On-Board Anomaly Detection on a Flight-Ready System

Andreas Koch*§, Alisa Krstova*, Florian Hegwein*, Mario Castro De Lera*, Filippo Ales*, Michael Petry*§, Rashid Ali†, Maen Mallah†, Laurent Hili‡, Max Ghiglione‡, Martin Werner§

*Airbus Defence and Space GmbH, †Fraunhofer Institute for Integrated Circuits IIS,
‡European Space Agency, §Technical University Munich

***Abstract*—Within the scope of an ESA funded activity, Airbus Defence and Space GmbH completed a research and development study in order to provide a novel dataset to ESA and develop a flight-ready system for on-board anomaly detection. This work includes the extraction of satellite telemetry data, the identification of anomalies, the development of machine learning models and the flight-ready system and finally the deployment of the machine learning algorithms via hardware acceleration. We present the benchmarking results of three accelerated ML algorithms from within the final flight-ready system.**

***Index Terms*—Machine learning, anomaly detection, FPGA, PUS, co-processor, on-board AI**

I. Introduction

Being operated mostly remotely, spacecrafts need to be highly reliable systems, which achieve the desired mission outcome despite potential failures. For this purpose, a Failure Detection, Isolation and Recovery (FDIR) strategy is employed, with the goal of fulfilling the specified mission target of reliability, availability, maintainability and operational autonomy. In order to notice non-nominal behavior, all identified feared events should be observable and necessary observations need to be included in the monitored telemetry. To this end, a key design parameter is the rate at which telemetries need to be recorded and reported. The most common approach to detect potential anomalies is to examine whether design limits for any instruments, modules and subsystems are crossed. These are referred to as out-of-limits (OOL) events and once such an event is recognized, the satellite safe mode is entered. The anomalies presented in this paper showed signs of an anomaly months before the OOL limits were crossed. Furthermore, with the rise of data-driven machine learning (ML) approaches to anomaly detection, an automatic observation system can be designed, which continuously processes telemetry to identify potential anomalies based on the telemetry behavior regardless of OOL limits. Besides detection, this can allow isolation of affected telemetry parameters, depending on the model in use. Accordingly, there have been several initiatives to explore the viability of ML-based anomaly detection, as well as other applications for ML in the space domain. In this paper, we present two machine learning models for anomaly detection, which have been integrated into a flight-ready system via hardware acceleration. Thereby, a design was chosen to include a module for anomaly detection, as well as a module for anomaly prognosis. The anomaly detection models are trained in an unsupervised manner to predict nominal telemetry, whereas the anomaly prognosis models are supervised models and directly predict whether telemetry is anomalous. For hardware acceleration, we opted to use specialized deep learning processors, which are the Xilinx Vitis AI Deep Learning Processor (DPU) and the Matlab Deep Learning Processor provided by the Deep Learning HDL toolbox. Both have been integrated into the FPGA part of the overall hardware image. For this reason, we opted to use a Xilinx Zynq Ultrascale+ MPSoC FPGA board and more specifically, this was the ZCU102 evaluation kit[1]. This paper is structured into a section on related work, a presentation of the novel anomaly dataset provided to ESA, a section on the ML algorithms and models, an overview of the flight-ready hardware and software system and finally a presentation of the results achieved by the ML models, both originally and when deployed on-board.

II. Related Work

One of the most rudimentary approaches to detecting point anomalies involves monitoring whether a predefined OOL threshold has been crossed. On actual satellites, this is implemented on a component basis, since the component design entails nominal operation within a certain range, e.g. a certain temperature range in case of temperature requirements. As these limits are provided by the manufacturer and are confirmed in the lab, OOL limits can be defined as critical for the spacecraft operation. For nominal telemetry, i.e. telemetry within the OOL limits, machine learning based approaches can offer further insight and preemptively detect anomalies. This might give the satellite operations team an early warning on a deteriorating situation and the opportunity to react in advance. In case of a validated approach, the satellite would also be able to react autonomously. Machine learning based approaches include decision trees [1] [2], support vector machines [3] [4] [5], autoencoders [6] [7] [8], recurrent neural networks [9] [10] [11], convolutional neural networks [12] [13] [14] and attention-based neural networks [15].

