[SSC21-P1-37]

Automated Fault-Detection for Small Satellite Pointing Control Systems using One-Sided Learning

Yujia Huang
Graduate Student, Department of Mechanical Engineering
University of Manitoba, Winnipeg, Manitoba, R3T 2N2, Canada
huangy25@myumanitoba.ca

Philip A. Ferguson
NSERC / Magellan Aerospace Industrial Research Chair in Satellite Engineering
University of Manitoba, Winnipeg, Manitoba, R3T 2N2, Canada
Philip.Ferguson@umanitoba.ca

ABSTRACT

In this paper, we propose a ground-based automated novelty detection system for a small satellite attitude dynamics control system using a one-sided learning algorithm: One-Class Support Vector Machine (OC-SVM) method. This fault-detection system was designed to only learn from nominal behavior of the satellite during the commissioning phase and to identify and detect anomalies when there was a subtle behavioral failure in the attitude control system. The detection system was trained by only observing the nominal attitude dynamics behavior of a small satellite for a period of time. Training data was obtained from reaction wheel outputs in a healthy attitude control system, and reaction wheel currents and angular velocities were selected as training features. A one-class classifier was built from a hyperplane decision function during training. An adaptive Sequential Minimal Optimization (SMO) method was utilized to solve the quadratic problem in the application of OC-SVM algorithm to provide an optimal solution for the hyperplane decision function. Two tests were performed on the system to validate its feasibility and detection accuracy. Untrained reaction wheel bearing failures were added into the attitude control system validation tests to examine whether the fault-detection system was capable of detecting and diagnosing the reaction wheel failures. Training and testing performance for the fault-detection system are presented with discussion.

INTRODUCTION

As the size and prevalence of small satellite constellations grow, so does the interest in Prognostic and Health Management (PHM). In keeping with the small-satellite philosophy to maintain low design, manufacturing and operating costs, the small-satellite community is interested in efficient ground operations and fault management that does not require excessive labor from trained space systems experts [1]. The expanding scale of small-satellite constellations has posed a significant challenge for ground operations: how to find a sustainable way to monitor and manage a large amount of satellites efficiently with minimal cost?

While many researchers have relied upon machine learning to detect faults ([2-8]), this approach suffered from the limitation that it could only detect failures that have been previously trained or modelled. In many cases, the most subtle and dangerous failures were the ones without consideration prior to the mission. Being able to autonomously detect unmodeled faults is critical to the health of a constellation of small satellites, given that many spacecraft operators cannot afford to dedicate specialized staff to monitor all telemetry on an on-going basis to look for faults.

To overcome this challenge, this paper introduces an autonomous, ground-based fault-detection system that was trained using only nominal data, without requiring any prior expert knowledge of the spacecraft systems. A One-Class Support Vector Machine (OC-SVM) with the combination of Sequential Minimal Optimization (SMO) method was implemented to achieve the automated fault-detection system. By observing nominal data during the commissioning phase of the satellite, the fault-detection algorithm learned how to differentiate normal data from abnormal data without a labelled set of abnormal data. Training and testing results are presented to show how this one-sided learning method of fault-detection could detect un-trained failures related to reaction wheel performance. Specifically, this paper demonstrates the utility of one-sided learning methods by autonomously detecting faults in reaction wheel bearing friction and wheel speed measurement, without any prior exposure to the failures.

RESEARCH BACKGROUND

A ground station provides a communication interface between the launched satellite and the satellite's operation team [2]. The satellite's operation team at the ground station monitors and tracks the performance of the satellites through downlinked data. With the expanding scale of small-satellite constellations, numerous satellites need to be periodically monitored and examined by operation teams. Additionally, the limited capacity of the ground station (*e.g.*, labor, costs, etc.) makes efficient commands and operations difficult to accomplish ([9], [10], [11]).

Automated fault-detection systems that use machine learning and data mining algorithms have been developed for various satellites to detect anomalies in real-time spacecraft telemetry ([2-7]). A similar fault-detection system can be developed for ground station operations to provide efficient satellite management.

Ibrahim et al. [7] proposed a fault diagnosis method based on an unsupervised machine learning algorithm to identify failures and anomalies of satellite subsystems for the Egyptsat-1, including the satellite's communication subsystem, on-board computer subsystem and power subsystem. Ibrahim et al. applied a general Support Vector Machine for Regression (SVM-R) method to learn and predict the bus voltage of the satellite from received time-series telemetry parameters. They used Logical Analysis of Data (LAD) to classify the binary categories of the satellite's behavior ("normal" and "abnormal") and generate the positive satellite behavior patterns. From the behavior patterns, a Fault Tree Analysis was used to determine the root cause and failure occurrence possibility for each subsystem.

Omran et al. [8] developed a fault-detection and identification system for reaction wheels in an attitude control system. The fault-detection system was created based on a Feed-Forward Neural Network (FFNN) with a back-propagation algorithm to detect if there was an anomaly in the reaction wheel voltage, current, and temperature. Firstly, the system used FFNN to predict the desired torque for a reaction wheel under a commanded bus voltage. Then, the normal operational torque curve was used as a comparable reference for anomaly detection. Failures could be detected as over-voltage, under-voltage, current gain or current loss using the residual signals between the predicted and real torque from the satellite's telemetry data.

Other approaches for fault detection were to identify and label data that deviated from acceptable ranges ([5], [6]). Specific features were selected from the flight measurements of the spacecraft and collected over a designated time. Filters were added before the training process to remove the abnormal data beyond the pre-defined lower and upper limits of satellite performance. The system was trained using the filtered data contained within most of the normal data. The detection system then would predict a theoretical model from the trained data to detect anomalies in real-time operations of the satellite.

