INVESTIGATING ADVANCED AOCS TECHNIQUES TO IMPROVE ROBUSTNESS AND RELIABILITY OF LOW-COST MISSIONS

Garima Pandey[1], Giovanni Lapicirella[2], Marco Lovera[3], Davide Invernizzi[4], Salvatore Meraglia[5], Carole Rosso[6], Ilario Cantiello[7]

[1]Airbus Defence and Space GmbH, Immenstaad, Germany, garima.pandey@airbus.com
[2]Airbus Defence and Space GmbH, Immenstaad, Germany, giovanni.lapiccirella@airbus.com
[3]Politecnico Di Milano, Milan, Italy, marco.lovera@polimi.it
[4]Politecnico Di Milano, Milan, Italy, davide.invernizzi@polimi.it
[5]Politecnico Di Milano, Milan, Italy, salvatore.meraglia@polimi.it
[6]ESA/ESTEC, Noordwijk, The Netherlands, carole.rosso@esa.int
[7]ESA/ESTEC, Noordwijk, The Netherlands, ilario.cantiello@esa.int

ABSTRACT

Commercial-Off-The-Shelf (COTS) equipment for the AOCS subsystem bring several advantages such as high processing power, lower mass/volume, lower power usage, and low cost. Their across-the-board adoption in space is however currently confronted by their robustness and reliability issues. Their susceptibility to single event upsets, unpredictable end-of-life performances, and maybe even permanent failures, directly impact mission availability and performance. To address these challenges, this paper proposes advanced and innovative control and FDIR techniques. In particular, the aim is early FDIR targeting a varied selection of failure cases for a variety of COTS AOCS equipment. A range of state-of-the-art algorithms are investigated, some of which have shown great potential in other flight control applications like UAVs. Innovative FDI techniques like Adaptable estimation, Model-based FDI, and Machine-learning based FDI, as well as advanced control techniques like adaptive control, and Non-holonomic control are studied to assess their potential benefits when coping with COTS failures. These techniques were prototyped and implemented in an FDIR focussed AOCS simulator (GAFE). To demonstrate possible advantages for space missions compared to standard FDIR practices in the industry, the techniques are applied to two mission use cases, EarthCARE and OneSat, covering different applications, orbit regime, environment, and design philosophies, thus providing a varied set of mission performance and availability requirements to be fulfilled by the AOCS subsystem. The paper then goes on to present the test results for each of the techniques as applied to chosen fault cases with varying fault types, magnitudes, and dynamic conditions. The detection and isolation capabilities and recovery performance are used as criteria to compare to classical fault management concepts for particular failure cases in the respective missions. The paper also discusses driving aspects and challenges of implementation of the methods for the missions as well as the potential benefits of their use as compared to classical FDI and control techniques for low-cost COTS equipment.

1 INTRODUCTION

The current rapid growth in the small satellite, large constellation market is evident across the space industry. With it, reducing costs at each level of system development is becoming an increasingly driving constraint. For large constellation missions, hardware procurement is a cost driver due to the sheer number of components, while for small satellite missions, the share of hardware in total spacecraft cost is relatively large. In these contexts, therefore, the interest towards shifting to standard, low-cost Commercial-Off-The-Shelf (COTS) equipment is enormous. The cost of a hardware unit for

space applications is driven by both, design for high performance and reliability in space environment, and quality control practices. Lower cost COTS equipment is thus inherently confronted by robustness and reliability issues, coming from both design as well as quality control practices, like lack of characterisation in radiation environment, shorter lifetime at design, and poor lot control.

A survey and analysis of Attitude and Orbit Control Systems (AOCS) COTS equipment showed that these issues manifest themselves in a variety of failure signatures at equipment, AOCS, and Fault Detection, Isolation and Recovery (FDIR) levels. While the failure characterisation does not differ from that of nominal hardware, the frequency of failure and probability of its occurrence is higher for COTS equipment. Classical, threshold- and/or redundancy-based FDIR techniques, while entirely capable of handling the failure types, might not be scalable to large constellation and small-sat context from a cost perspective.

This work thus focuses on addressing these COTS AOCS equipment failures using advanced AOCS and FDIR algorithms, assess their performance for FDIR and compare the results with classical AOCS and FDIR design solutions. The aim is to increase mission availability and to maintain performance, despite the use of low-cost COTS hardware.

To that end, a number of advanced estimation, control and FDI techniques were surveyed. Criteria such as robustness, ease of implementation, processing power needs, maturity etc. were evaluated base on the literature. This, complemented with an assessment of their suitability to COTS failure cases, was used to trade and select five of them. These five techniques are:

- Adaptable estimation,
- Model-based FDI,
- Machine-learning based FDI,
- Adaptive control, and
- Non-holonomic control

The paper describes the evaluation of these techniques going from development of the algorithm prototypes to assessment of their performance when applied to specific mission use cases and failure types. These performances are compared to standard methods and algorithms.

2 ALGORITHM ASSESSMENT

The developed prototypes were implemented on simulations of one of two mission use cases: EarthCARE or OneSat. The mission use cases were chosen based on their representativeness and complementarity. EarthCARE represents a typical classical low-earth-orbit earth observation mission with medium-high reliability and performance requirements. It on-boards classical space qualified equipment. Complementarily, OneSat represents geostationary earth orbit platform typically with telecom payloads. The OneSat product line comprises automotive qualified COTS equipment.

The algorithms were designed targeting a variety of failure types for different AOCS equipment. The test cases for assessment encompassed different orbit regimes (a result of the mission use case), dynamic conditions, modes, and failure types and magnitudes. Table 2-1 provides a top-level summary of the assessment cases for each technique. The assessment was supported by comparison to the nominal FDIR techniques on the two missions for the criteria, also shown in Table 2-1.

