Bayesian Active Sensing and Planning Applied to Attitude Dynamics and Spacecraft Control

By: Arnauld Martinez
Mentored by Jimmy Ragan, Hannah Grauer, and Soon-Jo Chung

Modern spacecraft must be equipped with a high degree of fault tolerance to ensure safe and successful system operation during critical moments in the spacecraft's mission. To combat total system failure, engineers implement redundancy to allow multiple failures to occur without compromising vehicle performance or objectives. However, the decision to use redundancy is not without penalty. If a redundant system fails, the source of the failure reduces to a family of potential problems that could require significant time to analyze and repair. The Autonomous Robotics and Control Lab at Caltech has combated this issue by developing FEAST (Fault Estimation via Active Sensing Tree Search), a novel algorithm that can identify failing sensors and actuators in a redundant spacecraft system. The accuracy of FEAST was verified on planar spacecraft models with three degrees of freedom. In this paper, we lay the groundwork to extend FEAST to consider the spacecraft's attitude when isolating and recovering from faults. We model the spacecraft using quaternions and vary its orientation using reaction wheels. Additionally, we build a more realistic sensing model to simulate and recover from known failure modes in monocular cameras. By designing a procedure to identify faulty sensors and actuators, we aim to increase spacecraft reliability and safety.

I. Introduction

For centuries, mankind has been captivated by the limitless expanse and beauty of the cosmos. However, until the twentieth century, humans were limited to marveling at the universe from Earth. Recent developments in flight hardware, propulsion, and autonomous control have enabled scientists to overcome the shackling pull of Earth's gravity and pioneer celestial travel. However, pushing the boundaries of scientific knowledge throughout the cosmos requires unparalleled spacecraft safety and reliability. At the heart of this challenge, scientists study fault tolerance to guarantee successful spacecraft operation despite the harsh conditions of space. The ability to isolate and recover from faults using tractable algorithms is critical in aerospace development because it maximizes mission success, protects on-board financial investments, and maintains data integrity. [1].

Between 2000-2016, the world launched an average of 65 small satellites per year, with 41.3% of them experiencing failure, of which 24.2% were total system failures [2]. These remarkably high figures suggest that in moments where equipment readings cannot be verified by humans, the effect of a single equipment failure can be magnified into vehicle/system damage. To make autonomous systems robust to these failures, engineers implement redundancy to allow multiple failures to occur without compromising system performance or objectives. Redundant systems can maintain and repair vehicle damage with no system downtime, ultimately prolonging vehicle operation time. By designing redundancy into critical operations, scientists improve system function and increase mission reliability [3].

Despite increasing mission reliability, redundancy cannot always be incorporated into a spacecraft's design. Redundancy increases the weight, complexity, and cost of the spacecraft and requires further time to test and debug. Notably, if a redundant system fails, the source of the failure is not easily identifiable. Engineers combat this issue by implementing fully redundant systems that switch when a component fails, but this doubles the weight and complexity of onboard instruments [3]. The need for inexpensive, lightweight, and accurate fault detection methods is growing, especially as organizations begin sending more complex and valuable instruments to space.

The Autonomous Robotics and Control Lab (ARCL) at Caltech attempts to solve this problem by developing novel Fault Detection, Isolation, and Recovery (FDIR) algorithms that focus on rapidly self-diagnosing faulty sensors and actuators on spacecraft. To achieve this, researchers in the lab have formalized the problem of system failure in a redundant setting as a Partially Observable Markov Decision Process (POMDP). The group developed a novel algorithm to approximate a solution to the POMDP using Partially Observable Monte Carlo Planning with Marginalized Filtering (POMCPMF). When optimized, the Fault Estimation via Active Sensing Tree Search (FEAST) framework improves on existing state-of-the-art fault detection methods by providing exact and efficient Bayesian updates [4]. The accuracy of the model was validated using mathematical analysis and onboard a 3-degree-of-freedom planar satellite (Figure 1).

