-Y. Park, and K.-H. Choi, "Unscented Kalman filtering for spacecraft attitude and rate determination using magnetometer," Journal of Astronomy and Space Sciences, vol. 26, no. 1, pp. 31–46, 2009.

[3] J. Crassidis and F. L. Markley, "Unscented filtering for spacecraft attitude estimation," AIAA Guidance, Navigation, and Control Conference and Exhibit, 2003.

[4] H.-S. Myung, K.-L. Yong, and H.-C. Bang, "Unscented Kalman filtering for hybrid estimation of spacecraft attitude dynamics and rate sensor alignment," Advances in Spacecraft Technologies, 2011.

[5] Y. Cheng and J. Crassidis, "Particle filtering for sequential spacecraft attitude estimation," AIAA Guidance, Navigation, and Control Conference and Exhibit, 2004.

[6] F. Markley, "Attitude determination and parameter estimation using vector observations," Astrodynamics Conference, 1988.

[7] M. Babaei, J. Shi, and S. Abdelwahed, "A survey on fault detection, isolation, and reconfiguration methods in Electric Ship Power Systems," IEEE Access, vol. 6, pp. 9430–9441, 2018.

[8] N. Tudoroiu and K. Khorasani, "Satellite fault diagnosis using a bank of interacting Kalman filters," IEEE Transactions on Aerospace and Electronic Systems, vol. 43, no. 4, pp. 1334–1350, 2007.

[9] J. Lee and C. G. Park, "Cascade filter structure for sensor/actuator fault detection and isolation of satellite attitude control system," International Journal of Control, Automation and Systems, vol. 10, no. 3, pp. 506–516, 2012.

[10] T. Dwyer. Exact nonlinear control of large angle rotational maneuvers. IEEE Transactions on Automatic Control, 29(9):769–774, 1984.

[11] A. Popovici, P. Zaal, and D. M. Pool, "Dual extended kalman filter for the identification of time-varying human manual control behavior," AIAA Modeling and Simulation Technologies Conference, 2017.

[12] N. Wassiliadis, J. Adermann, A. Frericks, M. Pak, C. Reiter, B. Lohmann, and M. Lienkamp, "Revisiting the dual extended Kalman filter for battery state-of-charge and state-of-health estimation: A use-case life cycle analysis," Journal of Energy Storage, vol. 19, pp. 73–87, 2018.

[13] T. Zhang, X. Ji, Z. Zhuang, and W. Xu, "Jamcatcher: A Mobile Jammer localization scheme for advanced metering infrastructure in Smart Grid," Sensors, vol. 19, no. 4, p. 909, 2019.