Table 2-1: Summary of implementation and test cases for individual algorithm assessment

Technique		Mission	Equipment	Failure in	Failure type	Assessment Criteria
1.	Adaptable estimation	OneSat	Star-tracker	Output Euler Angle	Output data freeze, drift	Failure detection time
		Earth-CARE	Magnetometer	Output Magnetic Field	Output data freeze, drift	Failure detection time
			Star-tracker	Output Euler Angle	Output data freeze, drift	Failure detection time
2.	Model based FDI	Earth-CARE	Reaction wheel	Friction Torque	Coulomb friction bias and drift	Failure detection time, Sensitivity to model, operational constraints
3.	ML Based FDI	OneSat	Star-tracker	Output Euler Angle	Output drift with varying amplitudes	Failure detection time
4.	Adaptive control	Earth-CARE	Reaction wheel	Output Actuator Torque	Output bias and drift of varying magnitudes	Control performance, convergence
5.	Non-Holonomic	OneSat	Reaction wheel	Output Actuator Torque	Complete failure (under-actuated spacecraft)	Control performance, convergence

The Generic AOCS/ Guidance Navigation and Control (GNC) Techniques and Design Framework (GAFE) simulator [1] was used to implement the assessment cases except for non-holonomic assessment, which was performed in a simplified simulator. GAFE is an FDIR focussed closed loop AOCS simulator providing convenient mechanisms to model equipment and algorithms based on existing libraries as well as create new algorithms. It allows for extensive treatment of failure description and induction for all equipment types, and easy implementation of parameter and functional monitors.

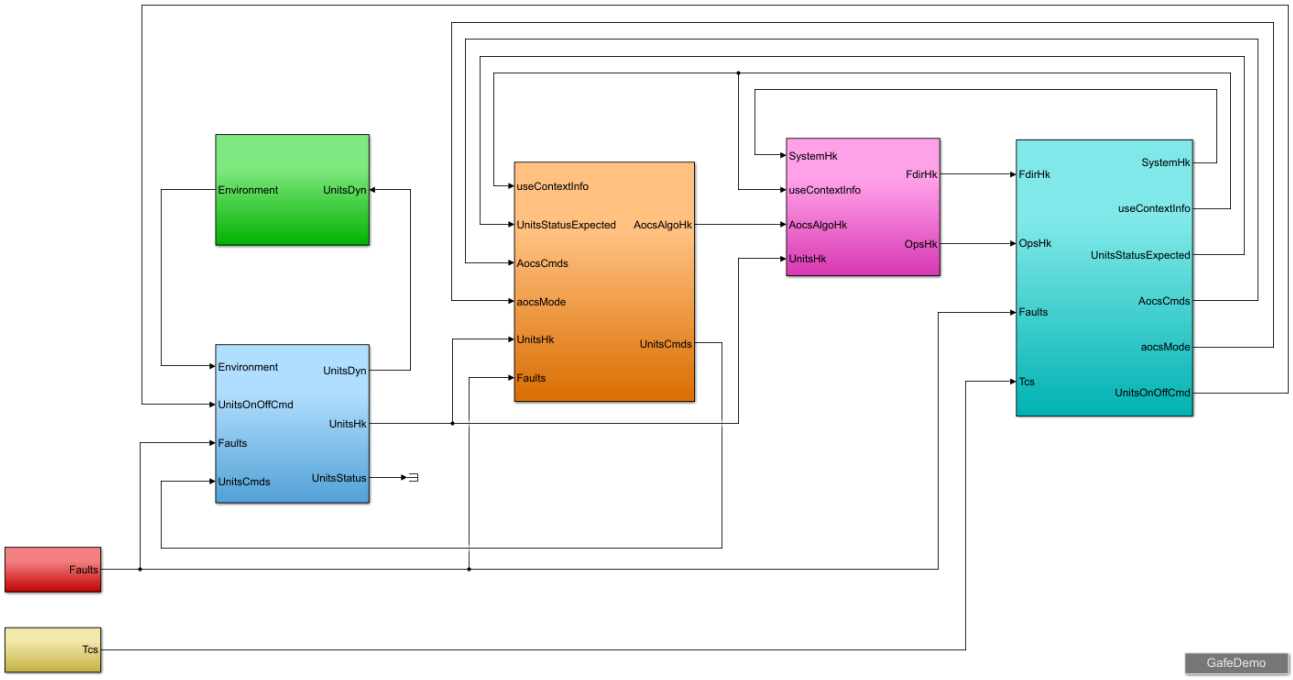

Figure 2-1 GAFE Simulator Top Level Blocks [1]

2.1 Adaptable Estimation

The formulation of the adaptable estimation algorithm relies on a dynamic model describing the relationship between the physical model and the measurement provided by the equipment. The formulation of an estimation problem allows recognising the discrepancy between the two.

For the case of the magnetometer, a dynamic model is derived that the geomagnetic field should satisfy. This is done as follows: the measurement equation for the magnetometer is given by:

$$b = Ab_0 \quad \textbf{(1)}$$

where b denotes the magnetic field measurement in the Spacecraft body frame, A is the attitude matrix and b_0 denotes the components of the magnetic field in the orbital reference frame. Differentiating with respect to time:

$$\dot{b} = S(\omega)b + A\dot{b_0}. \quad \textbf{(2)}$$

where ω is the vector of body angular rate. Discrete time formulation as well as the magnetometer output equations are then equations 3 and 4 respectively:

$$b(k+1) = \left(I + \Delta t S\left(\omega(k)\right)\right) b(k) + \Delta t A(k)\dot{b_0} \quad \textbf{(3)}$$

$$y(k) = b(k) + n(k) \quad \textbf{(4)}$$

where n represents the magnetometer's measurement noise (assumed white).

A one-step-ahead adaptive Kalman predictor is designed using this model to check sanity of the magnetic field measurement against the model and attitude and angular rate measurements. An adaptation scheme is considered to take variability or uncertainty in the process noise covariance into account. This is defined for the process noise covariance matrix Q as:

$$Q(k) = \alpha Q(k-1) + (1-\alpha)\bar{K}_k d(k) d'(k) \bar{K}_k{}^T, \quad \textbf{(5)}$$

Where α is the gain of the adaptive filter and $d(k)$ is the difference between the actual measurement and its predicted value.