The purpose of my project is to begin extending the FEAST algorithm to identify faulty components affecting the dynamics of a spacecraft's attitude. Namely, the algorithm will create a belief state spanning relevant motors and actuators (ranging from reaction wheels to thrusters) and classify each as faulty or nominal. This belief state will be updated with real-time information the spacecraft gathers as it takes various actions. Additionally, we seek to raise the level of realism present in the sensing matrix of the FEAST framework. We aim to create a model that can simulate and recover from common types of failure modes within monocular cameras.

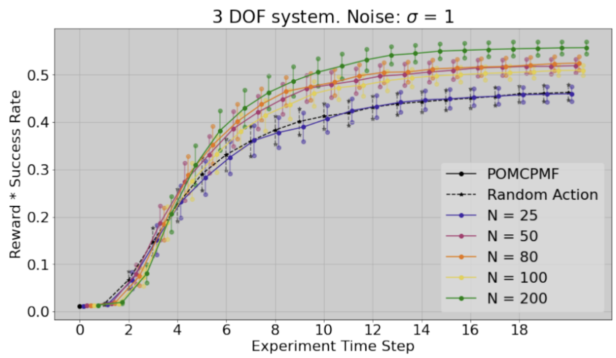

Fig. 1 Performance metric vs. experiment time for the proposed POMCPMF algorithm. As the number of simulations in the tree increases, the accuracy of the model significantly outperforms random actions. Each data point is averaged over 300 initial conditions with high noise. Reproduced from [4]

II. Goals

The current and most tested implementation of FEAST enables a spacecraft with three degrees of freedom to identify and recover from failed thruster outputs and sensor readings. However, this active-planning approach only considers translational and rotational movement in the XY plane. In practice, modern spacecraft must keep track of their position in 3D space with respect to multiple reference frames. As a result, we lay the framework that extends FEAST to consider the spacecraft's yaw, pitch, and roll angles when identifying and recovering from discrete failure modes. In addition to thrusters, reaction wheels, which transfer stored momentum to change the spacecraft's attitude, become a source of error that FEAST must identify and recover from.

Additionally, we aim to raise the level of realism for the sensing matrix used to update FEAST's belief state. The current implementation of FEAST multiplies sensor readings by 0 or 1, corresponding to a faulty or nominal belief for a given sensor. However, failure modes in existing spacecraft are rarely discrete. For example, hot pixels plague small portions of data with faulty but consistent information. Failure modes such as hot pixels are not well represented by multiplying input data by 0 or 1. As a result, we seek to create an algorithm that can generate realistic sensor data, identify non-discrete failure modes, and recover from these faults.

III. Methods

The first focus of the project is to lay the groundwork that extends FEAST to consider the spacecraft's attitude when isolating failure modes. We began by creating a simulation that determines how the torque generated by reaction wheels affects the orientation of a CubeSAT satellite. To create the attitude simulation, we began by deriving the relevant attitude kinematic equations relating torque to the satellite's orientation in terms of Euler angles and a quaternion 4-tuple. Specifically, we expressed the time rate of change of the rotation matrix of the spacecraft, which is given by Eq. (1) and (2), as a function of its angular velocity. Here, ${}^{\mathcal{I}}\mathcal{R}^{\mathcal{B}}$ represents the rotation matrix from the body frame, $\mathcal{B}$, to the inertial frame, $\mathcal{I}$. Additionally, we derived the time rate of change of Euler angles and quaternions as a function of the spacecraft's yaw, pitch, and roll angular velocities, given by Eq. (3) and (4). Note, q_0 through q_3 are elements of the quaternion 4-tuple representing the spacecraft's orientation, ω_1 through ω_3 are components of the spacecraft's angular velocity, and θ_1 through θ_3 is an XYZ Euler Angle sequence.

$$\frac{d}{dt} {}^{\mathcal{I}}\mathcal{R}^{\mathcal{B}} = {}^{\mathcal{I}}\mathcal{R}^{\mathcal{B}} S(\omega_{\mathcal{B}}) \tag{1}$$

$$S(\omega) = \begin{bmatrix} 0 & -\omega_1 & \omega_2 \\ \omega_1 & 0 & -\omega_3 \\ -\omega_2 & \omega_3 & 0 \end{bmatrix} \tag{2}$$