From above, conventionally, most of the fault-detection systems were trained by using datapoints containing both normal behavioral data and abnormal behavioral data of the satellite (*i.e*, both positive data and negative data were given). Then, the system learned the "knowledge" from the two-sided training data and developed a binary classifier once the learning stage completed. Finally, the output binary classifier was used to predict failures and differentiate normal data and abnormal data during testing.

In this research, a one-sided learning algorithm was adopted to develop an automated fault-detection system for a ground station where only normal behavioral attitude control system data was involved in the training process. The algorithm employed to achieve one-sided learning was OC-SVM, originally proposed by Schölkopf et al. [12] in 1999 and has been used in a variety of one-class classification problems ([13-18]). From this approach, the system was trained by using only the dataset collected from a healthy attitude control system without any failure for a satellite during commissioning phase. With the learned knowledge from the nominal data, the fault-detection system was expected to identify and diagnose failure during real-time operation.

METHODOLOGY

This research aimed to develop an automated fault-detection system used from a ground station to identify and diagnose anomalies of a satellite's attitude control system. The development of the automated fault-detection system started from model simulation of an attitude control system for a small satellite. Following the training process, a normal dataset was generated and collected from the simulated attitude control system. With the collected dataset, the OC-SVM method was implemented for training. Details of the model simulation and algorithm application are presented below.

Model simulation

An attitude control system for a small satellite was created from a closed-loop feedback system using MATLAB / Simulink. The attitude control system plays a significant role in the pointing direction and orientation of a spacecraft ([19], [20]). Figure 1 shows a design concept of a closed-loop attitude dynamics control system.

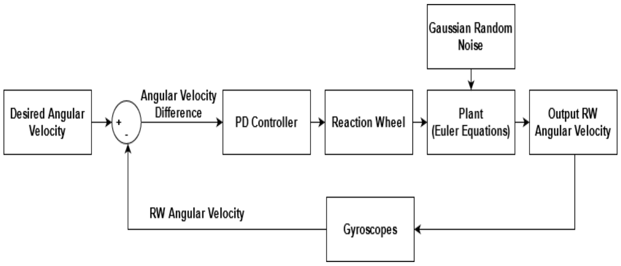


Figure 1: Attitude Dynamics Control System for a Spacecraft [2]

As presented in [2], the desired slew rates of the satellite were fed as inputs to the pointing control system. Reactions wheels were selected as actuators in the system to achieve the desired state of the angular velocity in three directions (x, y, and z-axis). A similar closed-loop feedback system was designed for reaction wheels where the commanded torque from the satellite was the input signal to the reaction wheel system (in Figure 2). The outputs from the reaction wheel dynamics system were reaction wheel momentum, torque, current, and angular velocity.

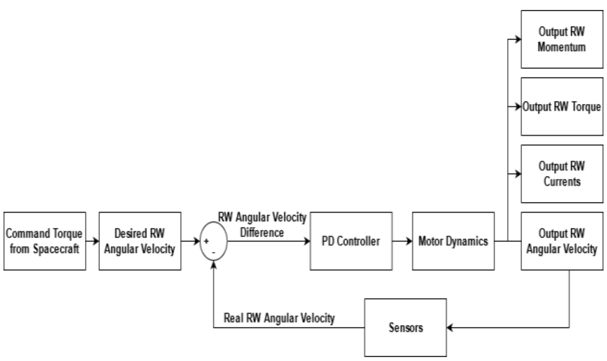


Figure 2: Reaction Wheel Dynamics [2]

Through Euler's equation (shown in Eq. (1)), the actual body rates of the satellite could be attained using the provided torque from the reaction wheel dynamic system [19]:

$$\boldsymbol{T}(\boldsymbol{t}) = \dot{\boldsymbol{h}}(\boldsymbol{t}) \tag{1}$$

Where $\boldsymbol{t}$ represents a time variable, $\boldsymbol{T}(\boldsymbol{t})$ is a torque vector provided by the reaction wheels with the satellite in time series, and $\boldsymbol{h}(\boldsymbol{t})$ is a vector presenting the angular momentum for the satellite in time series. The real body rates can be presented as a relationship between the provided torque by reaction wheels and the angular momentum:

$$\boldsymbol{h}(\boldsymbol{t}) = \boldsymbol{I}\boldsymbol{\omega}(\boldsymbol{t}) \tag{2}$$

$$\boldsymbol{T}(\boldsymbol{t}) = \boldsymbol{I}\dot{\boldsymbol{\omega}}(\boldsymbol{t}) \tag{3}$$

Where $\boldsymbol{I}$ is the moment of inertia of the satellite, $\boldsymbol{\omega}(t)$ is the output angular velocity vector, and $\dot{\boldsymbol{\omega}}(\boldsymbol{t})$ is the output angular acceleration vector.

As detecting a bearing degradation failure was the main purpose of this research, the torque provided by the reaction wheels to the satellite in Eq. (1) and (3) was computed after deducting the total friction from the reaction wheel system [2]. The total friction was:

$$\boldsymbol{f}_{total} = \boldsymbol{f}_{coulomb} + \boldsymbol{f}_{viscous} \tag{4}$$

$$\boldsymbol{T}(\boldsymbol{t}) = \boldsymbol{T}(\boldsymbol{t}) - \boldsymbol{f}_{total} \tag{5}$$

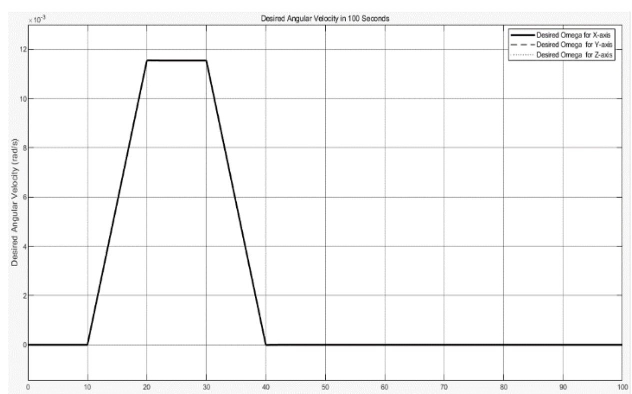
Where $\boldsymbol{f}_{total}$ is the total friction in the reaction wheel system, $\boldsymbol{f}_{coulomb}$ is the Coulomb friction, and $\boldsymbol{f}_{viscous}$ is the viscous friction.

