AAS 25-585

FDIR FOR AUTONOMOUS SPACE SYSTEMS FOR ANOMALIES AND CYBERATTACKS

Matthew M. Wittal[*] and Michael Czernec[†]

Fault detection, identification, and remediation (FDIR) is a challenging task for any space system. Faults are not necessarily binary in nature, and several different faults may all create the same symptoms or bad performance. This is compounded when the consideration of cyberattacks is likewise added, and further made daunting when the system in question is intended to perform its tasks autonomously. In this work, a spacecraft in the tug-cargo configuration is simulated during an RPOD maneuver. During this maneuver, several zero-dynamics attacks are simulated that attack the spacecraft's sensors, all of which first cause the spacecraft either fail catastrophically, or initiate an auto-abort. Then, a discriminative unscented Kalman filter previously used to identify unknown dynamics is used instead to identify faulty or misleading sensors, and successfully perform the docking maneuver in spite of these failures. The simulated spacecraft is simulated in multi body problem using the rigid body formulism on special Euclidean group $\mathsf{SE}(3)$. Additionally, Morse-Lyapunov control is applied considering constrained, distributed control among reaction control system thrusters with saturation limits and duty cycle constraints.

INTRODUCTION

A permanent human lunar settlement, as a goal of the Artemis program, will require extensive logistics support during the early years, and presumably throughout its sustained phase of operation. In addition to the Moon2Mars architecture that includes logistics support all to Mars and beyond, these challenges further demand the creation of robust autonomy in spacecraft. These spacecraft must perform reliably in the absence of direct human control, lack of availability of ground stations communications, in the presence of rapidly or frequently changing configurations, and even in the presence of anomalies including thruster and sensor malfunctions. This suggests that a notional navigation and control scheme must be robust to a wide range of scenarios, configurations, and anomalies such as stuck thrusters (i.e., thrusters that remain on despite being commanded to stop firing), unmodeled dynamics, and potential sabotage such as cyberattacks. There is an expectation that autonomy will be capable of handling such scenarios during long voyages between planets where communication is inhibited by long light-time delays and during rendezvous, proximity operations, and docking around distant celestial bodies. While previous work has demonstrated the conceptual ability of a spacecraft to generate thruster firing histories for both arbitrary thruster configurations [1], arbitrary mass properties [2], and abort scenarios [3], these simulations have not been performed in the presence of sensor errors or cyberattacks.

In pursuit of this level of autonomy, much work has been done to examine optimal path-planning and autonomous guidance for entry, descent, and landing (EDL) applications [4]. Furthermore, research has explored the behavior of thrusters and how control schemes might overcome or plan for true thruster behavior as compared to the modeled thruster behavior [5]. Similarly, this work depends on a thorough understanding of the spacecraft configuration, even if the configuration is arbitrary. However, it allows for state uncertainty (in the presence of both process and measurement noise), control uncertainty, and off-nominal thruster behavior. This demonstrates tolerance to changing parameters, so long as those changes are known.

[*]GN&C Engineer, Deep Space Logistics Project Office, NASA Kennedy Space Center, FL 32899, AAS Member
[†]Flight Software Engineer, Aegis Aerospace, Deep Space Logistics Project Office, NASA Kennedy Space Center, FL 32899

The use of Lie group $\mathsf{SE}(3)$ enables the consideration of coupled motion even in the presence of noise without encountering singularities, unwinding, or nonuniqueness issues present in other attitude parameterization sets [6–8]. There has been some recent work focusing on constrained control and distributed control on $\mathsf{SE}(3)$ and its tangent bundle $\mathsf{TSE}(3)$, but these algorithms do not consider autonomous thruster firing nor autonomous thruster firing histories or constraints generated by duty cycles, such as is the case in reaction control system (RCS) thrusters [8–11]. Furthermore, it has been demonstrated in [8] that the use of this controller in the presence of noise when interpreted through the use of an unscented Kalman filter (UKF) first presented in [12, 13] is robust to measurement and process uncertainty. This same controller and filter have also been used to demonstrate the stability of launch systems in the presence of large uncertainties [10, 14].

However, little has been done to explore the robustness of nonlinear controllers - especially those in SE(3) - in the presence of cyberattacks, specifically zero-dynamics Attacks (ZDA). ZDAs are a sophisticated subclass of deception attacks targeting Networked Control Systems (NCS). These attacks exploit the internal dynamics of an NCS, particularly focusing on non-minimum phase systems by leveraging unstable zeros. The stealthy nature of ZDAs makes them a significant threat, as they can cause substantial damage without immediate detection. ZDAs are designed to remain undetected by exploiting the system's internal dynamics. The attack input grows geometrically without causing apparent changes in the system's output. These attacks primarily target non-minimum phase systems with unstable zeros that can be exploited [15–26].

In this paper, the algorithm described demonstrates that a spacecraft can autonomously generate a time-optimal thruster firing history in the presence of noisy measurements and off-nominal or constrained thruster configurations, provided that the available thrusters have some access to all degrees of freedom (DoF). A case in which suboptimal thruster configurations restrict motion is considered. The spacecraft configuration is defined within the $\mathsf{SE}(3)$ framework, which allows the consideration of both the translational and rotational components of its thruster configuration simultaneously and aids in generating precise guidance protocols even in the presence of off-nominal or constrained thruster configurations. Finally and most significantly, a method for fault detection, identification, and remediation (FDIR) is presented that includes both malfunctioning thrusters and malfunctioning sensors as a result of a cyberattack, specifically a ZDA.

PRELIMINARIES

Formulation on Special Euclidean Group $\mathsf{SE}(3)$

The pose of a rigid body is expressed in $\mathsf{SE}(3)$ as

$$g = \begin{bmatrix} R & r \\ 0_{1\times 3} & 1 \end{bmatrix} \in \mathsf{SE}(3) \tag{1}$$

where $R \in \mathsf{SO}(3)$ is the rotation matrix from the body frame $\mathcal{B}$ to the inertial frame $\mathcal{N}$ and r is the position vector in the inertial frame. The rigid body's velocity can be defined using an augmented velocity vector $\mathbb{V}$ as

$$\mathbb{V} = \left[\omega^T, v^T\right]^T \in \mathbb{R}^6 \tag{2}$$

such that $\omega \in \mathbb{R}^3$ is the angular velocity in the body frame $\mathcal{B}$ and $v \in \mathbb{R}^3$ is the translational velocity relative to the inertial frame and expressed in the body frame. The dynamic equations for a rigid body with can be expressed in $\mathsf{TSE}(3)$ as

$$\dot{g} = g\mathbb{V}^\vee \tag{3a}$$

$$\dot{\mathbb{V}} = \mathbb{I}^{-1}\left(\mathrm{ad}^*_{\mathbb{V}}\mathbb{I}\mathbb{V} + \tau + u\right) \tag{3b}$$

For the purpose of developing a formulation for relative motion or tracking controller on $\mathsf{SE}(3)$, the following expression is introduced:

$$\epsilon_g = g_d^{-1}g = \begin{bmatrix} R_d & \epsilon_r \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_d^T R & R_d^T\left(r - r_d\right) \\ 0 & 1 \end{bmatrix} \in \mathsf{SE}(3) \tag{4}$$

$$\epsilon_{\mathbb{V}} = \mathbb{V} - \mathrm{Ad}_{\epsilon_g}\mathbb{V}_d \in \mathbb{R}^6$$

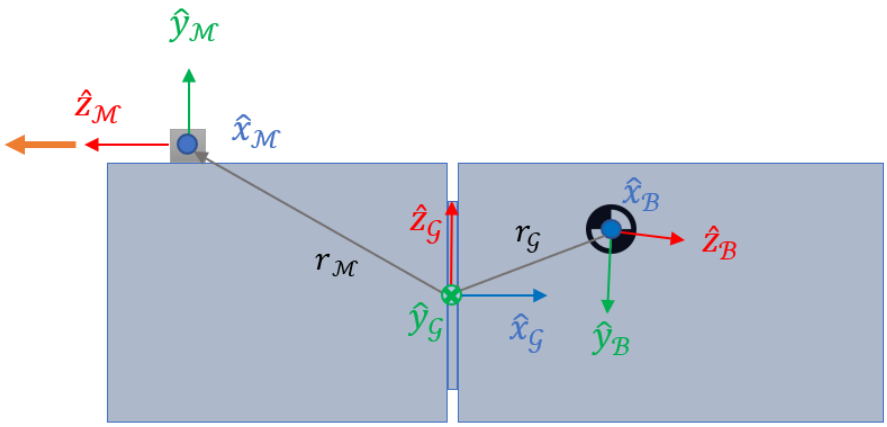

Figure 1: An illustration of the terminology associated with various frames within the spacecraft configuration.

where the subscript 'd' denotes the desired or modeled state or parameter. The relative dynamics of a rigid body with a changing center of mass and moments of inertia in $\mathsf{SE}(3)$ are represented as

$$\dot{\epsilon}_g = \epsilon_g \epsilon_{\mathbb{V}}^{\vee} \in \mathfrak{se}(3) \tag{5a}$$

$$\begin{aligned} \dot{\epsilon}_{\mathbb{V}} &= \frac{d}{dt}\left(\hat{\mathbb{V}} - \mathrm{Ad}_{\epsilon_g}\mathbb{V}_d\right) \\ &= \dot{\hat{\mathbb{V}}} - \mathrm{ad}_{\epsilon_{\mathbb{V}}}\mathrm{Ad}_{\epsilon_g}\mathbb{V}_d - \mathrm{Ad}_{\epsilon_g^{-1}}\dot{\mathbb{V}}_d \end{aligned} \tag{5b}$$

where

$$\mathbb{I} = \begin{bmatrix} J & 0_{3\times3} \\ 0_{3\times3} & mI_3 \end{bmatrix} \in \mathbb{R}^{6\times6}, \tag{6}$$

and the coadjoint operator is defined as

$$\mathrm{ad}_{\mathbb{V}}^{*} = \mathrm{ad}_{\mathbb{V}}^{T} = \begin{bmatrix} -\omega^{\times} & -v^{\times} \\ 0_{3\times3} & -\omega^{\times} \end{bmatrix} \in \mathbb{R}^{6\times6} \tag{7}$$

In Eq. (5), $\tau \in \mathbb{R}^6$ represents the impressed dynamics acting on the system expressed in the $\mathcal{B}$ frame. Furthermore, the augmented vector of control inputs is given by $u = \left[L^T, f^T\right]^T \in \mathbb{R}^6$ where $f \in \mathbb{R}^3$ and $L \in \mathbb{R}^3$ represent the control forces and torques, respectively.