The failure detection in the equipment unit is performed by observing the variations of the norm of Q. Fast variations of the values of the variance matrix are observed when a failure has occurred. This information is used to activate a flag, which indicates that the occurrence of failure. A preliminary tuning was performed to set the threshold of Q norm variation that triggers failure detection.

The adaptable estimation algorithm was implemented for two equipment, Star-trackers and Magnetometers. Adaptable estimation for star-trackers was also incorporated and tested on both missions, OneSat and EarthCARE, as shown in Table 2-1. OneSat does not have on-board a magnetometers. A comparison with the standard FDI is shown. Note that in standard FDI design, detection of data freeze occurs in the unit measurement-processing algorithm with simple threshold based checks. The detection of drifts occurs via intra-equipment check at estimation level, and thus needs redundant equipment. A summary of the tests performed along with failure parameters is provided in Table 2-2.

Table 2-2 Test result summary of adaptable estimation

Star-tracker		**Dynamic Condition**	**Mission**	**Detection Time – from injection**	
Failure type	**Magnitude**			**Standard existing FDI in mission**	**Adaptable Estimation**
Data freeze	N/A	Steady State	EarthCARE	0.5 s	0.5 s

Drift	0.3°/s	Steady State	OneSat	200 s	20 s
	0.3°/s	Steady State	EarthCARE	8.5 s	0.5 s
	0.3°/s	Slew Manoeuvre	EarthCARE	Not Detected	Not Detected
Magnetometer					
Data freeze	N/A	Steady State	EarthCARE	35.5 s	0.5 s

In general, the adaptable estimation algorithm shows comparable or lower failure detection time than the standard algorithm implemented in the mission. Further, the algorithm can isolate the failure to the unit, also in case of slow drifts, without the need for analytical/equipment redundancy.

One of the main challenges of using adaptable estimation for detection is the characterisation of the variation of the norm of Q. While it should ideally remain constant in the absence of anomalies (failures), in real simulations variations can occur in mission "nominal" situations triggering a false detection. For star-trackers for example, high noise in rate estimation led to jumps in norm of Q. This was not observed if rate was provided though the Gyro.

2.2 Model Based FDI

The formulation of Model-Based FDI, as with adaptable estimation, relies on a dynamic model of the system. For this study, the algorithm was formulated to estimate the coulomb friction coefficient of the reaction wheel. The use case was monitoring the friction torque to allow for early detection of degrading performance induced by increasing friction. The reaction wheel model is represented by Eq 6,

$$J\dot{\omega} = T_{act} - (c)\,tanh(\omega) - \ sign(\omega)(v)\omega^{v_{exp}} \quad \textbf{(6)}$$

where J is the wheel moment of inertia, ω is the wheel angular rate, c is the Coulomb coefficient, v is the viscosity gain, v_{exp} the viscosity exponent. T_{act} is the wheel actuation torque. The model is discretised, and linearised, similar to the adaptable estimation problem to formulate a robust Extended Kalman Filter to estimate c from measurements of T_{act} and ω.

For the purpose of early detection of increasing friction, a regression scheme is applied to reveal trends in the parameter variation. Every t_{int} s, the parameter estimation is regressed onto a model with a constant component and a linear component. Batch least squares is then used to estimate the intercept and slope of the regression model. The variation of this slope is characterised for varying dynamics in the mission use case (EarthCARE) to set threshold that trigger fault detection.

There were three types of tests performed on the model-based FDI for the fault case of friction anomaly. The test cases and results are tabulated in Table 2-3 as a summary. The first two focus on anomaly detection and time. The third test case varies the model parameters c, v, v_{exp} of Eq. 6 to assess the sensitivity of the algorithm. Note that this variation of model parameters should only represent the uncertainty in the model, which are unknown at the algorithm´s implementation phase. Any known model discrepancy can be updated parametrically in the algorithm.

Table 2-3 Test result summary of model based FDI

1. Failure Case 1 Definition	
Manipulated parameter	Coulomb friction coefficient
Manipulation type	Bias
S/C dynamics	Steady State

Coefficient (Nm)	Detect, Isolate	Detection Time	Comments
5e-5	No detection	--	Very low friction torque fault case
5e-4	Detection and unit isolation	10 min	An order of magnitude less than nominal total friction torque
5e-3	Detection and unit isolation	3 min	In the range of the nominal total friction torque
5e-2	Detection and unit isolation	3 min	Larger than max allowed total friction torque for EarthCARE
2. Failure Case 2 Definition			
Manipulated parameter	Coulomb friction coefficient		
Manipulation type	Drift (random walk coefficient)		
S/C dynamics	Steady State		
Coefficient (Nm)	**Detect, Isolate**	**Detection Time**	**Comments**
5e-6	No detection	--	Negligible drift in friction torque
5e-5	Detection and unit isolation	10 min	Nominal Drift
5e-4	Detection and unit isolation	2.8 min	Aggressive drift (changes the order of magnitude of total friction)
5e-3	Detection and unit isolation	2.8 min	Very aggressive drift
3. Sensitivity to Parameter Variation			
Deviant parameter	**Nominal value**	**Acceptable deviation w/o triggering false faults**	
Coulomb friction coefficient	1e-5	10x	
Viscous friction gain	1e-5	3%	
Viscous friction exponent	1.5	<1%	

Exemplarily, the coulomb friction coefficient as estimated by the algorithm is shown for a no-fault case compared to a bias or drift induced within the simulation in Figure 2-2. It can be seen that the algorithm correctly estimates this deviation and triggers a fault detection. For information, the first jump seen in the coefficient friction is the initialisation of the Kalman Filter.