Neglecting the Coulomb friction in this research, the total friction in the reaction wheels was denoted as:

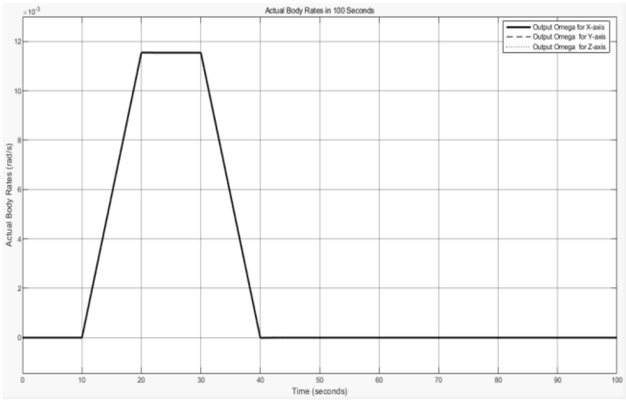
$$\boldsymbol{f}_{total} = \boldsymbol{f}_{viscous} = m\boldsymbol{f}_{viscous\_nominal} \tag{6}$$

Where $\boldsymbol{f}_{viscous\_nominal}$ is the nominal viscous friction calculated from real reaction angular velocity and viscous friction coefficient to compute all normal slews, and m is a scaled factor which will be identified as a failure indicator through training process (if m =1, $\boldsymbol{f}_{viscous} = \boldsymbol{f}_{viscous\_nominal}$) [2].

Figure 3 shows an example of a 100-second nominal slew of a small satellite from the simulated closed-loop pointing control system at the commissioning phase [2].



a)



b)

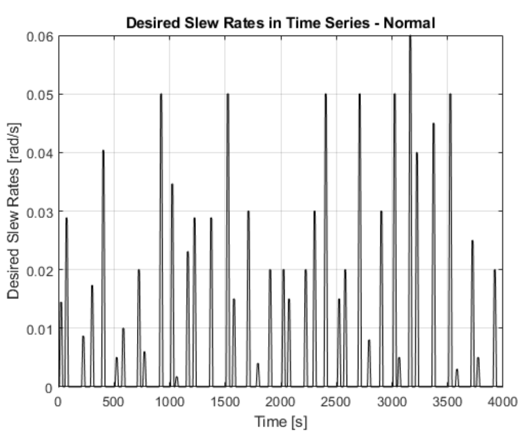
Figure 3: Simulated Closed-Loop Attitude Control System in 100 Seconds: a) Desired Slew Rates of a Spacecraft, b) Actual Body Rates of the Spacecraft [2]

The training dataset was prepared from the simulated closed-loop attitude control system; 40 random healthy slews were created in a 4000-second nominal simulation.

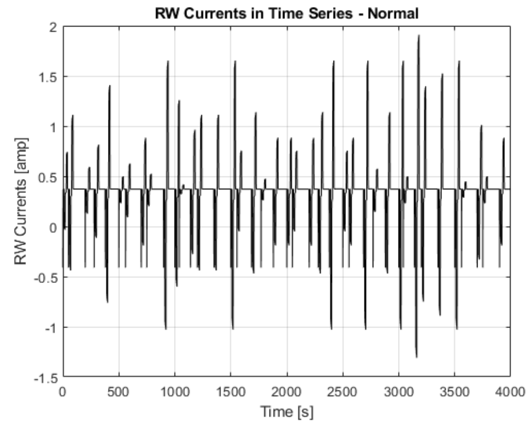
Refer to [2], the reaction wheel current was simulated from:

$$\boldsymbol{T_m}(\boldsymbol{t}) = K_t \boldsymbol{i}(\boldsymbol{t}) \tag{7}$$

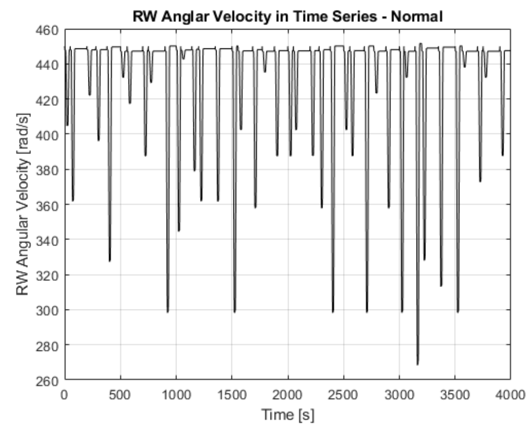
Where $\boldsymbol{T_m}$ represents a torque vector provided by motor in reaction wheel dynamics (Figure 2), K_t represents the torque coefficient of the motor, and $\boldsymbol{i}$ is a vector presenting the reaction wheel current in time series. Then, the reaction wheel current and angular velocity were selected as the training features over the other outputs from the reaction wheel dynamics due to their mutual independence. The outputs of the desired slew rates with reaction wheel current and angular velocity in 4000 seconds are shown in Figure 4.



a)



b)



c)

Figure 4: Normal Slews in 4000 Seconds: a) Desired Slew Rates, b) Corresponding Reaction Wheel Currents, c) Corresponding Reaction Wheel Angular Velocities