Consider an actuator $g_{\mathcal{M}} \in \mathsf{SE}(3)$ where $\mathcal{M}$ denotes the frame of a motor or actuator with respect to the rigid body. This frame has a pose defined as

$$g_{\mathcal{M}} = \begin{bmatrix} R_{\mathcal{B}/\mathcal{M}} & r_{\mathcal{M}} \\ 0_{1\times3} & 1 \end{bmatrix} \in \mathsf{SE}(3) \tag{8}$$

where $R_{\mathcal{B}/\mathcal{M}}$ is the attitude of the thruster in body frame $\mathcal{B}$, and $r_{\mathcal{M}}$ is the position of the body with respect to the thruster. The relative motion dynamics expressed above may be extended to describe the pose of a thruster relative to a central body in either a geometric frame $\mathcal{G}$, which is advantageous for unknown or arbitrary mass properties, or relative to a standard body frame $\mathcal{B}$. This may be expressed as $g_{\mathcal{G}/\mathcal{M}}$ or $g_{\mathcal{B}/\mathcal{M}}$, respectively, as illustrated in Figure 1. Take the linear adjoint operator of this relative configuration to yield

$$\mathrm{Ad}_{g_{\mathcal{M}}} = \begin{bmatrix} R_{\mathcal{B}/\mathcal{M}} & 0_{3\times3} \\ r_{\mathcal{M}}^{\times} R_{\mathcal{B}/\mathcal{M}} & R_{\mathcal{B}/\mathcal{M}} \end{bmatrix} \in \mathbb{R}^{6\times6} \tag{9}$$

The transpose of the adjoint operator is defined as

$$\mathrm{Ad}^T_{g_{\mathcal{M}}} = \begin{bmatrix} R^T_{\mathcal{B}/\mathcal{M}} & -R^T_{\mathcal{B}/\mathcal{M}} r^\times_{\mathcal{M}} \\ 0_{3\times 3} & R^T_{\mathcal{B}/\mathcal{M}} \end{bmatrix} \in \mathbb{R}^{6\times 6} \tag{10}$$

And finally,

$$\mathrm{Ad}^T_{g^{-1}_{\mathcal{M}}} = \begin{bmatrix} R_{\mathcal{B}/\mathcal{M}} & R_{\mathcal{B}/\mathcal{M}} \left(R^T_{\mathcal{B}/\mathcal{M}} r_{\mathcal{M}}\right)^\times \\ 0_{3\times 3} & R_{\mathcal{B}/\mathcal{M}} \end{bmatrix} \in \mathbb{R}^{6\times 6} \tag{11}$$

$\mathsf{TSE}(3)$-Based Unscented Kalman Filter

In this work, the UKF on $\mathsf{SE}(3)$ first introduced by [12] and extended to tracking control by [8] is used in this work. This filter works by first taking in initial state estimates $\hat{g}_{k|k} \in \mathsf{SE}(3)$ and $\hat{\mathbb{V}}_{k|k}$ and covariance $P_{k|k} \in \mathbb{R}^{p\times p}$ where p is the number of state variables. An initial guess is propagated to generate an *a priori* state estimate using some nonlinear dynamical function $f(g, \mathbb{V}, \eta_p)$, where $\eta_p \in \mathbb{R}^p$ is the process noise, in order to generate $\hat{g}_{k+1|k}$ and $\hat{\mathbb{V}}_{k+1|k}$. The UKF uses sigma points serve as small perturbations to $\hat{g}_{k|k}$ and $\hat{\mathbb{V}}_{k|k}$ in order to obtain an updated covariance $P_{k+1|k}$. The sigma points are expressed through the matrix $\chi_p \in \mathbb{R}^{p\times(2p+1)}$ of $2p+1$ sigma-point column vectors $\chi_{p,i}$ and the matrix $\chi_q \in \mathbb{R}^{q\times(2q+1)}$ of $2q+1$ sigma column vectors $\chi_{q,i}$. Additionally, the weight matrices are defined as

$$\begin{aligned} W^{(m)}_p &= [W^{(m)}_{p,0},\ W^{(m)}_{p,1},\ \cdots,\ W^{(m)}_{p,4p}] \\ W^{(c)}_p &= [W^{(c)}_{p,0},\ W^{(c)}_{p,1},\ \cdots,\ W^{(c)}_{p,4p}] \end{aligned} \tag{12}$$

where

$$\begin{aligned} W^{(m)}_{p,0} &= \frac{\lambda_p}{\lambda_p + p},\ W^{(c)}_{p,0} = \frac{\lambda_p}{\lambda_p + p} + (1 - \alpha^2 + \beta) \\ W^{(m)}_{p,i} &= W^{(c)}_{p,i} = \frac{1}{2(\lambda_p + p)} \qquad (i = 1, 2, \cdots, 4p) \end{aligned} \tag{13}$$

This same procedure is repeated for $W^{(m)}_q$ and $W^{(c)}_q$ except for using the corresponding $\lambda_q = (\alpha^2 - 1)q$ instead of $\lambda_p = (\alpha^2 - 1)p$. The true dynamics are then propagated using the variational integrator first introduced by [6] to take initial true states g_k and $\mathbb{V}_k$ to g_{k+1} and $\mathbb{V}_{k+1}$. These newly generated states are passed through a measurement model $h(g_{k+1}, \mathbb{V}_{k+1}, \eta_m) \in \mathbb{R}^q$ where $\eta_m \in \mathbb{R}^q$ is the measurement noise and q is the dimension of the measurement model. If all states are observable, then $p = q$. This new measured state is used to generate an updated Kalman gain which is, in turn, used to update the *a priori* state estimates to the *a posteriori* state estimates in the update step as $\hat{g}_{k+1|k} \to \hat{g}_{k+1|k+1}$ and $\hat{\mathbb{V}}_{k+1|k} \to \hat{\mathbb{V}}_{k+1|k+1}$. Finally, the *a posteriori* error covariance matrix is computed as

$$P_{k+1|k+1} = P_{k+1|k} - K_{k+1} P_{zz,k+1} K^T_{k+1} \tag{14}$$

A more detailed description of the implementation of the $\mathsf{TSE}(3)$-based UKF can be found in [8].

Discriminative Unscented Kalman Filter

The extension of the UKF above to the field of discriminative methodology is based on the discriminative Kalman filter (DKF) mentioned in [27]. However, the DKF has one major advantage: it acts as the $\mathsf{TSE}(3)$ and $\mathsf{T^2SE}(3)$ and not directly on the manifold. This means that it is incapable of filtering pose directly and must use information only on the tangent spaces, sometimes resulting in erratic estimated values for position and attitude. To solve this, the discriminative methodology used in the DKF first presented in [27] is applied in an additional step during the UKF pose estimation process. Generally, this works by examining the residuals between the filtered, measured state and the predicted state using only the dynamic model and using those to update the assumptions about the dynamical environment. In this work, model complexity is incrementally

increased and with enough measurements, an increasingly large vector of environmental variables can be known. In this work, a modification of this approach is applied to understanding sensor anomalies and thruster aberrations, explained in detail in Algorithm 1, where residual analysis between current estimated state $\hat{g}_{k+1|k+1}$ and measurements y_k.