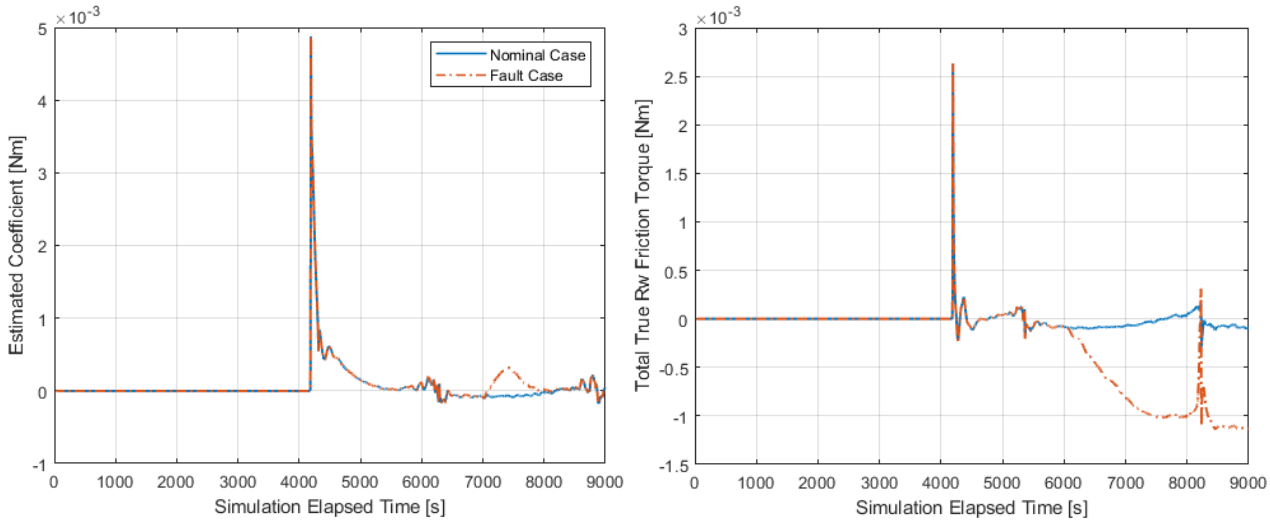

Figure 2-2 Model based estimated Coulomb coefficient for nominal case vs bias induced at 7000s (left); drift induced at 6000s (right)

For comparison for test cases 1 and 2 in Table 2-3 , in the standard EarthCARE mission implementation, friction monitoring is performed using the friction estimator whose primary function is to feed friction torque forward in the control loop. The fault detection threshold for monitoring is set to +-45mNm as maximum nominal friction limit. This mechanism does not trigger a failure detection for the fault parameters shown in Table 2-3. The FDIR in the EarthCARE mission triggers transition to ASM if the maximum threshold is surpassed, leading to loss of mission availability.

The model-based algorithm thus clearly shows promising results in early detection of the fault it targets, including detection of very low magnitudes of faults. The algorithm shows a clear advantage compared to the existing handling of friction wheel anomaly in EarthCARE, which, given the reliability of the equipment it uses, is quite sufficient to meet the mission requirements. However, for any mission relying on COTS hardware with increased sensitivity to faults, the algorithm provides an intelligent way of ensuring mission availability by identifying small deviations in friction behaviour early on prompting quick recovery without performance loss.

The driver in implementation of model-based FDI solution was characterisation of slope variation to quantify threshold for triggering faults. While a series of simulations was performed to characterise what would be “nominal” variation over the course of the mission, this characterisation will need an extensive campaign if the algorithm is to be flown. Another particular challenge of using model-based FDI on missions will be model integrity. It is known that the reaction wheel mechanical and thermal properties change over the course of years of use. This in turn will change the friction properties. If the model based FDI is to be used over the entire mission duration, it should ideally be based on a more involved model considering all possible variations. Otherwise, its operational use should be limited to a certain mission phase/duration.

2.3 Machine Learning-Based FDI

The application of deep learning in FDI is still limited. A large amount and wide variety of fault samples are needed to train a deep learning model with acceptable reliability and generalisation capability. It is often difficult to satisfy such requirement as faults occur infrequently. In addition, data obtained at different times or plant sites may follow different distributions due to parameter changes or performance degradation. Consequently, the diagnostic deep models will not be applicable until sufficient fault data covering all these changes become available. This limitation may be overcome

by using the data generated from computer simulations, which comes with the inevitability of model-plant mismatch. Transfer learning is a promising approach to address such mismatch.
The transfer-learning task can be accomplished by using one-class support vector machines (OCSVMs) to convert raw telemetry into an abstract representation that is effectively identical for both a spacecraft simulator and the flight unit [2]. An LSTM network designed to diagnose faults from this common representation can therefore train on data from the simulator and transfer its learning to operate on data from the real satellite.

Formulated for a single angle output (θ) of a star-tracker, the measurement model can be given as

$$\theta_i(t) = \theta^*(t) + n(t) \quad \text{for } i = 1,..N_s \tag{7}$$

Where $\theta^*(t)$ is the real attitude angle $n(t)$ is a Gaussian noise, and N_s is the number of star trackers. The process is summarised as follows:

- N_s independent linear combinations of the measurement signals are computed. If this step is not performed, the faults are not uniquely identifiable, and the problem is not well-posed. For example, since $\theta^*(t) \neq constant$, we cannot discriminate faults from manoeuvres.
- Dimensionality is reduced using a sliding window to process each signal into a cluster of lower-dimensional vectors. The simplest sliding window extracts the signal's mean μ and variance σ^2 from the window to form a cluster of data in (μ, σ^2) space. This reduces the sensitivity to noise and the input dimension of the OC-SVM obtaining a faster training procedure.
- Due to modelling errors and the variety of ways anomalies can manifest in telemetry signals, the simulated telemetry may not resemble the actual telemetry on orbit. To ensure that the algorithm is robust to these differences, the sliding window outputs are processed by a layer of OC-SVMs
- Finally, a Neural Network (NN) classifies the type of faults. Temporal context is critical because anomalies can represent temporary false positives or be related to each other across signals and across time. For this reason, a long short-term memory (LSTM) layer can model contextual temporal relationships. LSTM could outperform handcrafted fault detection and diagnosis rules because it is able to leverage patterns such as the intermittent versus persistent activation of the anomaly detectors that are complicated to capture with *if–then* rules. Furthermore, LSTM monitoring the evolution in time of the signals could detect trends rather than simply comparing the signals to given thresholds (which might delay detection).