This activity has been funded by the ESA General Support Technology Programme.

[1] https://www.xilinx.com/products/boards-and-kits/ek-u1-zcu102-g.html

III. Dataset

Within the scope of the ADAP project, a new anomaly dataset covering three different anomalies in real satellite telemetry was used. Each contains nominal as well as anomalous segments and an overview is available at the end of this section.

A. Anomaly A: AOCS Sensor

An AOCS Pitch and Roll Sensor misbehavior, which is leading first to a subsystem and afterwards to a system critical anomaly with consequent transition to satellite SAFE Mode. This anomaly was observed on other satellites in similar units as the sensors were impacted by an external perturbation that was unknown until it appeared. The perturbation phenomena was created by very low temperatures on the upper atmospheric layers which don't allow the sensor to clearly detect the transition between Earth (hot side) and Deep Space (cold side). The perturbation phenomena was leading to an increased noise in the pitch and roll measurements and ultimately to an erroneous estimation of pitch and roll pointing.

B. Anomaly B: Payload Antenna Corona Discharge

Since the classical FDIR was not able to detect a unit malfunction, the payload antenna of the satellite was damaged, which had severe mission impact. The main problem for the detection of this event was that, although temperatures were indeed abruptly rising across the payload subsystem, these were still within the temperature limit ranges of the classical FDIR of all affected thermal lines. During nominal operations, the payload antenna is experiencing significant temperature variations depending on the sun illumination profile along the satellite‘s orbital position. However, the rising temperatures across payload components were very different from what is expected during nominal operations. Especially the combinations of affected thermal lines, as well as the very steep increase of the temperatures were very clear signs of a non-nominal payload condition. These ultimately culminated with a corona discharge event causing serious damage of the antenna.

C. Anomaly C: Solar Array Micro Meteoroid Degradation

This anomaly dataset contains manipulated data to simulate the impact of a micro meteoroid on the solar array of a satellite. As known from various missions, a micro meteoroid impact usually creates irreversible damage, causing a full or partial outage of one of its sections. During the satellite lifetime, the solar array currents show large variations due to many factors, such as the mission phase (e.g. the satellite orientation to the sun), the presence of earth or moon eclipses, the varying solar activities over the year and last but not least the degradation induced by the radiation. All these factors complicate the handling of a monitoring with the classical FDIR, especially as a power drop is just another variation which under some circumstances is actually expected.

TABLE I
Anomaly Dataset

Anomaly	Training set	Test set	Window size	Channels
A	500000	50000	60	13
B	148757	8578	100	49
C	68472	9972	72	6

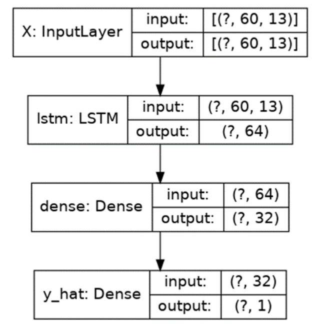

Fig. 1. 1-layer LSTM prognosis model with 22,081 parameters used for anomaly A.

IV. Algorithms

Machine learning models can be trained either in a supervised or an unsupervised fashion. For anomaly detection, this means either directly predicting whether telemetry is anomalous for supervised models, or predicting what the nominal telemetry values would be for unsupervised models and then comparing these to the actual telemetry. In line with [1] and others, supervised models need to be trained with a dataset that has an equal number of nominal and anomalous samples. This approach represents the anomaly prognosis module of the ADAP system and will be explored for a subset of 25000 nominal and anomalous samples each of anomaly A. For this purpose, a 1-layer LSTM model was trained, as depicted in Figure 1.