Algorithm 1 Discriminative Unscented Kalman Filter for State and Dynamic Environment Estimation on $\mathsf{SE}(3)$

Initialize

$\hat{\mu}_0, g_0, \mathbb{V}_0, \hat{g}_0, \hat{\mathbb{V}}_0, P_0, W$

Begin Main Iteration Loop:

for $k = 1, 2, ...$ **do**

1: Predict
$[\hat{g}_{k+1|k}, \mathbb{V}_{k+1|k}, P_{k+1|k}] = \hat{f}(\hat{g}_{k|k}, \mathbb{V}_{k|k}, P_{k|k}, \hat{\mu}_k)$

2: Integrate
$[g_{k+1}, \mathbb{V}_{k+1}] = f(g_k, \mathbb{V}_k)$

3: Measure
$y_k = h(g_{k+1}, \mathbb{V}_{k+1})$

4: Update State
$[\hat{g}_{k+1|k+1}, \mathbb{V}_{k+1|k+1}, P_{k+1|k+1}] = h(\hat{g}_{k|k}, \mathbb{V}_{k+1|k}, P_{k+1|k}, y_k)$

5: Update Model
$\hat{\mu}_{k+1} = \lceil(t, \hat{g}_{k+1|k+1}, y_k, \hat{\mu}_k)$

6: Update Variables
$\hat{g}_{k|k} = \hat{g}_{k+1|k+1}$ $\hat{\mathbb{V}}_{k|k} = \hat{\mathbb{V}}_{k+1|k+1}$

end

Estimation-Based Morse-Lyapunov Control

The $\mathsf{TSE}(3)$-based UKF described was implemented in [9, 10, 14] using the Morse-Lyapunov (M-L) controller from [28]. The controller is defined as

$$u_0 = -\mathbb{I}^{-1}\left(\mathrm{ad}^*_{\hat{g}}\,\mathbb{I}\hat{\mathbb{V}} - \mathrm{Ad}_{\epsilon_g}\,\mathrm{ad}^*_{g_d}\,\mathbb{I}\mathbb{V}_d\right) - \epsilon_\tau + \mathrm{ad}_{\epsilon_\mathbb{V}}\,\mathrm{Ad}_{\epsilon_g}\,\mathbb{V}_d - \mathbb{I}K_1\dot{l} - \mathbb{I}K_2\psi - \mathbb{I}\kappa[0^T_{1\times 3}\,(\epsilon_R\epsilon_r)^T]^T \in \mathbb{R}^6 \quad (15)$$

where

$$\epsilon_\tau = \mathbb{I}^{-1}\hat{\tau} - \mathrm{Ad}_{\epsilon_g}\mathbb{I}^{-1}\tau_d \quad (16)$$

is the relative external forces and torques between the estimated and desired models. The gains $K_1 = \mathrm{blkdiag}(k_{11}I_3, k_{12}I_3) \in \mathbb{R}^6 \geq 0$ and $K_2 = \mathrm{blkdiag}(k_{21}I_3, k_{22}I_3) \in \mathbb{R}^{6\times 6} \geq 0$ are positive definite diagonal gain matrices and κ is a scalar gain coefficient. The backstepping function is defined as

$$\psi = \epsilon_\mathbb{V} + K_1 l(\epsilon_g) \quad (17)$$

where

$$l(\epsilon_g) = [s^T(\epsilon_R), \epsilon_r^T]^T \quad (18)$$

such that

$$s(\epsilon_R) = \sum_{i=1}^{3} a_i \left(\epsilon_R^T \check{e}_i\right)^\times \check{e}_i \quad (19)$$

and $a \in \mathbb{R}^3$ is a matrix of tunable gain values, where $1 \leq a_1 < a_2 < a_3$ and $\check{e}_i$ $(i = 1, 2, 3)$ are unit vectors aligned with the vehicle's principal axes. The initial control force $u_0 \in \mathbb{R}^6$ is generated based on the desired control input from the M-L controller, and then constrained based on the spacecraft configuration.

The process for generating control on estimated states, constraining that control based on the spacecraft configuration, and then generating the next estimated state is described in Algorithm 2. The numerical stability of this method has been demonstrated here and in previous work [1, 10, 14], but further work is needed to prove the analytical stability of the controller and UKF.

Algorithm 2 Estimation and Control Sequence

Initializing:

Define Initial Conditions $g_0, \mathbb{V}_0, P_0$

Begin Main Iteration Loop:

for $k = 1, 2, ...$ **do**

1: Estimate Control
$u_0 = u_0(\hat{g}_{k|k}, \hat{\mathbb{V}}_{k|k})$

2: Optimally Distribute Control Based on Spacecraft Configuration
$c = \min \left((\tilde{U}c - u_0)^T (\tilde{U}c - u_0) + kc^T c \right)$

3: Impose duty cycle by scaling c and recalculate control vector
$c_{dc} = \frac{c}{\max(c)} d_c$
$u_{\text{com}} = \tilde{U} c_{dc}$
$u_k = u_{\text{com}}(1 + \mathcal{N}(0, \eta_T))$

4: Predict Based on Model
$[\hat{g}_{k+1|k}, \hat{\mathbb{V}}_{k+1|k}, P_{k+1|k}] = f(\hat{g}_{k|k}, \hat{\mathbb{V}}_{k|k}, P_{k|k}, u_{act})$

5: Propagate True State
$[g_{k+1}, \mathbb{V}_{k+1}] = f(g_k, \mathbb{V}_k, u_k)$

6: Update Based on Measurements
$[\hat{g}_{k+1|k+1}, \hat{\mathbb{V}}_{k+1|k+1}, P_{k+1|k+1}] = h(\hat{g}_{k+1|k}, \hat{\mathbb{V}}_{k+1|k}, P_{k+1|k}, u_{act})$

7: Update Variables
$g_{k+1} = g_k$
$\mathbb{V}_{k+1} = \mathbb{V}_k$
$\hat{g}_{k|k} = g_{k+1|k+1}$
$\hat{\mathbb{V}}_{k|k} = \hat{\mathbb{V}}_{k+1|k+1}$
$P_{k|k} = P_{k+1|k+1}$

end

AUTONOMOUS, DISTRIBUTED, CONSTRAINED CONTROL WITH SATURATION LIMITS

Thruster Configuration Definitions for Autonomous Distributed Control

For a distributed control scheme within the $\mathsf{SE}(3)$ framework, consider an arbitrary control input in the body frame $u_0 \in \mathbb{R}^6$. It is desired to determine how this control input is distributed about various actuators and how it determines the gimbaling of those actuators. To distribute control among arbitrary configurations of static and gimbaling thrusters, the components of such a control scheme must first be considered. The static thruster does not move, but when located away from the center of mass has contributions in both translational and rotational motion. The gimbaling thruster may contribute to either or both translational and rotational motion, depending on gimbal limits. Additionally, gimbaling thrusters have constraints on this contribution within the time domain due to imposed gimbal rate limits.

The total translational and rotational control of the central body is assessed by assembling a matrix of the control vectors with a coefficient describing the throttling or duty cycle of each thruster. The method of distributing the control input to the thrusters as presented as an optimization problem such that

$$u = \sum_{n=1}^{N} u_n c_n = \sum_{n=1}^{N} \text{Ad}^T_{g^{-1}_{\mathcal{M},n}} T_{\mathcal{M},n} c_n = \sum_{n=1}^{N} \text{Ad}^T_{g^{-1}_{\mathcal{M},n}} T_{\mathcal{M},n} c_{dc,n} \in \mathbb{R}^6 \tag{20}$$

which could be written as

$$u = \tilde{U}c \in \mathbb{R}^6 \mid \tilde{U} := [u_1, u_2, ..., u_N] \in \mathbb{R}^{6\times N} \tag{21}$$

where $\tilde{U}$ is the mapping between individual thrusters and the body frame and c is the duty cycle that best matches the optimal distributed control u. The optimal coefficients for each thruster determined by Eq. (20) may be obtained by solving the optimization problem of the form

$$u^* = \min \left((\tilde{U}c - u_0)^T (\tilde{U}c - u_0) + kc^T c \right) \tag{22}$$

by finding the optimal value of c such that $k \in \mathbb{R}$ is a scalar gain that tunes the deadband region to prevent chattering. To describe the required ratio of duty cycles required to compensate for a maneuver. To implement a guidance algorithm, the above algorithm is extended into the time domain, imposing known constraints on thruster duty cycles. Because a given spacecraft configuration could be configured in a suboptimal way, residual motions about any axis must be given a threshold to avoid saturating the controller.

Saturation Limits

Various methods of imposing saturation limits were considered during the development of this algorithm. It was found that imposing saturation limits on the control vector before distributing thrust was ineffective. Furthermore, imposing saturation limits within the optimization algorithm itself was likewise problematic, resulting in all thrusters being on at all times. Instead, an ideal solution of normalizing the vector of thruster ontime coefficients, c demonstrated good behavior. This can be expressed as

$$c_{dc} = \frac{c}{\max(c)} d_c \tag{23}$$

where d_c is the maximum duty cycle during any given time step. For example, for a thruster that is capable of being on 100% of the time during a 1-second thruster firing period during a 1-second time step, $d_c = 1$. For a thruster that can fire at 2 Hz with a full duty cycle of 0.1 second during a 1-second time step, then $d_c = 0.2$. With this, the total control force acting on a given spacecraft configuration with N thrusters is

$$u_{\text{com}} = \tilde{U} c_{dc} \tag{24}$$

where u_{com} is the final actuator value to be commanded to the thrusters and c_{dc} is the optimal value of c solved for in Eq. (22).