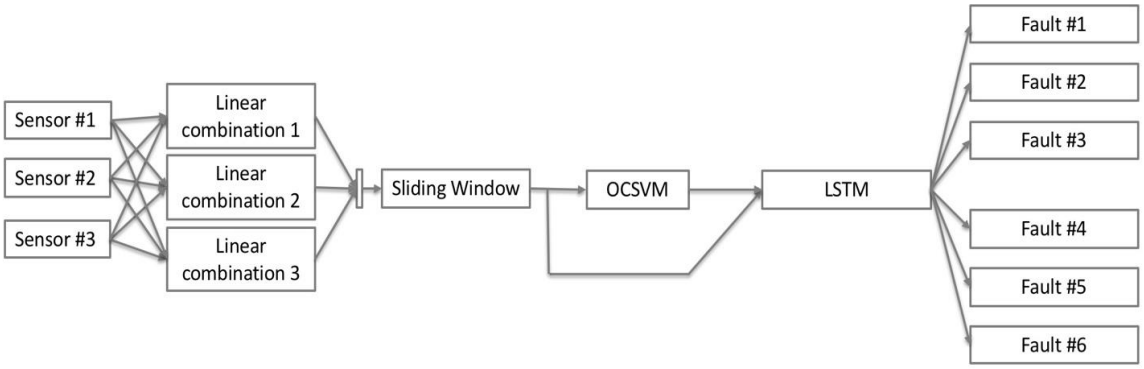

Figure 2-3 Proposed ML-based FDI architecture for a 3-equipment set

The OC-SVM was implemented in Matlab exploiting the *Statistics and Machine Learning Toolbox* and the LSTM NN based fault classifier was implemented in Matlab exploiting the *Deep Learning Toolbox* to perform sequence-to-one classification.

For implementation in this study on the OneSat use case to detect and isolate a degrading star-tracker, training data (single angle) was generated for 2 orthogonal star-trackers. To speed up the training process, coupling among different axes was not considered. Using the nominal performance (bias, NEA) data from the Sodern Auriga datasheet, random output values were generated using a simple Simulink model. The two-stage OC-SVM/LSTM network was trained thus:

- OC-SVMs were trained on simulated nominal data. 200 simulations of 500 s were performed for 20 values of ω (Ranging from 0 to 2°/s), where the signals are sampled at a rate of 10 Hz. Then, a sliding window with a window size of 1000 steps (100s) with a 50% of overlapping between adjacent windows was applied. 16000 sequences of 9 elements with 3 features (mean, variance and angular rate) were thus obtained.
- LSTM NNs were trained using both the simulated nominal data and performing 400 more simulations of 500 s for 20 values of ω. In 200 simulations, different values of the standard deviation were used ranging from 2 to 5 times the nominal value. In the other 200 simulations, different values of the bias term were used ranging from 10 to 15 times the corresponding standard deviation. Different values of faults are injected to avoid overfitting and make the classifier networks more generalizable.

The training of both OC-SVM and LSTM NNs was performed in a few iterations. To avoid polluting the training data with the test data, the GAFE simulation environment was not used for training purposes. However, to ensure consistency in magnitude with the output of the "real" (GAFE-simulated) star-tracker, training data generated purely on the basis of the data set had to be updated with an offset. This offset also partly accounted for the lack of intra-sensor alignment in the training data.

Finally, the results of the tests performed on ML-based FDI in the OneSat GAFE simulation are shown in Table 2-4, where $\sigma_n = 0.0178 \cdot 10^{-3}$ rad is the standard deviation of the base noise increase, and $c = 5.236 \cdot 10^{-6}$ rad is the coefficient of the base drift.

Table 2-4 Test result summary of machine-learning-based FDI

Failure Case Definition			
Manipulated parameter	STR Euler Angles		
Manipulation type	Noise, drift		
Simulation duration	6000s		
Failure type	**Fault Intensity**	**ML-Based Detection Time**	**Standard OneSat Detection time**
Noise std increase	σ_n	144.8 s	1236.0 s
Noise std increase	2 σ_n	79.8 s	1242.9 s
Noise std increase	3 σ_n	59.8 s	861.5 s
Noise std increase	10 σ_n	29.8 s	849.4 s
Drift	c	74.8 s	-
Drift	2 c	59.8 s	-
Drift	3 c	54.8 s	-
Drift	10 c	39.8 s	-

Note that SVM did not trigger any fault, i.e., the ML-based detection time is purely the detection time shown by the LSTM. This shows that the LSTM generalised much better than the SVM to new data, in that it detected faults at intensity levels at which the SVM does not trigger any faults. This may be partially due to the difference in the datasets used during training and during testing. The SVM thus appeared to be more sensitive to the dissimilarity between training and testing data.

The very low magnitudes of the faults that were tested for ML-based FDI showed that it was extremely sensitive to faulty behaviour and performed much better at early detection than the standard measurement processing and estimation algorithms. The detection times for the ML-based FDI are more than ten times faster for the LSTM than for the standard FDI mechanisms in OneSat. The ML-based FDI also was able to detect drift faults with very low magnitudes that do not trigger the standard FDI.

The absolute challenge in the implementation of ML-based algorithm in the study, and on any real space missions is ensuring training data representativeness of test/final application data. While discrepancies are addressed in theory by the transfer learning mechanism of the proposed algorithm, it was found that implementation for the tested mission use case required retraining the algorithm based on the final simulation environment and equipment model. For example, mismatch of the intra-sensor alignment between training and testing data could partly be solved by applying offsets resulting from frame transformations, but the noise characteristics do not transform easily and are not representative.

Future work could also look into applications where the current architecture SVM flagging and LSTM classification show clear benefits. It is recommended to apply this architecture to detection problems with high dimensionality (e.g., with gyro data, or direct use of raw telemetry data) and with classifying faults. While processing time for the algorithm was not assessed as part of this study, it is not estimated to be infeasible for on-board applications, if training is done pre-flight. This assessment should addressed in any future work on the architecture.