Since having class parity is quite a limitation in terms of scaling the dataset size and model complexity, all other ML models are trained in an unsupervised manner and predict nominal telemetry only. For anomalies A and C, an autoencoder model containing convolutional and pooling layers was chosen. Its architecture is displayed in Figure 2. The model was implemented with hardware acceleration in mind and for this reason it only incorporates layers, which are supported by the Xilinx Vitis AI deep learning processing unit (DPU). For anomaly B, a 2-layer LSTM model was selected in order to forecast the next sample based on a telemetry data interval. Its architecture is visualized in Figure 3. It was implemented in hardware based on the processor provided by the Matlab Deep Learning HDL toolbox.

V. Hardware System

The goal of the ADAP project was to provide satellites with a self-standing co-processor, able to autonomously process telemetries and detect potential anomalies. This is achieved by presenting an interface to the main processor, which is

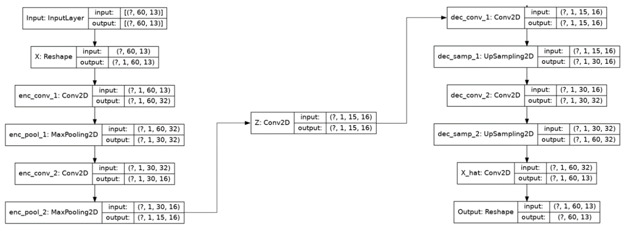

Fig. 2. Autoencoder anomaly detection model containing convolutional and pooling layers. Only layers supported by the Xilinx Vitis AI deep learning processing unit (DPU) are used. A small version of this model with 11,293 parameters was used for anomaly A, and a version with 62,145 parameters was used for anomaly C.

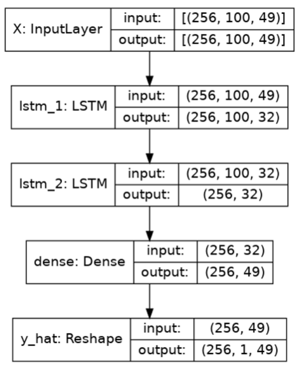

Fig. 3. 2-layer LSTM forecast model with 21,466 parameters.

based on PUS packets. In doing so, the main processor is able to control the ADAP system's high level functions, such as booting, and initially supply all parameters required by the machine learning algorithms. In this setting, the supervisor runs on a real-time Cortex-R5F processor and interfaces to the outside processor via PUS commands. It receives new telemetries, writes them to an intermediate buffer memory and notifies the corresponding ML application running on a Cortex-A53 processor. Once the computation completes, the predictions and potential anomalies are returned to the supervisor and in case of an anomaly, they are reported back to the outside processor. Although the supervisor acts as a co-processor, it supports full functionality of an on-board computer (OBC) and is able to run independently as is the case in the experimental setup.

Furthermore, the ADAP system manages ML applications running on the Cortex-A cores from a Xen Hypervisor. It allows encapsulation of full operating systems, offering a way to configure respective interfaces and manage memory access rights. Overall, this leads to a more stable system design [16] and it allows running different OS in parallel, such as Petalinux and FreeRTOS. Due to the Matlab DLP requiring access to DDR memory space for storing weights and performing calculations and Petalinux likely not accounting for this need to access memory space, interfacing to the DLP was unsuccessful from a Petalinux OS. On the other hand, interaction was very much possible from FreeRTOS. For this reason, we interfaced with the Matlab DLP via a C application writing to its registers. This was achieved from FreeRTOS being run as a guest application in the overall Xen Hypervisor system. By contrast, interaction with the Vitis AI processor is possible from Petalinux only. As a result, we run both Petalinux and FreeRTOS in parallel in the final ADAP system. The Ultrascale+ board's FPGA is programmed to encompass both ML processors, reaching a total resource utilization as specified in Table II.