FAULT DETECTION, IDENTIFICATION, AND REMEDIATION USING DUKF

In previous work, a stuck thruster at 1 kilometer was simulated and an abort scenario was initiated. In this work, that logic is extended to not only identify the anomaly and initiate an abort but to identify the nature of that anomaly and overcome it in order to still complete a successful docking. Similar to previous work, the residuals between the commanded control force and the measured control forces are examined. To do this, two variables are defined for the commanded control forces and torques u_{com} and the true control forces and torques $\hat{c}_{\text{com}}$ as

$$\begin{aligned} u_{\text{com}} &= \tilde{U} c_{\text{com}} \\ u_{\text{true}} &= \tilde{U} c_{\text{true}} (1 + \eta_T) \end{aligned} \tag{25}$$

such that c_{com} is the commanded duty cycle at the given time step and c_{true} is the actual duty cycle. To extend this work to consider anomalies in the measurement matrix as well, we must use discriminative methodology to compare the expected forces and torques from the dynamical model $\hat{\tau}$ to the measured forces and torques. Since the control force is not measured directly, the estimated states from the previous time step using Eq. 3b is rearranged yielding

$$\epsilon_u = ||\tilde{U}^+(\hat{u} - u_m)|| = ||\tilde{U}^+ \left(\hat{u} - \left(\mathbb{I}\dot{\hat{\mathbb{V}}} - \text{ad}^*_{\hat{\mathbb{V}}} \mathbb{I}\hat{\mathbb{V}} - \hat{\tau} \right) \right) || \tag{26}$$

Zero Dynamics Attack

ZDAs are a sophisticated subclass of deception attacks targeting networked control systems (NCS). These attacks exploit the internal dynamics of an NCS, particularly focusing on non-minimum phase systems by leveraging unstable zeros. The stealthy nature of ZDAs makes them a significant threat, as they can cause substantial damage without immediate detection. They are designed to remain undetected by exploiting the system's internal dynamics, and the attack input grows geometrically without causing apparent changes in the system's output. These attacks primarily target non-minimum phase systems, which have unstable zeros that can be exploited. Consider a multiple-input multiple-output (MIMO) nonlinear system represented in the Byrnes-Isidori normal form:

$$\begin{aligned} \dot{\xi} &= A\xi + B\left[F(\xi,\eta) + G(\xi,\eta)u^a\right] \\ \dot{\eta} &= Z(\xi,\eta) \end{aligned} \tag{27}$$

where $\xi \in \mathbb{R}^r$ is the internal state vector, $\eta \in \mathbb{R}^l$ is the external state vector, $u^a = u + a$ is the injected input signal, $A \in \mathbb{R}^{r\times r}$ and $B \in \mathbb{R}^{r\times m}$ are constant matrices, and $F(\xi,\eta)$ and $G(\xi,\eta)$ are smooth nonlinear functions. The output equation is given by $y = C\xi$ where $y \in \mathbb{R}^m$ is the output vector and $C \in \mathbb{R}^{m\times r}$ is a constant matrix. The attacker designs the ZDA signal a using estimated states $\tilde{\xi}$ and $\hat{\eta}$. The attacker's estimate of the external state is obtained by:

$$\dot{\hat{\eta}} = Z(\tilde{\xi},\hat{\eta}) \tag{28}$$

where $\hat{\eta}$ is the estimated external state. The attack signal a is designed as:

$$a = G^{-1}(\tilde{\xi},\hat{\eta})\left[-F(\tilde{\xi},\hat{\eta}) + F(\xi,\eta) - G(\xi,\eta)u + G(\tilde{\xi},\hat{\eta})u\right]. \tag{29}$$

A similar method may be applied to the tangent space $\mathsf{TSE}(3)$ on which addition is defined. The attack signal a of arbitrary form is injected into the measurement model to cause instability in the target system. In this case, when obtaining measurements $y \in \mathbb{R}^q$ where q is the dimension of the measurements, it may simply take the form of

$$y_z = y + \eta + \tilde{\eta} + a \tag{30}$$

where η is the Gaussian measurement noise, $\tilde{\eta}$ is the non-Gaussian random walk, and a is the aforementioned attack.

FDIR

A fault detection vector is created to store binary knowledge of the status of various systems. These systems could be sensors, thrusters, thermal control systems, environmental control systems, or anything on the spacecraft. The format could be a vector, a matrix, or a series of matrices. For this work, a single vector containing the fault status of the sensors, which stores the knowledge of the sensors has a good measurement of angle, position, angular velocity, and velocity, concatenated to a vector of fault status of each thruster, of the same dimension of the number of sensors. In Algorithm 4 below, the method is explained. Specifically, when examining Flag(i), the 'i' is not the time step, but rather the index of the sensor. In other words, this method cycles through the sensors to identify any measurement that significantly exceeds the expected $3-\sigma$ bounds. Of course, this approach depends on having a good understanding of sensor performance before the spacecraft is in orbit.

NUMERICAL SIMULATION RESULTS

The $\mathsf{TSE}(3)$-UKF used in this work employs a dynamic model and a measurement model that estimates pose and augmented velocity. Thruster noise is also considered whenever the thrusters are fired, modeled as a noise parameter. The list of noise values used in this work is provided in Table 1. (m) denotes measurement noise while (p) denotes process noise.

Algorithm 3 Thruster Anomaly Logic

Define Initial Conditions $\hat{g}, \hat{\mathbb{V}}, g_d, \mathbb{V}_d$

Determine Desired Control
$u_d = f(\hat{g}, \hat{\mathbb{V}}, g_d, \mathbb{V}_d)$
Generate Estimated Thruster Firing History
$c_{\text{com}} = \min\left((\tilde{U}c - u_0)^T(\tilde{U}c - u_0) + kc^Tc\right)$
Estimate New Control Vector
$u_{\text{com}} = \tilde{U}c_{\text{com}}$
Check Accuracy
$\epsilon_u = ||\tilde{U}^+\left(u_{\text{com}} - \left(\mathbb{I}\dot{\hat{\mathbb{V}}} - \text{ad}^*_{\hat{\mathbb{V}}}\mathbb{I}\hat{\mathbb{V}} - \hat{\tau}\right)\right)||$
Check tolerance
if ϵ_u<||$\sqrt{N\xi^2}$ **then**
    Thruster anomaly detected
else
    $c_{\text{com}}(1) = 10$
    Thruster anomaly is detected, Abort Scenario Initiated
end if
if $c_{\text{com}}(1) == 10$ **then**
    $u_d = f(\hat{g}, \hat{\mathbb{V}}, g_{d,0}, \mathbb{V}_{d,0})$
    $c_{\text{com,new}} = \min\left((\tilde{U}c - u_0)^T(\tilde{U}c - u_0) + kc^Tc\right)$
    $\hat{u} = \tilde{U}c_{\text{com,new}}$
end if

Algorithm 4 Sensor Anomaly Logic

Define Predicted Pose and Velocity $\hat{g}_{k+1|k}, \hat{\mathbb{V}}_{k+1|k}, g_d, \mathbb{V}_d$

Take measurements
$y = h(g, \mathbb{V}, \dot{\mathbb{V}})$
Examine residuals between predicted state and measured state
$\tilde{y} = \hat{y} - \hat{x}_{k+1|k}$
Compare residuals to 3σ bounds & provide a tolerance TOL and apply flag
if $\tilde{y}(i) > \hat{x}_{k+1|k}TOL\eta(i)$ **then**
    Flag(i) = 1
end if
if Flag(i) $\neq 0$ **then**
    $\hat{y}(i) = \hat{x}_{k+1|k}(i) + \eta(i)$
end if

Nominal RPOD Scenario

Existing simulations are similar to previous work, but include the addition of measurement and process noise interpreted with a UKF on $\mathsf{SE}(3)$ 12, 13. Furthermore, thruster noise has likewise been added to the system, which introduces uncertainty in control output. The control of the spacecraft is accomplished using a tracking M-L controller described by [8, 28].

The approach profile for the RPOD simulation consists of a reference trajectory created using a 3-DoF simulation with impulse burns to create each phase. The simulation begins with an approach and hold at the rendezvous sphere, selected to be 10 km in radius, followed by an approach of 2-3 m/s lasting 170 seconds, a close approach of 1.5 m/s lasting 890 seconds, and a final approach of 50 cm/s lasting 850 seconds, resulting in a final 75 seconds of docking with a velocity of 15 cm/s. This 3-DoF simulation is then used as a reference from which to build the 6-DoF counterpart. To include attitude, the vehicle is presumed to be facing along the

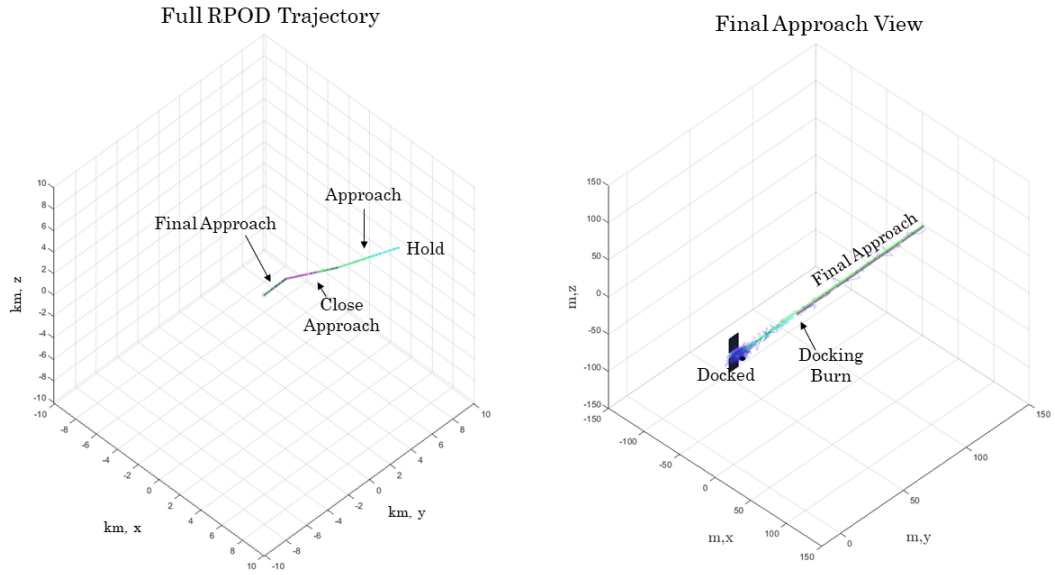

Figure 2: The approach of a 16-thruster tug-cargo configuration spacecraft along a reference 3-DoF RPOD trajectory with impulse burns. Purple and cyan indicate approach segments.