One-Class Support Vector Machine

In this section, the OC-SVM methodology and equations are from [12]. The principle of using this method is that labels or data responses are not required for training. Therefore, the training process only included normal behavioral attitude control system data. As reaction wheel current and angular velocity were selected as the training features, 10 consecutive data points were collected for reaction wheel current and angular velocity at each time point, to enable the algorithm to deduce relevant time-series features. Hence, there were 4000 data points collected for training with a data rate of 10Hz, and each training data contained 10 consecutive data points of reaction wheel currents and angular velocities. The training data then was given as:

$$\boldsymbol{X} = \{(\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_N})\} \tag{8}$$

$$\boldsymbol{x} = \{(\boldsymbol{i}, \boldsymbol{\omega})\} \quad (9)$$

Where $\boldsymbol{X}$ is the whole training data set of nominal data in 4000 seconds, N is the total number of training data sets, $\boldsymbol{x}$ is a subset of training data, $\boldsymbol{i}$ is a vector represents 10 consecutive nominal reaction wheel currents at one timepoint and $\boldsymbol{\omega}$ is a vector that represents 10 consecutive nominal reaction wheel angular velocities. All nominal data were used to train the detection system using OC-SVM to differentiate the behavioral labels. The OC-SVM algorithm trained the system to learn the decision hyperplane with maximum margin to separate normal behavioral data points with untrained abnormal behavioral data points from the origin in its feature space. Before training the system, the input training data was first mapped by a Kernel function into the feature space [21]:

$$K(\boldsymbol{X}, \boldsymbol{X}') = \Phi(\boldsymbol{X})\,\Phi(\boldsymbol{X}') \quad (10)$$

Where, K is the Kernel function (*e.g.*, Gaussian, linear, sigmoid, polynomial Kernel functions), and Φ is a mapping function.

General kernel functions that can be used for OC-SVM are listed in Table 1. σ, α, c , and d shown in the functions below are the tuning parameters in the process of training.

Table 1: General Kernel Functions for OC-SVM ([14], [22], [23])

Name	Kernel Functions
Gaussian	$k(x_i, x_j) = exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2})$
Linear	$k(x_i, x_j) = \alpha x_i^T x_j + c$
Sigmoid	$k(x_i, x_j) = tanh(\alpha x_i^T x_j + c)$
Polynomial	$k(x_i, x_j) = (\alpha x_i^T x_j + c)^d$

In this research, the sigmoid Kernel function was used to map the training features. To obtain the decision function of the separable hyperplane between normal and untrained-abnormal attitude behavioral data of the satellite, a quadratic function was defined as:

$$\min_{w \epsilon F, \xi \epsilon R^N, \rho \epsilon R} \frac{1}{2}\|w\|^2 + \frac{1}{\nu N}\sum_{i=1}^{N} \xi_i - \rho \quad (11)$$

Subject to $\left(w \cdot \Phi(x_i)\right) \geq \rho - \xi_i$

$\xi_i \geq 0, \forall i = 1, \dots, N$

Where w and ρ are the parameters of the hyperplane decision function and are computed after iterations in training, w is the coefficient of the decision function and ρ is the offset of the decision function. $\frac{1}{2}\|w\|^2$ is a regularizer term of the function [24]. ν is a trade-off regularization parameter with a range of (0,1) [18]. ξ is the slack variable in the training data.

To solve the quadratic problem above, a Lagrange function was then proposed to solve:

$$L(w, \boldsymbol{\xi}, \rho, \alpha, \beta) = \frac{1}{2}\|w\|^2 + \frac{1}{\nu N}\sum_{i=1}^{N} \xi_i - \rho$$

$$- \sum_{i=1}^{N} \alpha_i \left(\left(w \cdot \Phi(x_i)\right) - \rho + \xi_i\right) - \sum_{i=1}^{N} \beta_i\, \xi_i \quad (12)$$

Where, α_i and β_i are the positive multiplier parameters in the Lagrange function [25].

After taking partial derivatives of the Lagrange function with regard to w, ρ and ξ, both w and α_i then could be solved by setting the derivative functions to zero (shown in Eq. (13) and (14)):

$$w = \sum_{i=1}^{N} \alpha_i \Phi(x_i) \quad (13)$$

$$\alpha_i = \frac{1}{\nu N} - \beta_i \leq \frac{1}{\nu N}, \sum_{i=1}^{N} \alpha_i = 1 \quad (14)$$

By substituting Eq. (11) and (12) into Eq. (10), the dual problem becomes:

$$\min_{\alpha} \frac{1}{2}\sum_{i=1, j=1}^{N} \alpha_i \alpha_j\, K(\boldsymbol{x_i}, \boldsymbol{x_j}) \quad (15)$$

Subject to $0 < \alpha_i \leq \frac{1}{\nu N}, \sum_{i=1}^{N} \alpha_i = 1$

$$\forall i = 1, \dots, N, \forall j = 1, \dots, N$$

with $\alpha_i \epsilon\ (0, \frac{1}{\nu N}]$ and $\beta_i \neq 0$, the last parameter required for the decision function of the hyperplane, ρ, can be computed by combining Eq. (9) and (13) as:

$$\rho = \left(w \cdot \Phi(x_i)\right) = \sum_{j=1}^{N} \alpha_j K(\boldsymbol{x_j}, \boldsymbol{x_i}) \quad (16)$$

Therefore, the general function of the separable hyperplane to determine the behavioral labels of the satellite with Kernel mapping functions was:

$$f(\boldsymbol{x}) = (w \cdot \Phi(x_i)) \geq \rho$$

$$f(\boldsymbol{x}) = sgn(\sum_{i=1}^{N} \alpha_i k(x_i, x) - \rho) \quad (17)$$

Where x_i in N number of training data is treated as a positive datapoint which is responsible for computing the decision function of the hyperplane for satellite attitude behavior classification.