2.4 Adaptive Control

The adaptive controller investigated in this study is based on an L1 adaptive control architecture [3] designed to maintain (within a predefined frequency range) the dynamics of the satellite with its baseline (PD) controller in nominal operating conditions. The key feature of such controller architectures is guaranteed robustness in the presence of fast adaptation, which leads to uniform performance bounds both in transient and steady-state operation. These properties can be achieved by appropriate inclusion of a low-pass filter and by appropriate formulation of the control objective with the understanding that uncertainty in a feedback loop cannot be compensated outside of the control channel bandwidth.
The proposed architecture comprises a baseline controller and the L1 adaptive augmentation (Figure 2-4). The L_1 adaptive augmentation scheme can be used to compensate for uncertainties that were not addressed in the baseline controller design, such as actuator degradation and faults, severe external disturbances, and/or parameter uncertainties.

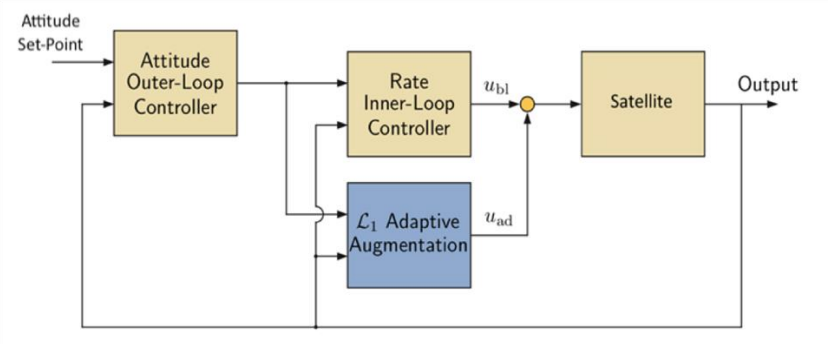

Figure 2-4 Scheme of the L1 adaptive augmented controller architecture [3]

The considered baseline controller is a cascading PP control architecture where the inner loop computes the torque u_{bl} needed to track the desired angular velocity ω_B, which is fed by the outer loop

using a purely proportional action (Figure 2-4). The control algorithm is given by the following expression:

$$\omega_B^o = K_p^o \, sgn(q_{4_e}) \, \rho_e \quad \textbf{(8)}$$

$$\tau_{BL} = K_p^i \, (\omega_B^o - \omega_B) + \omega \times J\omega \quad \textbf{(9)}$$

where $\rho_e \in \mathbb{R}^3$ and $q_{4_e} \in \mathbb{R}$ are, respectively, the vector and the scalar part of the quaternion error q_e, which is computed as the Hamiltonian product between the desired quaternion q_d and the conjugate of the measured quaternion q, *i.e.* , $q_e \coloneqq q_d \otimes q^*$. Furthermore, K_p^o and K_p^i are the proportional gain of the outer loop and of the inner loop respectively. The term $\omega \times J\omega$ is the feedback gyroscopic compensation.

Given the baseline controller, the objective of the L1 augmentation scheme is to design a state-feedback control signal to ensure that the body fixed angular rates track the closed-loop response given by the baseline controller in the nominal conditions. The L_1 adaptive augmentation law (blue block in Figure 2 4)includes a predictor and a filter to modify the baseline controller output with the adaptive signal u_{ad} in order to compensate the modelled uncertainties and disturbances within the bandwidth of the actuators.

The adaptive controller was implemented on the EarthCARE mission to account for degrading reaction wheel performance, simulated as a bias or drift on the wheel's output actuation torque. To isolate the effect of this algorithm, no other system characteristics were changed when compared to standard EarthCARE simulation with the nominal PD controller. The test parameters and results are shown in Table 2-5. The steady state APE for the adaptive controller compared to the nominal controller (keeping all other simulation parameters unchanged) is also shown in Table 2-5. The 3-axis error profile is exemplarily shown for one of the test cases (injection of 0.1Nm bias) in Figure 2-5 for the nominal controller and the adaptive controller.

Table 2-5 Test result summary of L1 adaptive controller compared to nominal mission-use-case controller

Failure Case 1 Definition			
Manipulated parameter		Actuation Torque	
Manipulation type		Bias	
S/C dynamics		Steady State	
Coefficient (Nm)	**Convergence**	**Steady State Error Adaptive Controller RPY [°]**	**Steady State Error Nominal PD Controller RPY [°]**
1e-2	Yes	1.205e-05, 9.568e-05, 0.427e-05	2.077e-05, 1.301e-05, 0.329e-06
3e-2	Yes	10.85e-05, 13.883e-05, 3.346e-05	22.997e-05, 1.421e-05, 2.859e-05
5e-2	Yes	28.733e-05, 21.28e-05, 7.306e-05	92.985e-05, 2.292e-05, 11.660e-05
1e-1	Yes	88.41e-05, 62.059e-05, 19.26e-05	677.57e-5, 13.749e-05, 78.726e-05
Failure Case 2 Definition			
Manipulated parameter		Actuation Torque	
Manipulation type		Drift	
S/C dynamics		Steady State	
Coefficient (Nm)	**Convergence**	**Steady State Error Adaptive Controller RPY [°]**	**Steady State Error Nominal PD Controller RPY [°]**

1e-4	Yes	0.719e-06, 86.921e-06, 0.5974e-06	1.581e-06, 14.534e-06, 0.654e-06
5e-4	Yes	5.331e-06, 88.879e-06, 2.1290e-06	6.228e-06, 14.508e-06, 1.150e-06
1e-3	Yes	20.245e-06, 93.295e-06, 7.659e-06	23.821e-05, 14.570e-06, 2.993e-06
2e-3	Yes	473.743e-06, 378.220e-06, 139.342e-06	132.805e-06, 16.016e-06, 15.244e-06
5e-3	Diverges 2000s	NA	NA