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} = \begin{bmatrix} \frac{\cos\theta_3}{\cos\theta_2} & -\frac{\sin\theta_3}{\cos\theta_2} & 0 \\ \sin\theta_3 & \cos\theta_3 & 0 \\ -\tan\theta_2\cos\theta_3 & \tan\theta_2\sin\theta_3 & 1 \end{bmatrix} \omega_{\mathcal{B}} \tag{3}$$

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 & q_0 \\ q_0 & -q_3 & q_2 & q_1 \\ q_3 & q_0 & -q_1 & q_2 \\ -q_2 & q_1 & q_0 & q_3 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ 0 \end{bmatrix} \tag{4}$$

Using these kinematic equations, we developed a simulation that can inductively propagate the given system forward one time step using numerical simulation instead of analytical analysis, which provides much faster and equally accurate results [5]. The spacecraft is modeled as a cube to simulate the moment of inertia found in a CubeSAT satellite. Additionally, off-axis reaction wheels were placed on the spacecraft's perimeter to imitate the hardware found on the M-STAR satellite prototype in Caltech's spacecraft simulation room (Figure 2). Future hardware testing of the fault detection, isolation, and recovery algorithm will be performed on the M-STAR prototype.

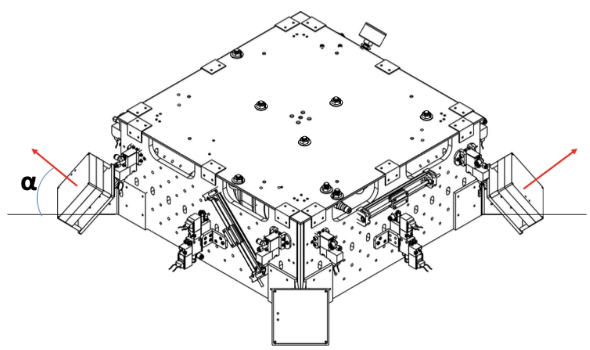

Fig. 2 Orientation of Reaction Wheels on the M-STAR. Reaction Wheels are found on all four corners of the spacecraft and offset from the XY plane by an angle α. Adapted from [6]

Lastly, we integrated our simulation into the existing FEAST framework. We created a spacecraft model with three degrees of freedom to consider only the attitude dynamics of a spacecraft. In an attitude specific setting, FEAST begins by initializing a trial based on several user-defined parameters. Users exercise control over all parameters of the simulation, ranging from the system's failure state, reward function, and belief solvers (in our case we utilize POMCPMF). For the first iteration, FEAST begins by verifying the system is solvable. If the system is solvable, it generates a random belief state across all actuators and sensors and takes an action to verify the accuracy of the guess. The result is cached and stored in an output log. For every successive iteration, FEAST reads the most recent data stored in the output log and generates a physical and failure state. Using the observation generated by the previous action, the system can inductively generate the physical state, failure state, and observation at the next time step. This three-tuple is used to generate the system's next belief state and reward. Following the trial, the data is saved to the output log, and the simulation ends if the generated reward is greater than a user-defined confidence threshold.

The second focus of this project revolves around creating a framework that can generate realistic sensor data, identify discrete failure modes, and recover from these faults. In addition to the physical behavior of the reaction wheels, the output of the generated sensing matrix is a potential source of error that should be identified by the FEAST algorithm.

Since the reaction wheels in the attitude dynamics simulation were modeled after the M-STAR satellite prototype, our algorithm will attempt to mimic and recover from hot pixels present on the satellite's infrared cameras. To create a model that can identify hot pixels in a given infrared image, we must first have a large dataset of faulty and nominal infrared satellite images. Since no such dataset currently exists, but large datasets of RGB satellite images are easily accessible online, we created a framework that can convert RGB satellite images to IR images.