After learning the hyperplane decision function from all 4000-second nominal training data, the detection system should be able to identify failures. Test data is fed into Eq. (17) with tuned parameters: α_i, w and ρ. Then, labels are assigned to each test datapoint. Label 0 is assigned to all normal data during testing (decision function indicates positive), and label 1 is assigned to all abnormal data (decision function indicates negative).

$$f(\boldsymbol{x_{test}}) = sgn(\sum_{i=1}^{N} \alpha_i K(\boldsymbol{x_i}, \boldsymbol{x_{test}}) - \rho) \quad (18)$$

$$y = \begin{cases} 0, & f(x_{test}) \geq 0 \\ 1, & f(x_{test}) < 0 \end{cases} \quad (19)$$

Where, $\boldsymbol{x_{test}}$ is test dataset. When $f(x_{test}) \geq 0$, it represents that the test datapoint falls inside of the defined decision hyperplane. When $f(x_{test}) < 0$ it represents that the test datapoint falls outside of the defined decision hyperplane. Therefore, if $f(x_{test})$ shows a positive non-zero value, the predicted label for the test datapoint will be 0, otherwise the datapoint will be labelled as 1.

Sequential Minimal Optimization

The Sequential Minimal Optimization (SMO) method was used to define the decision function of the hyperplane better. The origin of the SMO method was proposed by Platt [26] to solve the quadratic problem in the general Support Vector Machine (SVM) method and to provide a faster and more efficient solution. An adaptive SMO algorithm was applied in this research to provide an efficient approach to solve the quadratic problem in the OC-SVM method. The adaptive SMO algorithm was modified by Schölkopf et al. and the following methodology and equations are referenced to [12], [26].

As mentioned in the previous section, α_i, w and ρ were the tuned parameters through the training process where w and ρ were dependent on α_i. Thus, the principal optimization problem for the OC-SVM method is to optimize α_i. By applying the SMO method, a pair of α(s) will be selected and optimized for each iteration. For example, the optimizing function for α_1 and α_2 are as follows:

$$\min_{\alpha_1,\alpha_2} \frac{1}{2} \sum_{i=1,j=1}^{2} \alpha_i \alpha_j K_{ij} + \sum_{i=1}^{2} \alpha_i C_i + C \quad (20)$$

$$C_i = \sum_{j=3}^{N} \alpha_j K_{ij}, \; C = \sum_{i=3,j=3}^{N} \alpha_i \alpha_j K_{ij}$$

Subject to $0 \leq \alpha_1 \leq \frac{1}{\nu N}, 0 \leq \alpha_2 \leq \frac{1}{\nu N}$

Where, α_1 and α_2 are the selected pair from α_i for optimization. From linear equality constraint of α_1 and α_2, the summation of α_1 and α_2 is the same before and after optimization ([26], [27], [28]). Therefore,

$$\alpha_1 + \alpha_2 = s \quad (21)$$

Where s represents the sum of the selected α_1 and α_2 before optimization. Then, the optimized α_2 could be updated as:

$$\alpha_{2\_new} = \frac{s(K_{11}-K_{12})+C_1-C_2}{K_{11}+K_{22}-2K_{12}} \quad (22)$$

From Eq. (21), the optimized α_1 could be attained:

$$\alpha_{1\_new} = s - \alpha_{2\_new} \quad (23)$$

For every step of optimization, the offset parameter of the decision function would require re-calculated and updated using Eq. (24).

$$\rho = \sum_{j=1}^{N} \alpha_j k(\boldsymbol{x_j}, \boldsymbol{x_i}) \quad (24)$$

The stopping criteria for SMO were the filter conditions for α optimization following the Karush-Kuhn-Tucker (KKT) conditions [29]. The first α selected for optimization was from the whole training dataset where any α that violated one of the KKT conditions would be chosen for SMO (shown in Eq. (25) and (26))

$$(f(\boldsymbol{x_i}) - \rho) \cdot \alpha_i > 0, \text{ or} \quad (25)$$

$$\left(\rho - f(\boldsymbol{x_i})\right) \cdot \left(\frac{1}{\nu N} - \alpha_i\right) > 0 \quad (26)$$

Where $f(\boldsymbol{x_i})$ is derived from Eq. (17).

$$f(\boldsymbol{x_i}) = K_{1i}\,\alpha_1 + K_{2i}\alpha_2 + C_i \quad (27)$$

Where C_i is denoted as $C_i = \sum_{j=3}^{N} \alpha_j K_{ij}$.

The second α, α_j, can be selected using the argument in Eq. (28):

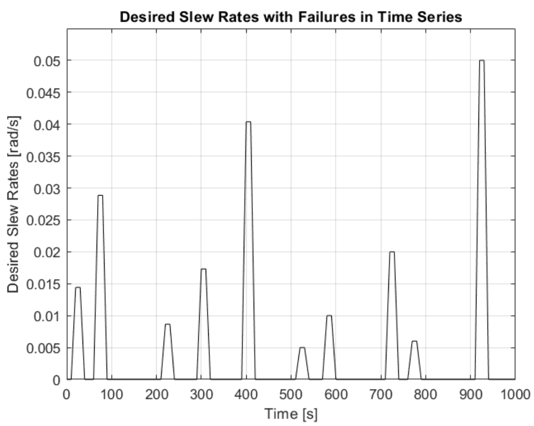
$$j = arg \max_{n \epsilon N} |\, f(\boldsymbol{x_i}) - f(\boldsymbol{x_n})| \quad (28)$$

Once every α has satisfied the KKT conditions (listed in Eq. (25) and (26)), the learning process is terminated. With the optimized output α(s), tuning parameters, w and ρ, were then could be computed by Eq. (13) and (16).

A Library for Support Vector Machines (LIBSVM) was used to implement the learning algorithm of OC-SVM with SMO method ([12], [23], [24], [30]). Two tests were performed on the trained system using LIBSVM to validate the feasibility of the method.