(relative) velocity direction during approach until entering the approach sphere, at which point the desired attitude of the spacecraft is one in which the first principal axis points along the Gateway y-axis (docking axis), and the z-axes are aligned.

A deadband is imposed for the control that may be tuned to obtain the desired load on the jet firing history. With no deadband, thrusters constantly fire and oscillate between controllable configurations. By allowing the vehicle to drift freely when tighter constraints on pose are not needed, a more fuel-optimal approach is obtained. The results of the current deadband constraint $D = 0.1\|r_{rel}\|$ where r_{rel} is the relative position is visible in the tracking, measurement, and estimation errors illustrated in Figure 3 and 4.

The control input from the M-L controller, the estimated distributed control, and the true distributed control are illustrated in Figure 5. The attitude component of the M-L control stands out as the saturation limits imposed by the given thruster configuration make obtaining those attitude rates while still maintaining a controlled translational approach unobtainable. That being said, the desired final pose is obtained by gradually pursuing the the optimal available jet firing histories over the simulation time. The oscillatory behavior of the

Table 1: Standard deviation for each source of noise.

Property	Parameter	Value	Units
Attitude (m)	η_R	$5\frac{\pi}{180}$	rad
Position (m)	η_r	10	m
Angular Velocity (m)	η_ω	$0.5\frac{\pi}{180}$	rad/s
Velocity (m)	η_v	0.1	m/s
Attitude (p)	η_R	$1.6\frac{\pi}{180}$	rad
Position (p)	η_r	3.3	m
Angular Velocity (p)	η_ω	$0.16\frac{\pi}{180}$	rad/s
Velocity (p)	η_v	0.033	m/s
Thrust	η_T	5	%

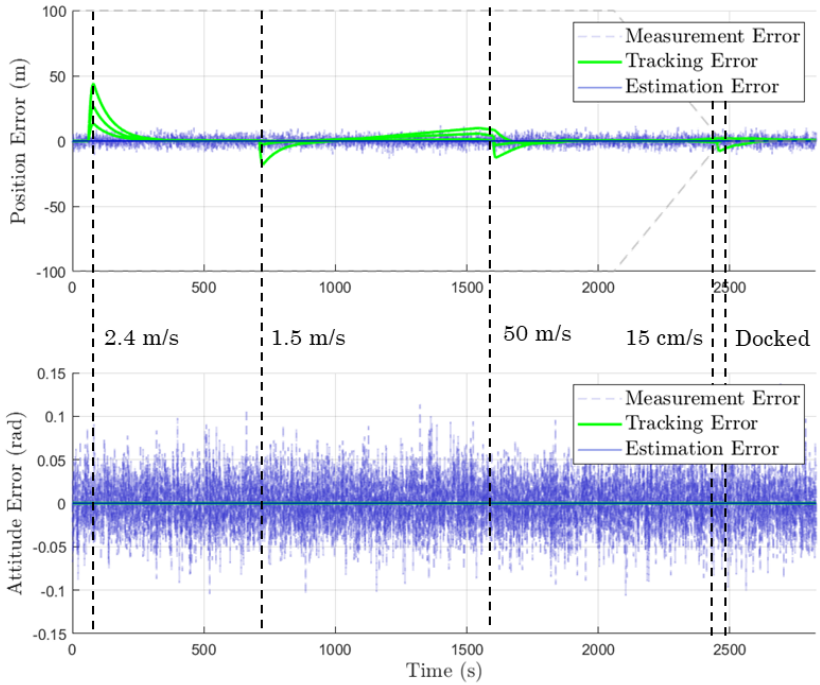

Figure 3: The position and attitude tracking error, where the dotted line is the deadband region for deadband control scalar D.

true and estimated thruster input is a again result of the given thruster configuration in which all the thrusters are mounted at one end of the vehicle and the center of mass of the vehicle lies outside of the geometric center of the thrusters.

The jet firing history resulting from the control input in Figure 5 is illustrated in Figure 6, where the y-axis is scaled as simply on/off (alternatively, it can be thought of as a normalized value between d_c and 0). These values are not distributed and constrained, but not discretized. Since true duty cycles are discretized on some number of intervals between 0 and d_c, it is desired to implement the discretization of normalized outputs in future work. However, efforts to impose this discretization on the current methodology have resulted in unstable behavior.

The propellant use during the nominal RPOD simulation, both in terms of delta-V and propellant mass, is illustrated in Figure 7. The values of around 100 kg and 20 to 25 m/s align with expected values for a vehicle of this size and for these approach speeds. Other burns required for rendezvous between spacecraft such as the insertion burn into the orbit are not considered.

RPOD Scenario with Zero Dynamics Attack without FDIR

In the second scenario, a ZDA is simulated on the accelerometers for the spacecraft configuration with no mitigation strategy implemented. The purpose of this simulation is to illustrate the consequences of ineffective or insufficient FDIR. This attack could be considered an attack on the accelerometers, which integrate to measure velocity in many spacecraft, an attack from ground-based RADAR, or from relative

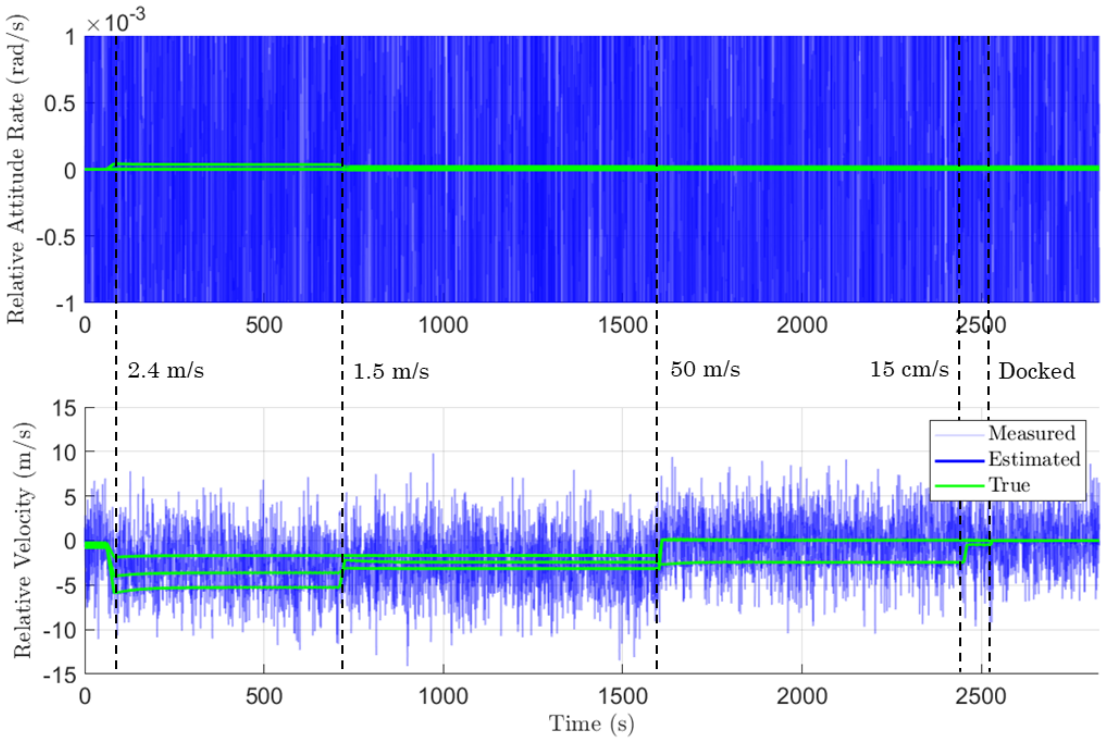

Figure 4: The angular and translational velocity during the nominal RPOD trajectory.

motion LIDAR systems. The specific attack generated is of the form

$$y_z = \begin{bmatrix} \sin\left(\frac{2\pi t(k)}{12}\right) \\ \cos\left(\frac{2\pi t(k)}{8}\right) \\ \sin\left(\frac{-2\pi t(k)}{11}\right) \end{bmatrix} e^{\frac{t(k)-2000}{80}} \tag{31}$$

where y_z is the ZDA attack. This is applied to the noisy measurement vector after it passes through the measurement function to simulate an attack that augments. Functionally, y_z becomes large enough that it is indistinguishable from a signal that replaces existing measurements. Figure 8 shows the ZDA scenario playing out from rendezvous down to 1 km. After an attack is initiated, the coupled formulation that is usually the strength of this method becomes a weakness as the largely oscillating measurements induce instability across the spacecraft system. The spacecraft begins to spin, all the while attempting to track the reference trajectory and using up valuable propellant. Comparing Figures 9 and 10 with their counterparts from the nominal scenario, it is clear that the first phases of the approach are identical. However, the undesired motion imposed by the ZDA overwhelms the otherwise nominal approach. The tracking error grows over time due to the vehicle departing the original desired trajectory.

RPOD Scenario with Zero Dynamics Attack with active FDIR

In the third scenario, a ZDA is simulated on the velocity measurement for the spacecraft configuration including the FDIR approach described in Section . This method of applied discriminative unscented Kalman filtering applies an additional step between the prediction and update steps that examines the residuals between the prediction and the measurement, and another step after the update that examines the residuals between the measurement and the update. In the first part, it seeks to identify anomalies in the sensor model.