TABLE II
FPGA RESOURCE UTILIZATION

Anomalies	Processor	DSPs	BlockRAM	LUTs
A & C	Vitis AI	704	261	58592
B	DLP	73	50	58475
A, B & C	Total	777 / 2520 (30.8%)	311 / 912 (34.1%)	117067 / 274080 (42.7%)

TABLE III
ML PERFORMANCE FOR ANOMALY DETECTION MODELS

Anomaly	Model	Accuracy	Precision	Recall	Specificity
A	Prognosis	0.483	0.392	0.063	0.902
A	Detection	0.825	0.881	0.752	0.899
A	Quantized	0.589	0.568	0.736	0.442
A	QAT	0.661	0.664	0.647	0.674
B	Original	0.923	0.912	0.919	0.927
B	DLP	0.81	0.725	0.933	0.708
C	Original	0.999	0.999	0.999	0.999
C	Quantized	0.999	0.999	0.999	0.999

TABLE IV
ON-BOARD PERFORMANCE FOR ANOMALY DETECTION MODELS

Anomaly	Model	Throughput [MB/s]	Peak [W]	Delta [W]	Energy [μJ/Bit]
A	QAT	1.431	5.040	0.077	0.430
B	DLP	0.505	5.221	0.007	1.325
C	QAT	6.929	5.283	0.324	0.090

VI. RESULTS

Since the overall system was described in the previous section, only the results of the AI algorithms on-board implementation will be presented in the following. To judge the performance of any on-board ML algorithm, it is important to not only compute the typical ML metrics, such as accuracy, but to measure the power consumption as well as throughput. To increase both power consumption and throughput, integer quantization was performed for the models utilizing the Xilinx Vitis AI framework.

For anomaly A, the prognosis model showed overfiting, since the validation accuracy was significantly higher compared to the test set accuracy. Consequently, a less complex model or regularization could lead to better performance. The detection model on the other hand exhibits reasonable performance and it will be used for hardware acceleration. As can be seen in Table III, quantization leads to a loss in accuracy,

although it is possible to recover this loss via quantization aware training (QAT). The 2-layer LSTM model trained on data for anomaly B retained its performance of an accuracy of 92.3% after it was imported in Matlab, although there was a drop-off in performance when executing it with the Matlab DLP accelerator from within our own hardware image. This drop-off was only observed in our own image and might be caused by memory interference of another hardware block.
Lastly, the bigger version of the autoencoder model depicted in 2 retained all of its accuracy after quantization. Therefore, quantization aware training was not required.

Table IV exhibits the on-board performance of the three hardware accelerated ML models. To put the throughput results into context, it is important to note that the shape of a sample is different for every anomaly, as listed in Table I. For this reason, throughput is presented in megabytes per second. With the model for anomaly C being five times the size of the model for anomaly A, a significantly higher throughput is quite surprising. This might be due to the DPU being able to leverage its acceleration capabilities better with an increased size of parameters. In comparison, the DLP has a smaller overall throughput, although it was only evaluated in a configuration with a batch size of 1. Increasing the batch size might yield a higher throughput. Besides, the Matlab Profiler reports a throughput, which is 2.5 times the throughput achieved in our system. Delta refers to the difference in average power measured before and during the computation. Finally, energy is calculated based on the average power measured during the computation. This is done to judge the cost of processing telemetry data within the full flight-ready system. Considering the differences in throughput for anomalies A and C, it is reasonable for the larger model of anomaly C to consume significantly less µJ per Bit.
In these experiments, power measurement was carried out with the shunt resistors of the ZCU102 evaluation board, which are connected to TI INA226 power monitors. These allow measurement of both current and voltage. The final power measurement application closely followed this example application provided by Xilinx[2].

VII. Conclusion

In this work, we presented a machine learning based on-board anomaly detection system, able to detect three distinct anomalies. Corresponding machine learning models have been integrated into a flight-ready system via hardware acceleration. For this purpose, we measured the performance of our hardware accelerated models achieved by utilizing the Vitis AI framework and Matlab Deep Learning HDL toolbox respectively. Contributions of this work are a novel anomaly dataset, which has been provided to ESA, as well as a novel realization of ML-based on-board anomaly detection for application in the space domain.