This framework is an extension of cycleGAN, a neural network used to transform images between two different domains. CycleGAN is well suited for our purposes because the testing images do not have to be paired. In practice, it is extremely difficult to take paired infrared and RGB images of satellites because the camera must not move with respect to the spacecraft or background objects. To transform images between IR and RGB domains, we create two generators G and F: generator G transforms images from RGB to IR, and generator F transforms images from IR to RGB. With no other restraints, the generators do not accurately transform images to their respective domain (Figure 3). To ensure accurate and consistent mapping, we define and minimize the generator's cycle consistency loss and identity loss. Cycle consistency loss measures the difference between an image and the same image passed between both generators (Eq 5). Identity loss measures the difference between an image and the image passed through the incorrect generator (Eq 6). Here, X represents in arbitrary RGB image and Y represents an arbitrary IR image. A visual representation of cycle consistency loss is given in Figure 4.

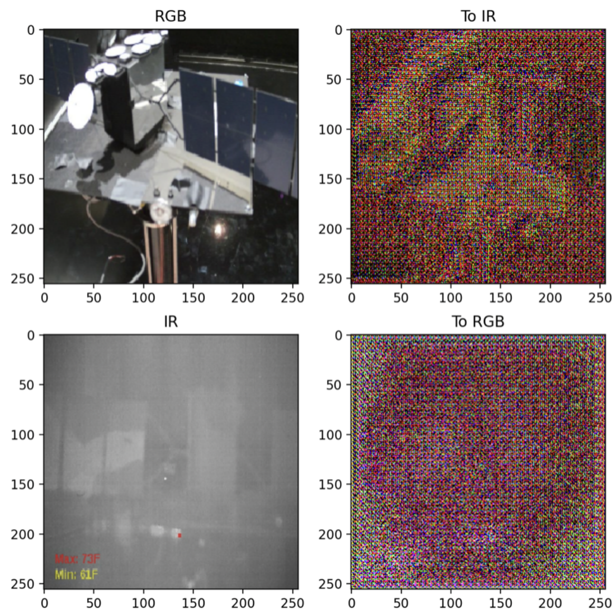

Fig. 3 Generator G (top) and generator F (bottom) mapping RGB and IR images to the corresponding domain.

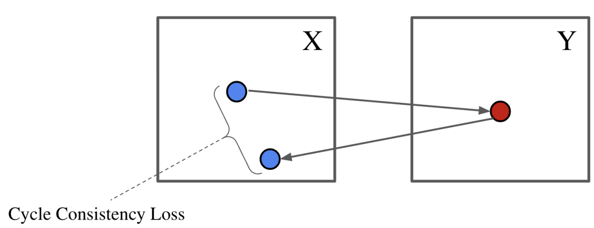

Fig. 4 Visual representation of generators mapping between two different domains and measuring cycle consistency loss.

$$\text{Cycle Consistency Loss} := |F(G(Y)) - X| + |G(F(X)) - Y| \tag{5}$$

$$\text{Identity Loss} := |G(Y) - Y| + |F(X) - X| \tag{6}$$

After implementing and minimizing both loss functions, the resulting model can accurately transform RGB images of satellites to infrared images. Using Figure 5, we verify the model is not merely converting to grayscale because the hottest part of the satellite (the radio dish) is the brightest region of the generated IR image and the coldest part of the satellite (the solar panels) are the darkest region of the generated IR image.

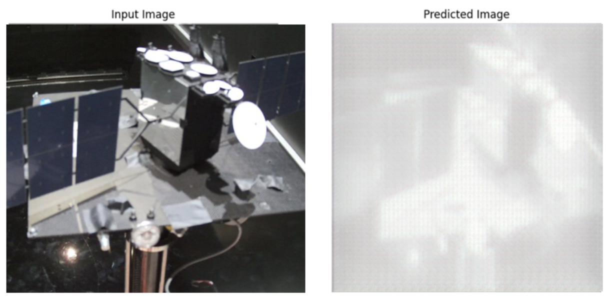

Fig. 5 RGB satellite image (left) and the predicted infrared image (right) after passed through generator G.

Next, we artificially overlayed hot pixels on the generated infrared images. We pseudo-randomly choose ten sites on the generated image and maximize the neighboring pixels' intensities. The resulting images are realistic representations of data given by faulty infrared cameras on satellites. To identify these hot pixels, we utilized openCV to apply a median blur to the faulty image, then we determined the difference in pixel intensity between the original and blurred image. We classified pixels whose intensity difference was greater than 0.25 (from a normalized 0 to 1 scale) as hot pixels. Finally, to remove hot pixels, we replaced identified hot pixels in the original image with the corresponding pixel in the median blur counterpart. This algorithm ensures critical infrared data is not lost or compromised. The generation and recovery process for hot pixels in infrared satellite images is shown in Figure 6.