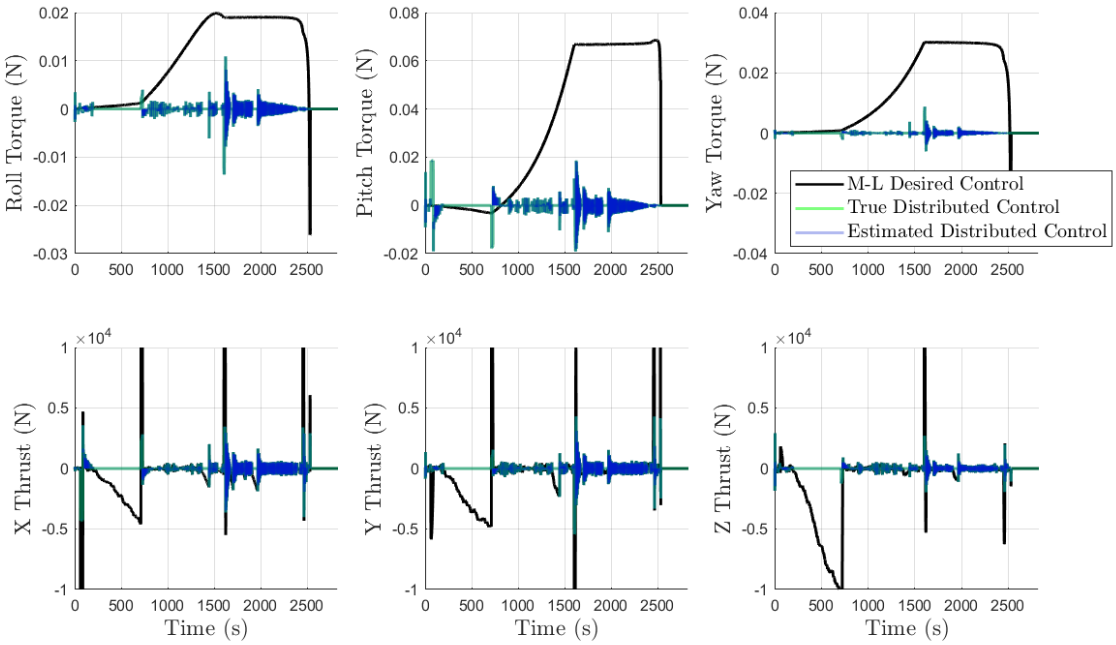

Figure 5: The desired (Morse-Lyapunov) control about the body frame compared to the estimated distributed, and true distributed control input.

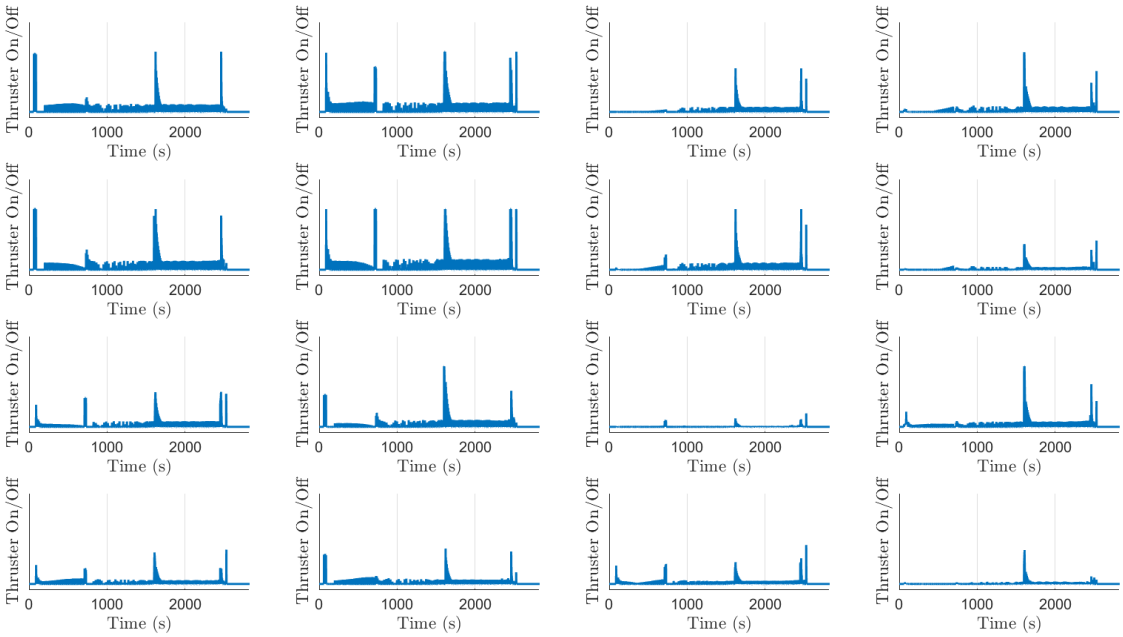

Figure 6: The distributed jet firing history during a nominal RPOD trajectory.

In the second part, it seeks to identify anomalies in the dynamical model. Since the focus of this work is to identify measurement errors or attacks from bad actors, only the first part was actively used, though the

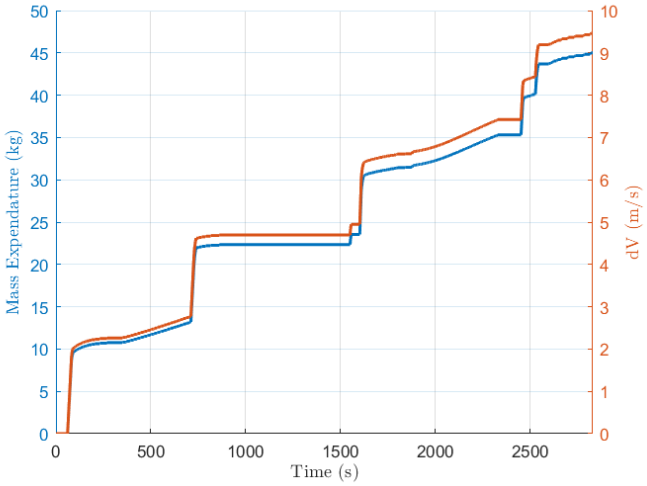

Figure 7: The propellant used for the full, nominal RPOD trajectory.

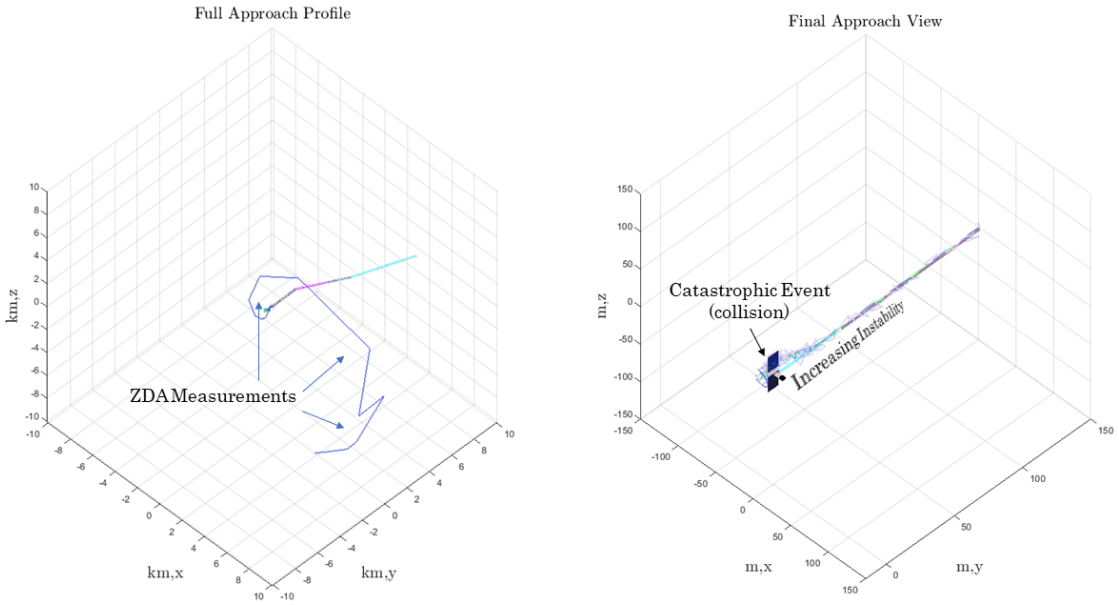

Figure 8: The approach of the same spacecraft during the ZDA scenario.

second portion remained active in the background.