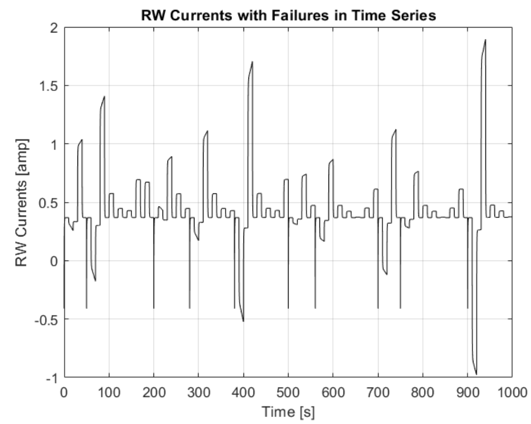
TESTS AND RESULTS

Two 1000-second tests were performed for the trained detection system. Huang et. al [2] provided an outline for the tests. Each test contained ten random slews with simulated bearing degradation failures. The bearing degradation failures were created based on Eq. (6). Adjusting the value of the viscous friction, m, would affect the performance of the attitude control system. Figure 5 shows an example of the effect of friction on the performance of the reaction wheel. The friction failures were added for every 10 seconds.

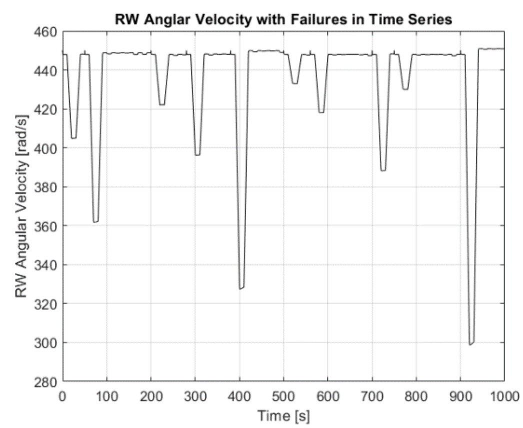


a)

b)



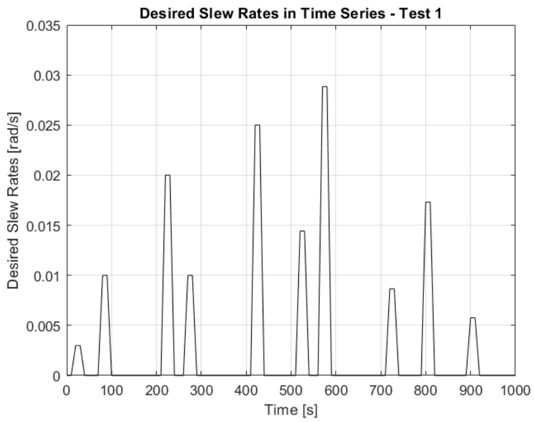
c)



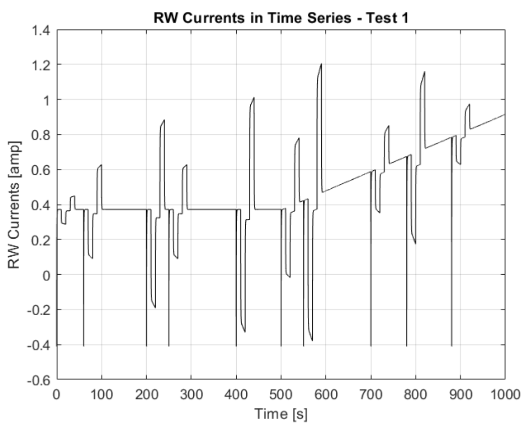
d)

Figure 5: Slews with Mimic Bearing Failures in 1000 Seconds: a) Desired Slew Rates with Failures, b) Viscous Frictions with Failures in 1000 Seconds, c) Corresponding Reaction Wheel Currents, d) Corresponding Reaction Wheel Angular Velocity [2]

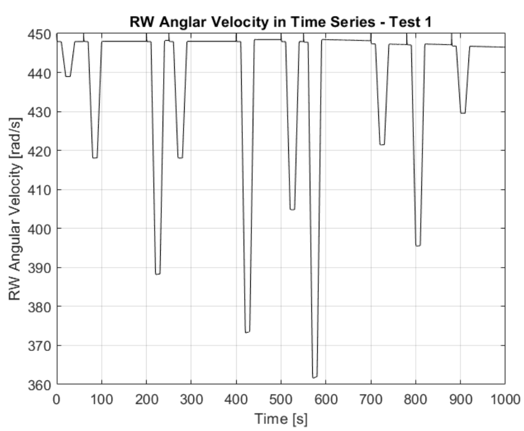
In the first test, friction failures were added at the 500th second of the simulation. At the 500th second, the friction failure occurred and began to increase gradually with time. For the second test, friction failures were added in two different phases of the simulation following trapezoidal-like viscous friction profiles. The magnitude of friction failure from each phase was increased with time, then remained at a constant level for a short time (20-30s). Finally, the friction failure decreased back to the nominal value of the friction. The purpose of performing these two tests was to examine the detection capability of the trained system on long-term failures and intermittent failures in the attitude control system. Figure 6 and Figure 7 show the plots for the two test datasets (the same test data was used in [2], but for a different fault detection methodology). The corresponding reaction currents and angular velocities are also presented.



a)

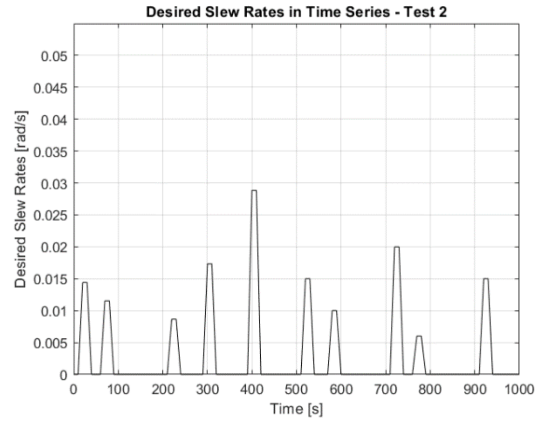


b)