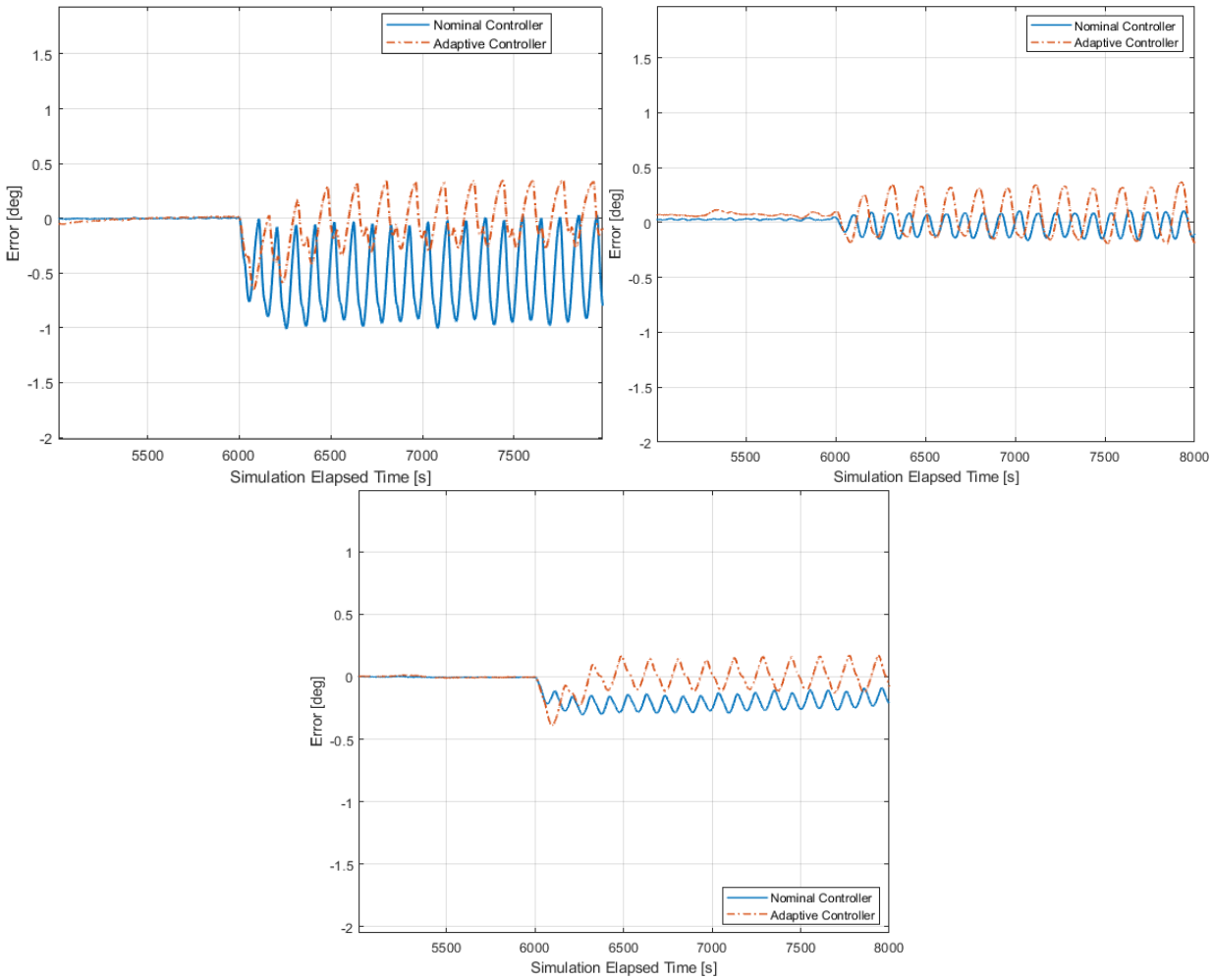

Figure 2-5 Error profile after 0.1Nm bias injection at 6000s for nominal and adaptive controllers in Roll (top left), pitch (top right) and yaw (bottom)

Considering all other factors are the same, the adaptive controller performs comparably to or better in rejecting fault when tested for a loss of reaction wheel performance (step and drifting) than the nominal controller. The performance of the nominal controller in the standard EarthCARE (all standard equipment) implementation is demonstrably better than the adaptive controller performance with COTS hardware. This might be attributed to the fact that the nominal controller in the EarthCARE simulation is in a mature development stage, and in a simulator simulating its mission characteristics more closely, it is robust with optimal performance.

The main challenge of the adaptive controller was attaining optimal tuning for the system that was beyond the scope of the study. Like any controller, an extensive tuning campaign will have to be performed and a full V&V for robustness undertaken to mature the controller design. A further nuisance to adaptive control performance is the control loop running frequency which might be limited in spacecraft applications.

2.5 Non-holonomic control

This algorithm was formulated for the case of providing partial three axis control on an under actuated spacecraft with only two reaction wheels. The selected method for this scenario is based on a time-varying control technique leveraging the transverse function approach [4] and directly developed on the set of attitude configurations (SO(3)), i.e., with no parametrisation. First, a given reference trajectory is classified as either feasible or unfeasible based on whether it adheres to or violates the momentum conservation principle. Notably, the control law ensures ultimately bounded tracking errors for any reference trajectory. In the case of feasible reference trajectories that meet specific persistence of excitation conditions, asymptotic tracking is attained by establishing an asymptotically stable zero dynamics for the closed-loop system. The algorithm requires, as inputs, the measured attitude matrix R and reference attitude trajectory $R_d(t)$, the measured angular velocity ω and the reference angular velocity $\omega_d(t)$ and knowledge of the inertia matrix. The unactuated spacecraft axis must be known as well, since the algorithm is formulated and tuned for a particular axis.

In summary, the approach relies on constructing, using the transfer function approach, a modified attitude error given by Eq 10

$$Z = R_e F^{\top}(\alpha) \tag{10}$$

where $R_e = R_d^{\top} R$ is the attitude error matrix; $F(\alpha)$ is an auxiliary rotation matrix (see equation 19 in [4]). Then, $Z = I_3$ is made exponentially stable using a pseudo input dependent on α. A dynamic time-varying law for $\alpha(t)$ is constructed to ensure $F(\alpha(t)) \to I_3$ when the reference trajectory $R_d(t)$ is persistently exciting, thus achieving $R_e \to I_3$. The attitude error can be made arbitrarily small (in the ideal setting) by properly tuning the control law at the expense of higher control effort.