[2]https://www.xilinx.com/developer/articles/accurate-design-power-measurement.html

References

[1] K. Naik, A. Holmgren, and J. Kenworthy, "Using machine learning to automatically detect anomalous spacecraft behavior from telemetry data," in *2020 IEEE Aerospace Conference*. Big Sky, MT, USA: IEEE, 2020, p. 1–14. [Online]. Available: https://ieeexplore.ieee.org/document/9172726/

[2] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation-based anomaly detection," *ACM Transactions on Knowledge Discovery from Data*, vol. 6, no. 1, p. 1–39, 2012. [Online]. Available: https://dl.acm.org/doi/10.1145/2133360.2133363

[3] M. Amer, M. Goldstein, and S. Abdennadher, "Enhancing one-class support vector machines for unsupervised anomaly detection," in *Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description - ODD '13*. Chicago, Illinois: ACM Press, 2013, p. 8–15. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2500853.2500857

[4] J. He, Z. Cheng, and B. Guo, "Anomaly detection in satellite telemetry data using a sparse feature-based method," *Sensors*, vol. 22, no. 17, p. 6358, Aug 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/17/6358

[5] B. Liu, Y. Xiao, L. Cao, Z. Hao, and F. Deng, "Svdd-based outlier detection on uncertain data," *Knowledge and Information Systems*, vol. 34, no. 3, p. 597–618, 2013. [Online]. Available: http://link.springer.com/10.1007/s10115-012-0484-y

[6] T. Amarbayasgalan, B. Jargalsaikhan, and K. Ryu, "Unsupervised novelty detection using deep autoencoders with density based clustering," *Applied Sciences*, vol. 8, Aug 2018.

[7] R. Boumghar, A. Venkateswaran, H. Brown, and X. Crespo Álvarez, *Behaviour-based anomaly detection in spacecraft using deep learning*, Aug 2021.

[8] K. Yang, Y. Wang, X. Han, Y. Cheng, L. Guo, and J. Gong, "Unsupervised anomaly detection for time series data of spacecraft using multi-task learning," *Applied Sciences*, vol. 12, no. 13, p. 6296, Jun 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/13/6296

[9] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom, "Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, Jul 2018, p. 387–395, arXiv:1802.04431 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1802.04431

[10] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "Lstm-based encoder-decoder for multi-sensor anomaly detection," no. arXiv:1607.00148, Jul 2016, arXiv:1607.00148 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1607.00148

[11] G. Xiang and R. Lin, "Robust anomaly detection for multivariate data of spacecraft through recurrent neural networks and extreme value theory," *IEEE Access*, vol. PP, p. 1–1, Dec 2021.

[12] M. Munir, S. A. Siddiqui, A. Dengel, and S. Ahmed, "Deepant: A deep learning approach for unsupervised anomaly detection in time series," *IEEE Access*, vol. 7, p. 1991–2005, 2019, conference Name: IEEE Access.

[13] Z. Zeng, J. Lei, G. Jin, C. Xu, and L. Zhang, "Detecting anomalies in satellite telemetry data based on causal multivariate temporal convolutional network," p. 12, Jul 2021. [Online]. Available: https://doi.org/10.1145/3449726.3459411

[14] H. Zhao, M. Liu, and T. Luo, "Outlier removal of discontinuous satellite telemetry data based on deconvolutional reconstruction network," p. 5, Aug 2022. [Online]. Available: https://doi.org/10.1109/DDCLS55054.2022.9858392

[15] S. Gundawar, N. Kumar, P. Yash, A. K. Singh, M. Deepan, R. Subramani, B. R. Uma, G. Krishnapriya, B. Shivaprakash, and D. Venkataramana, *Multihead Self-attention and LSTM for Spacecraft Telemetry Anomaly Detection*. Cham: Springer International Publishing, 2022, vol. 1528, p. 463–479, series Title: Communications in Computer and Information Science. [Online]. Available: https://link.springer.com/10.1007/978-3-030-95502-1_35

[16] R. C. Amorim, R. Martins, P. Harikrishnan, M. Ghiglione, and T. Helfers, "Dependable mpsoc framework for mixed criticality applications," Jun 2021. [Online]. Available: https://zenodo.org/record/5521521