The results of this simulation are shown in Figure 11, which shows that despite the existence of exponentially growing measurements, the spacecraft was able to reject these signals and successfully dock with the target, in this case, Gateway. The position and attitude error shown in Figure 12 and the velocity errors shown in 13 are comparable to the nominal except for the measurement error, which the controller ignores. The process by which the FDIR Flag tracks anomalies is illustrated in Figure 14, where each of the sensors is either flagged as within reasonable bounds (given a 0) or as potentially in a failure mode (1). The threshold to which this is tuned. Tuning may be performed using a constant, such as a scalar multiplier of the standard deviation,

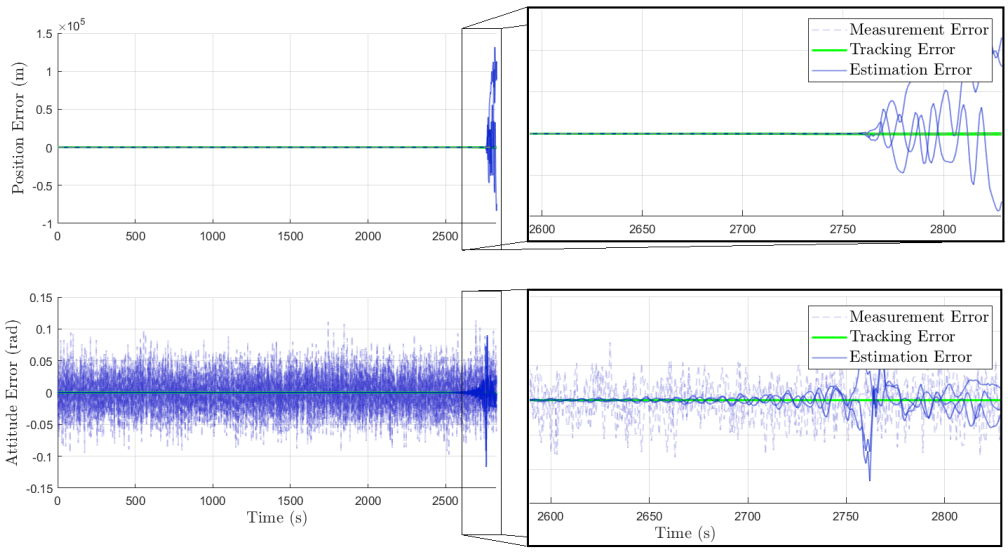

Figure 9: The relative attitude and position during the RPOD trajectory during ZDA.

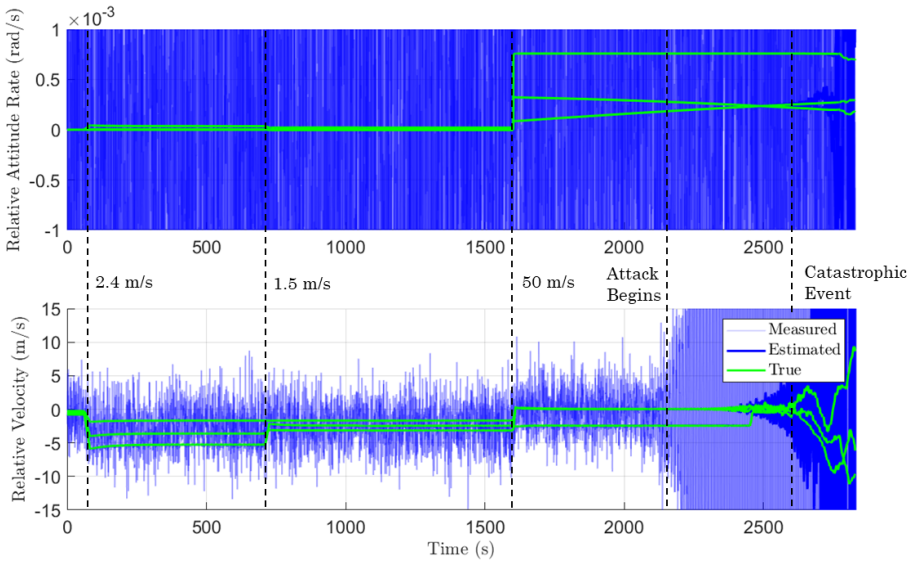

Figure 10: The relative velocity and angular velocity during the ZDA scenario.

or by some dynamic property, such as a combination of relative distance or if that sensor has a history of bad performance. Some approaches may include a variety of factors such as remaining propellant, collision risk, and other factors that require more complex decision-making. For this study, a simple multiplier of 1.3 times the $3-\sigma$ distribution of that sensor was selected.

CONCLUSIONS

In this work, a Morse-Lyapunov (M-L) tracking controller on $\mathsf{TSE}(3)$ first introduced in [8, 28] has been implemented for the problem of a rendezvous, proximity operations, and docking (RPOD) problem between a logistics module and NASA's Gateway in the presence of thruster, measurement, and dynamical model uncertainties. This included autonomous jet firing history generation, even when the thruster configurations

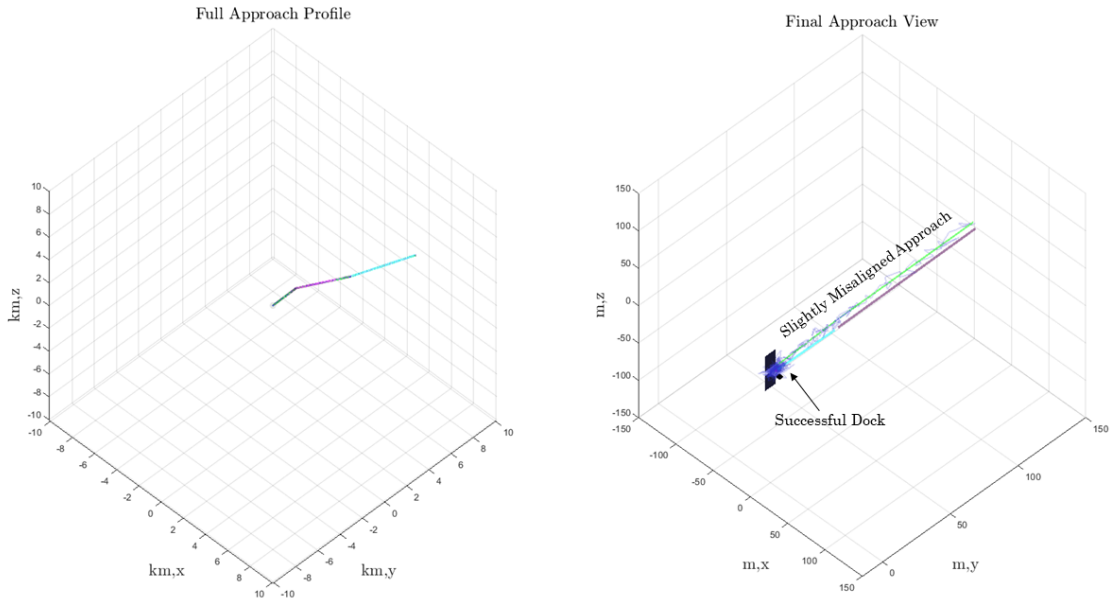

Figure 11: The spacecraft trajectory during the ZDA scenario with FDIR.

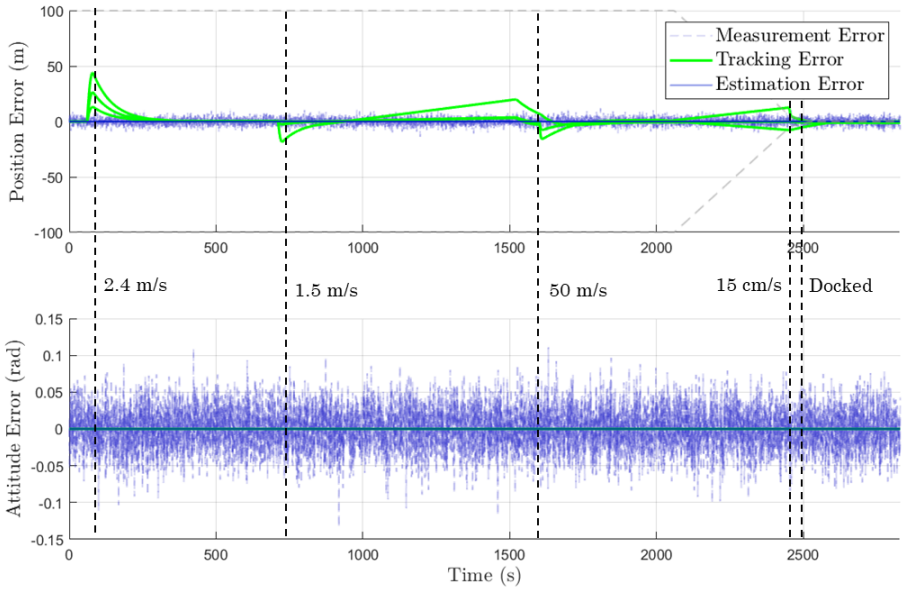

Figure 12: The relative position and attitude during the ZDA scenario with FDIR.

are not oriented about the center of mass of the vehicle based on the findings of [1, 2] and extended to include autonomous abort scenarios in [3].The uncertainties have been filtered using a discriminative unscented Kalman filter (DUKF) on $\mathsf{SE}(3)$ first proposed by [12, 13] and extended in this and upcoming work to include discriminative methodology. Additionally, a rudimentary fault isolation, detection, and remediation (FDIR) approach has been implemented using an analysis of measurement residuals. The simulation was able to guide the spacecraft to a safe docking state with a velocity of 15 cm/s at the axial docking port. The ben-

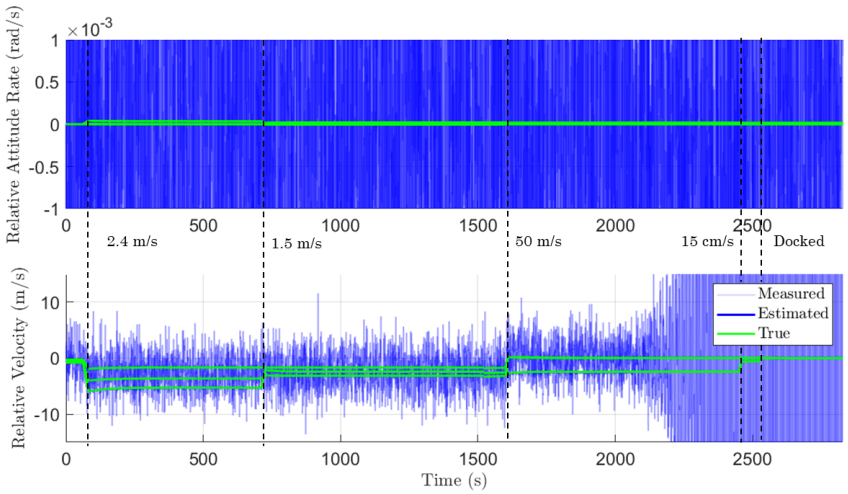

Figure 13: The relative velocity and angular velocity during the ZDA scenario with FDIR.