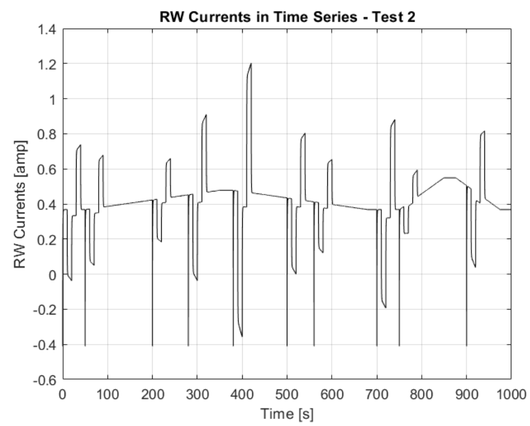


c)

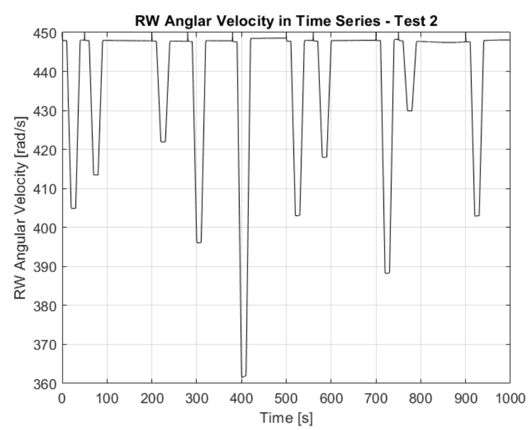
Figure 6: Test 1 Dataset: a). Desired Slew Rates – Test 1, b) Corresponding Reaction Wheel Current - Test 1, c) Corresponding Reaction Wheel Angular Velocity – Test 1 [2]



a)



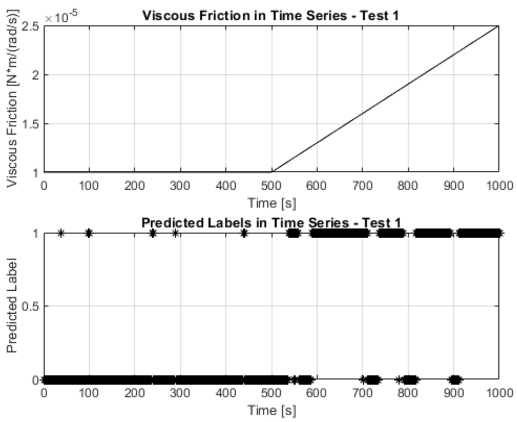
b)



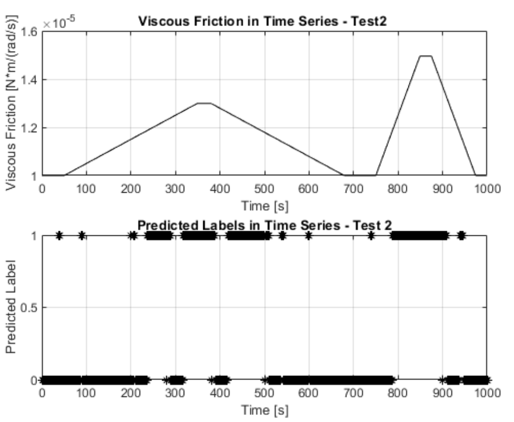
c)

Figure 7: Test 2 Dataset: a). Desired Slew Rates – Test 2, b) Corresponding Reaction Wheel Current - Test 2, c) Corresponding Reaction Wheel Angular Velocity – Test 2 [2]

The tests were then performed using LIBSVM ([12], [30]). The behavioral labels for the two tests were predicted by applying Eq. (18) and Eq. (19). The predicted results are presented in Figure 8. Comparing the results from the two tests, the proposed one-sided learning-based detection system showed good performance on test 1, where the detection accuracy reached approximately 90%. However, the system showed a lower detection accuracy on test 2 where the detection accuracy was around 60%. From the results, it is observed that the detection system showed a better performance on identifying and diagnosing normal data points through slews.



a)



b)

Figure 8: Prediction Results for Tests: a). Viscous Friction Vs. Predicted Behavioral Labels for Test 1, b). Viscous Friction Vs. Predicted Behavioral Labels for Test 2.

CONCLUSION

A ground-based automated fault-detection system for a small satellite has been developed in this research, utilizing the OC-SVM algorithm, to achieve a one-sided learning algorithm that detects reaction wheel bearing failures in an attitude control system. The fault-detection system was trained using data only from normal attitude control behaviors of the satellite over a designated time. A one-class classifier was created and trained via LIBSVM, and underwent two separate tests to examine its detection capability for various forms of failures. The results showed that the detection system demonstrated a good performance on detecting long-term failures and performed relatively poorly when detecting intermittent failures. This one-sided learning-based detection system also performed better when detecting normal data as compared to detecting anomalies. Future research will put effort into developing different fault-detection systems by implementing other suitable one-sided learning algorithms.

ACKNOWLEDGEMENT

We would like to thank NSERC and MDA for providing funding to this research.

(The test data presented in this paper were originally presented in “Attitude Dynamics and Control Anomaly Detection Using an Autonomous Ground Station” published in AIAA SciTech 2021 Forum by Yujia Huang and Philip Ferguson, copyright © 2021 by Yujia Huang. The training and test data used in this analysis with a different detection algorithm is presented here with permission from the American Institute of Aeronautics and Astronautics, Inc.)