The tests for this algorithm on the OneSAT use case were performed within a simplified Simulink environment. The simplified simulation isolates the control loop problem, including the spacecraft rigid body dynamics, actuators and sensors noises and linear limits, and the controller. Environmental effects are not considered in this simulation. The unactuated axis is the Z-axis assuming failure of the reaction wheel along this axis. The considered algorithm allows achieving satisfactory performance under the following assumptions and constraints:

- The three reaction wheels are oriented along the three spacecraft principal axes. In this specific case, the reaction wheels 1, 2 and 3 are oriented along the x, y and z-axis respectively.
- The total angular momentum (spacecraft and wheels) must be compatible with the desired attitude trajectory (feasible reference) or be sufficiently small in case of a constant reference attitude and non-feasible trajectories.
- The algorithm starts operating after the faulty reaction wheel has stopped spinning.

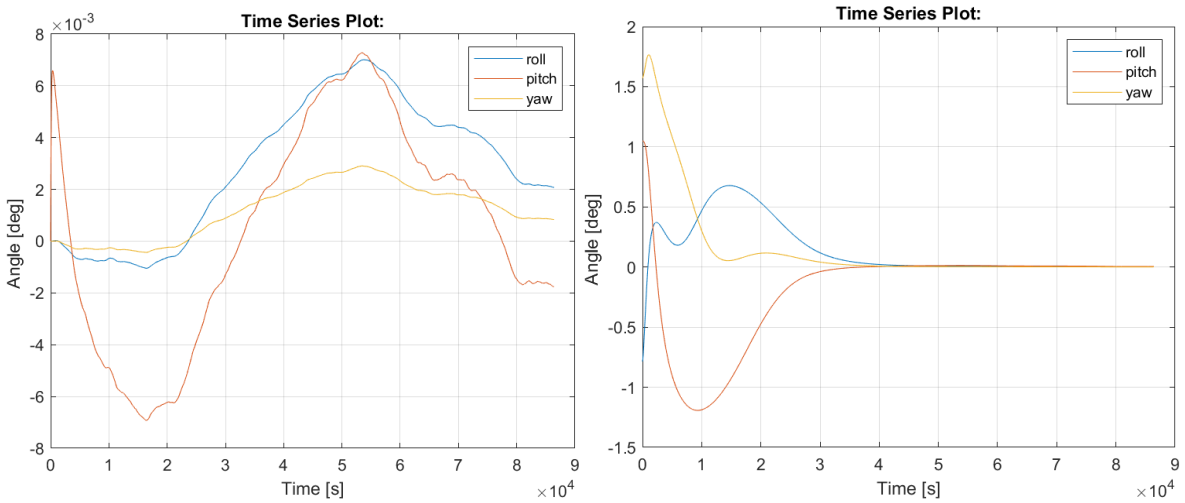

Figure 2-6 3 axis attitude error with non-holonomic control for nadir pointing profile (left); and set point attitude acquisition (right).

As seen in Figure 2-6, within this simplified simulation, the results show very high potential of the control strategy. The attitude control error is maintained within mrad range in the case of keeping the spacecraft in Nadir pointing attitude for one full orbit in GEO. Due to the need or zero initial angular momentum on the spacecraft for the algorithm, in its current formulation, it cannot be seen as a solution for temporary unavailability of a reaction wheel in a fully actuated spacecraft without loss of mission. A mode change is required to first bring the momentum to zero.

The challenge in the assessment of this algorithm was its employment in the full functional AOCS simulator. The operational constraints mentioned above (starting when the failed wheel has stopped running, having the actuators aligned with the body axes, avoiding loss of measurements due to blinding), lead to simplifications that were best simulated in a simplified environment. Additional work needs to be performed to apply the proposed technique in a more general and representative environment of space applications.

3 CONCLUSIONS AND RECOMMENDATIONS

Although the focus of this work was on COTS hardware, the general implications of the results produced here are applicable to all missions and hardware types. In case of COTS hardware with higher rates of fault, every benefit in detection and isolation leads to enhancement of favourable statistics for mission availability and potential performance improvement.

The FDI algorithms, adaptable estimation, model-based FDI, and ML-based FDI have shown better detection performance than standard threshold or inter-equipment check based methods. For the filter-based methods, the responsiveness relies greatly on the tuning of both the filter itself and the detection mechanism that triggers faults. Their robustness is a direct product of model fidelity for the filter-based methods, and of the training data representativeness for the ML-Based algorithm. The control algorithms have shown promising initial results. They will greatly benefit from an extensive tuning and robustness campaign to reveal optimum performance. Especially for adaptive control, its true advantage over nominal PID controller requires identification of use cases where a rightly tuned PID controller does not meet requirements. This may not be an equipment level fault, but a more complex sequence of events.

This paper forms a foundation of preliminary testing and comparison to standard methods for a variety of techniques. As further work, it is recommended to study each technique individually and to focus on its industrial use to increase its TRL, currently estimated at TRL 3.

4 ACKNOLEDGMENTS

The results presented in this paper have been achieved under funding of the ESA Contract. No. 4000133547/20/NL/CRS for the Technology Study "*AOCS COTS hardware fostering AOCS architecture and functions innovation*" (ADAPT), led by Airbus Defence and Space GmbH and in collaboration with Politecnico di Milano.

5 REFERENCES

1. Generic AOCS/GNC Techniques & Design Framework for FDIR (GAFE) http://gafe.estec.esa.int/simulator
2. J. R. Mansell, David A. Spencer, "Deep Learning Fault Diagnosis for Spacecraft Attitude Determination and Control", *Journal of Aerospace Information Systems*, 102-115, 2021
3. Magnus Bichlmeier, Florian Holzapfel, Enric Xargay and Naira Hovakimyan. "L1 Adaptive Augmentation of a Helicopter Baseline Controller," AIAA 2013-4855. *AIAA Guidance, Navigation, and Control (GNC) Conference.* August 2013.
4. H. Gui, G. Vukovich, & S. Xu, "Attitude tracking of a rigid spacecraft using two internal torques", *IEEE Transactions on Aerospace and Electronic Systems*, Vol 51, Iss. 4, 2015