IV. Conclusion

In this paper, we set the groundwork to expand the fault detection algorithm FEAST to consider the spacecraft's attitude when isolating and recovering from failure modes in a redundant system. Additionally, we created a framework that can generate realistic infrared images of satellites subject to hot pixels. Finally, we developed a method that can identify and remove hot pixels from satellite images.

In the future, we hope to build off the groundwork set by this paper's work to allow FEAST to consider the spacecraft's orientation when detecting faults. Namely, we aim to develop an algorithm that will generate a belief, between [0, 1], that an observed infrared image contains hot pixels. From this belief, the algorithm will determine the next best action the spacecraft should take to verify its belief. For example, the algorithm may choose to take a photo of the same object, using the same camera, but from a different angle. The resulting 3-tuple of an observation, belief, and action is a partially observable markov decision problem which can be solved using Monte-Carlo Tree Search with Marginalized Filtering to update the spacecraft's overall belief state. Additionally, we hope to experiment with other types of failure modes in infrared cameras, including solar glare, and investigate failure modes across other sensors, such as gyroscopes or barometers.

V. Acknowledgements

I want to sincerely thank my mentor, professor Soon-Jo Chung, for giving me the opportunity to conduct amazing research in the ARCL lab. I also want to thank my graduate student mentors, Jimmy Ragan and Hannah Grauer, for their constant support and guidance throughout the research project. Finally, I am extremely thankful to Dr. Lawrence Taylor, The Aerospace Corporation, and JPL for providing me with the technical funding and resources to explore attitude dynamics and computer vision.

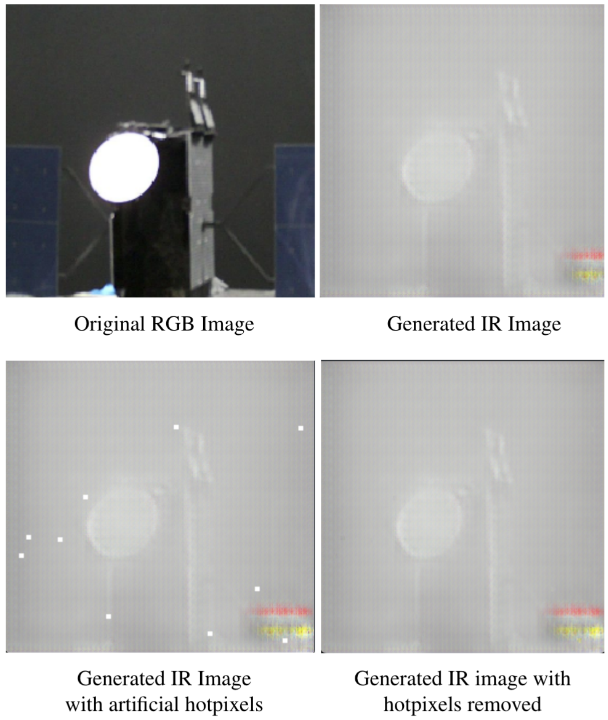

Fig. 6 Generation and recovery process for hot pixels on satellite images.

References

[1] NASA, "Redundancy in Critical Mechanical Systems," `https://llis.nasa.gov/lesson/659`, 1999.

[2] Jacklin, S. A., "Small Satellite Mission Failure Rates," 2018.

[3] NASA, "Active Redundancy," `https://llis.nasa.gov/lesson/697`, 1999.

[4] Ragan, J., Riviere, B., and Chung, S.-J., "Bayesian Active Sensing for Fault Estimation with Belief Space Tree Search," 2023.

[5] Garcia, M., "Attitude From Angular Rate," 2019.

[6] Yashwanth Kumar Nakka, E. S. L., Rebecca C. Foust, "A Six Degree of Freedom Spacecraft Dynamics Simulator For Formation Control Research," 2018.