REFERENCES

[1] A. E. Hassanien, A. Darwish and E.-A. Hesham, Machine Learning and Data Mining in Aerospace Technology, Springer Nature, 2020.

[2] Y. Huang and P. A. Ferguson, "Attitude Dynamics and Control Anomaly Detection Using an Autonomous Ground Station," in AIAA SciTech Forum, Online, 2021.

[3] T. Yang, B. Chen, Y. Gao, J. Feng, H. Zhang and X. Wang, "Data mining-based fault detection and prediction methods for in-orbit satellite," in Proceedings of 2013 2nd International Conference on Measurement, Information and Control, 2013.

[4] Q. Li, Z. XingShe, L. Peng and L. Shaomin, "Anomaly Detection and Fault Diagnosis Technology of Spacecraft based on Telementry-mining," in 3rd International Symposium on

Systems and Control in Aeronautics and Astronautics, 2010.

[5] T. Yairi, Y. Kawahara, R. Fujimaki and Y. Sato, "Telemetry-mining: A Machine Learning Approach to Anomaly Detection and Fault Diagnosis for Space Systems," in 2nd IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT'06), 2006.

[6] Y. Gao, T. Yang, M. Xu and N. Xing, "An Unsupervised Anomaly Detection Approach for Spacecraft Based on Normal Behavior Clustering," in Fifth International Conference on Intelligent Computation Technology and Automation, 2012.

[7] S. K. Ibrahim, A. Ahmed, M. A. E. Zeidan and I. E. Ziedan, "Machine Learning Techniques for Satellite Fault Diagnosis," Ain Shams Engineering Journal, vol. 11, no. 1, pp. 45-56, 2019.

[8] E. A. Omran and W. Murtada, "Efficient anomaly classification for spacecraft reaction wheels," Neural Computing and Applications, vol. 31, no. 5, pp. 2741-2747, 2017.

[9] G. Minelli, M. Karpenko, I. M. Ross and J. Newman, "Autonomous operations of large-scale satelliteconstellations and ground station networks," in AAS/AIAA Astrodynamics Specialist Conference, 2017.

[10] G. Gruzi, D. Modenini and P. Tortora, "Large Constellations of Small Satellites: A Survey of Near Future Challenges and Missions," Aerospace , vol. 7, no. 9, p. 133, 2020.

[11] J. Howard, D. Oza and D. S. Smith, "Best Practices for Operations of Satellite Constellations," in Proceedings of the 9th International Conference on Space Operations, 2006.

[12] B. Schölkopf, J. C. Platt, A. Smola and R. C. Williamson, "Estimating the Support of a High-Dimensional Distribution," Neural Computation, vol. 12, no. 7, pp. 1443-1471, 2001.

[13] L. M. Manevitz and M. Yousef, "One-Class SVMs for Document Classification," Journal of Machine Learning Research , pp. 139-154, 2001.

[14] T. Hofmann and S. B, "Kernel Methods in Machine Learning," The Annals of Statistics, vol. 36, no. 3, pp. 1171-1220, 2008.

[15] B. Schölkopf, R. Williamson, A. Smola, S.-T. John and J. Platt, "Support vector method for novelty detection," in Proceedings of the 12th International Conference on Neural Information Processing Systems, 1999.

[16] J. Zhou, K. L. Chan, V. F. H. Chong and S. M. Krishnan, "Extraction of Brain Tumor from MR Images Using One-Class Support Vector Machine," in IEEE Engineering in Medicine and Biology 27th Annual Conference, Shanghai, China, 2005.

[17] J. Santiago-Paz, D. Torres-Roman, A. Figueroa-Ypiña and J. Argaez-Xool, "Using Generalized Entropies and OC-SVM with Mahalanobis Kernel for Detection and Classification of Anomalies in Network Traffic," Entropy, vol. 17, no. 9, pp. 6239-6257, 2015.

[18] J. Ma and S. Perkins, "Time-series novelty detection using one-class support vector machines," in Proceedings of the International Joint Conference on Neural Networks, 2003.

[19] P. C. Hughes, "Spacecraft Attitude Dynamics," in Dover Publications, Inc., Mineola, NY, 2004.

[20] M. F. Landis and J. L. Crassidis, Fundamentals of Spacecraft Attitude Determination and Control, Springer, 2016.

[21] V. N. Vapnik, Statistical Learning Theory, Wiley-Interscience, 1998.

[22] B. Schölkopf and A. J. Smola, Learning with kernels: support vector machines, regularization, optimization, and beyond, MIT press, 2002.

[23] H. J. Shin, D.-H. Eom and S.-S. Kim, "One-class support vector machines—an application in machine fault detection and classification,"

Computers & Industrial Engineering, vol. 48, no. 2, pp. 395-408, 2005.

[24] C. C. Aggarwal, "3.4 One-Class Support Vector Machines," in Outlier Analysis, Springer, Cham, 2017, pp. 88-95.

[25] S. Yin, X. Zhu and C. Jing, "Fault detection based on a robust one class support vector machine," Neurocomputing, vol. 145, no. 5, pp. 263-268, 2014.

[26] J. C. Platt, "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines," Microsoft Research, 1998.

[27] J. Jia and H.-r. Zhang, "A Fast Learning Algorithm for One-Class Support Vector Machine," in Proceedings of the Third International Conference on Natural Computation, 2007.

[28] H. Wang, G. Zhao and H. Gu, "A fast training method for OC-SVM based on the random sampling lemma," in Sixth International Conference on Natural Computation, 2010.

[29] D. P. Bertsekas, Nonlinear Programming, Athena Scientific, 1995.

[30] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," ACM Transactions on Intelligent Systems and Technology, vol. 2, no. 3, pp. 27:1-27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm. .