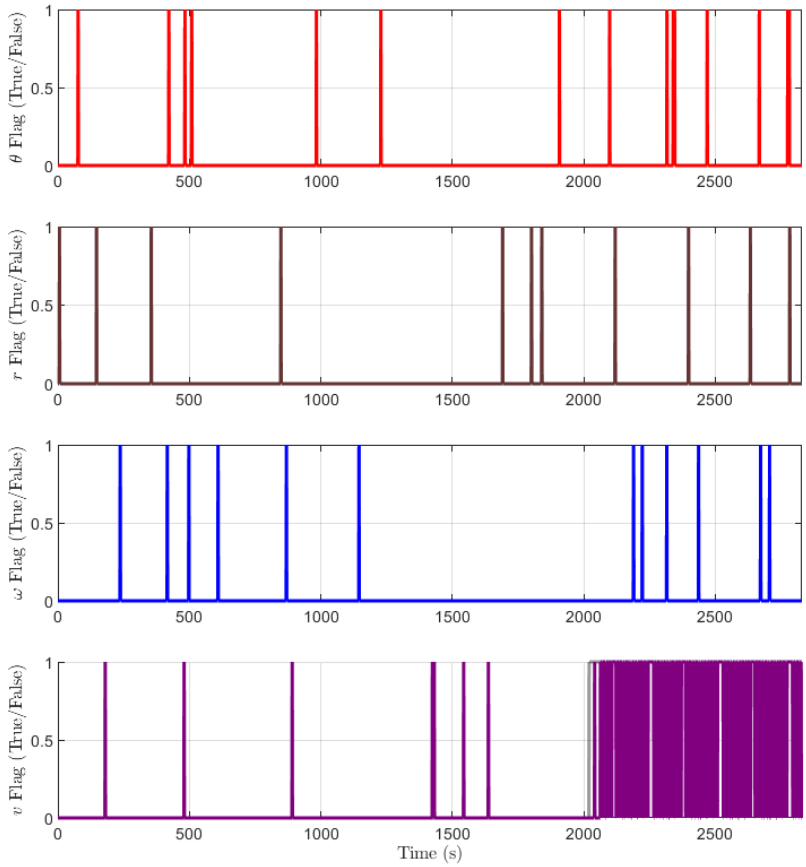

Figure 14: The algorithm that detects anomalies among each of the sensors. There are some erroneous detections, but the attack is visible along with the FDIR.

efits of this work are an increased robustness in the current state-of-the-art RPOD operations, which have historically been well within the regime of human intervention and observation. Furthermore, the ability of the vehicle to overcome faults or attacks on its sensors has the potential to reduce the risk of implementing autonomous control systems near high-value assets such as the Gateway.

REFERENCES

[1] M. M. Wittal and M. Nazari, "Autonomous Constrained Control for Arbitrary Thruster Configurations in SE(3)," in *2023 AAS Astrodynamcis Specialist Conference*, no. AAS 23-307, 2023.

[2] M. M. Wittal, B. W. Asher, B. McCann, and M. Nazari, "Autonomous Control for Arbitrary Thruster Configurations and Mass Properties in Special Euclidean Group SE(3)," in *74th International Astronautical Congress*, no. IAC-23,C1,3,2,x78246, 2023.

[3] M. M. Wittal, B. W. Asher, and M. Nazari, "Autonomous RPOD for Arbitrarily Configured Spacecraft with Anomaly Detection," in *2024 AAS Astrodynamcis Specialist Conference*, no. AAS 24-439, 2024.

[4] T. Hashimoto, T. Kubota, J. Kawaguchi, M. Uo, K. Baba, and T. Yamashita, "Autonomous descent and touch-down via optical sensors," *Advances in the Astronautical Sciences*, vol. 108 I, pp. 469–480, 2001. Proceedings of the AAS/AIAA Space Flight Mechanics Meeting ; Conference date: 11-02-2001 Through 15-02-2001.

[5] J. Pei, M. N. Choueiri, R. Elandt, M. A. Peck, and P. Finch, "An Innovative Control Allocation Scheme to Address Reaction Thruster Interactions on a 3U CubeSat," in *AIAA Guidance, Navigation, and Control Conference*, 2018.

[6] T. Lee, M. Leok, and N. H. McClamroch, "Lie group variational integrators for the full body problem," *Computational Methods in Applied Mechanics and Engineering*, vol. 196, no. 8, 2005, doi: 10.1016/j.cma.2007.01.017.

[7] D. Seo and M. Nazari, "Rigid body adaptive stabilization on the tangent bundle of the Lie groups," *AIAA Scitech 2019 Forum*, 2019, doi: 10.2514/6.2019-0653.

[8] G. Mangiacapra, M. Wittal, E. Capello, and M. Nazari, "Unscented Kalman filter and control on TSE(3) with application to spacecraft dynamics," *Nonlinear Dynamics*, vol. 108, pp. 2127–2146, 2022.

[9] B. McCann, H. V. Gunter, M. Nazari, and T. Henderson, "Generalized Formalism for Variable-Mass-Property Spacecraft Dynamics Utilizing Rigid Body Motion on Lie Groups," in *AIAA/AAS Astrodynamics Specialist Conference*, vol. 22-790, 2022.

[10] M. M. Wittal, B. McCann, J. Benavides, and M. Nazari, "Ambiguity Remediation in Launch Vehicles with Parameter Uncertainties: A Comparison between Special Euclidean Group and Dual Quaternion," in *AIAA/AAS Astrodynamics Specialist Conference*, vol. 22-623, 2022.

[11] B. S. McCann, M. Fagetti, M. Nazari, M. M. Wittal, and J. D. Smith, "Mass Property Estimation on $\mathsf{TSE}(3)$ via Unscented Kalman Filter Using RCS Thrusters," *Acta Astronautica*, 2024.

[12] M. Brossard, S. Bonnabel, and J. Condomines, "Unscented Kalman filtering on Lie groups," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, doi: 10.1109/IROS.2017.8206066.

[13] M. Brossard, S. Bonnabel, and A. Barrau, "Unscented Kalman filter on Lie groups for visual inertial odometry," *"IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, doi: 10.1109/IROS.2018.8593627.

[14] B. S. McCann, M. M. Wittal, and M. Nazari, "Relative Spacecraft Position and Attitude in the Circular Restricted Three-Body Problem: TSE(3) vs. Dual Quaternions," in *AIAA SCITECH 2023 Forum*, 2023.

[15] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "A secure control framework for resource-limited adversaries," *Automatica*, vol. 51, pp. 135–148, 2015.

[16] F. Pasqualetti, F. Dorfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2715–2729, 2013.

[17] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," *ACM Transactions on Information and System Security*, vol. 14, no. 1, pp. 1–33, 2011.

[18] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure estimation and control under adversarial attacks," *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1454–1467, 2014.

[19] Y. Mo and B. Sinopoli, "Secure control against replay attacks," in *47th Annual Allerton Conference on Communication, Control, and Computing*, pp. 911–918, 2009.

[20] Y. Yuan, Q. Zhu, F. Sun, Q. Wang, and T. Basar, "Resilient control of cyber-physical systems against denial-of-service attacks," *International Symposium on Resilient Control Systems*, vol. 6, pp. 54–58, 2013.

[21] H. Zhang, P. Cheng, L. Shi, and J. Chen, "Optimal denial-of-service attack scheduling with energy constraint," *IEEE Transactions on Automatic Control*, vol. 60, no. 11, pp. 3023–3028, 2015.

[22] G. K. Befekadu, V. Gupta, and P. J. Antsaklis, "Risk-sensitive control under markov-modulated denial-of-service (dos) attack strategies," *IEEE Transactions on Automatic Control*, vol. 60, no. 12, pp. 3299–3304, 2015.

[23] C. De Persis and P. Tesi, "Input-to-state stabilizing control under denial-of-service," *IEEE Transactions on Automatic Control*, vol. 60, no. 11, pp. 2930–2944, 2015.

[24] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Security Privacy*, vol. 9, no. 3, pp. 49–51, 2011.

[25] J. P. Farwell and R. Rohozinski, "Stuxnet and the future of cyber war," *Survival*, vol. 53, no. 1, pp. 23–40, 2011.

[26] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "Revealing stealthy attacks in control systems," in *50th Annual Allerton Conference on Communication, Control, and Computing*, pp. 1806–1813, 2012.

[27] M. M. Wittal, B. S. McCann, and M. Nazari, "Discriminative Kalman Filtering in Special Euclidean Group SE(3) for Modeling Unknown Spacecraft Environments," in *AIAA SCITECH 2023 Forum*, 2023.

[28] M. Nazari, E. A. Butcher, and A. K. Sanyal, "Spacecraft attitude fractional feedback control using rotation matrices and exponential coordinates," *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 10, pp. 2185–2198, 2018.