Behaviour-based anomaly detection in spacecraft using deep learning

Redouane Boumghar[1*], **Adithya Venkataswaran**[2], **Hugh Brown**[1], **Xabier Crespo Álvarez**[1] and **contributors**

[1]Libre Space Foundation

[2]Indian Institute of Space Science and Technology

red@libre.space, adithya.aiml@gmail.com, aardvark@libre.space, xabi@libre.space

Abstract

Spacecraft operators carefully review the engineering design of their satellites in order to select the most important telemetry parameters to supervise during operations. They set out-of-limits (OOL) warnings based on manufacturers guidelines, hardware manuals and system tests as well as their own expertise and mission conditions. The limits are set on single parameters or on synthetic parameters (functional composition of several parameters) and represent the automation of operators expertise. From the assumption that, most of the time, a spacecraft operates in good conditions, our approach aims at detecting breakpoints in the entire spacecraft behaviour to extract a sequence of multiple behaviours. Unusual changes of behaviour, detected by a probabilistic approach, are then reported as anomalies. This approach does not require prior knowledge of what anomalies may look like and is complementary to operators' OOL approach. We can extract a list of telemetry parameters that are the largest contributors of the difference in an unusual change between two consecutive behaviours by analyzing the changes in the dependency graphs generated before and after the change.

1 Spacecraft behaviour extraction

Spacecrafts are complex dynamic systems that are carefully designed, assembled, operated and monitored. However, the search for the cause of most anomalous events is non-trivial [Schlag *et al.*, 2018]. OOL checks come with the drawback of needing to be defined manually. And they often offer coarse detection of subtle changes in the telemetry that might be precursor to future anomalies.

The Polaris project, supported by the Libre Space Foundation [Foundation, 2021b], offers an open source machine learning tool suite to analyze spacecraft telemetry data, including related external data sources, for a deeper understanding of the varied internal subsystems interactions and the impact of the environmental situations and events in space.

*Contact Author

A project component, called BETSI (Behaviour Extraction for Time-Series Investigation) [Foundation, 2021a], uses deep learning to find the breakpoints between the different spacecraft behaviours, then detect which behaviour changes are most abnormal. To detect breakpoints, BETSI uses a timeseries segmentation approach [Lee *et al.*, 2018] based on an auto-encoder neural network architecture. Unlike [Hundman *et al.*, 2018], all input telemetry is given as input to the neural network. The rationale is that a behavior is defined by the respective status of all telemetry, and cannot be extracted by single telemetry analysis. The auto-encoder thus builds features that reside in the central layer for each consecutive chunk of time.

The automatically learned features of each time period are then compared using the L2 norm based distance. As they are the result of a dimensionality reduction, they can be considered as input signatures sensitive to input behaviour changes.

Three input parameters are required to be set in BETSI:

- Time window size: time period on which features will be computed. If it's too narrow, not enough information can be gathered; if it's too wide, the differences between consecutive windows' features risk not being consequent enough;
- Stride: the time space between the start of two time windows. Combined with the time window size, it determines the overlap between two consecutive windows defining how much information is shared;
- A threshold to extract the most important distances differences that would highlight the most important breakpoints. This threshold is defined as a percentage over the average value making it easily adaptable to all telemetry. A high value would mean that fewer events will be detected and vice-versa.

2 Case study: Characterizing behaviour changes for BOBCAT-1 cubesat mission

The cubesat mission BOBCAT-1 is a 3U cubesat built by the Russ College (Ohio University, USA) and aimed at studying the performance of GNSS navigation systems of other satellites. On 2021-04-25 the mission operators reported suffering from absence of de-tumbling; therefore, the Polaris project team decided to run BETSI from 2021-04-01 to 2021-05-05, a period enclosing that event.

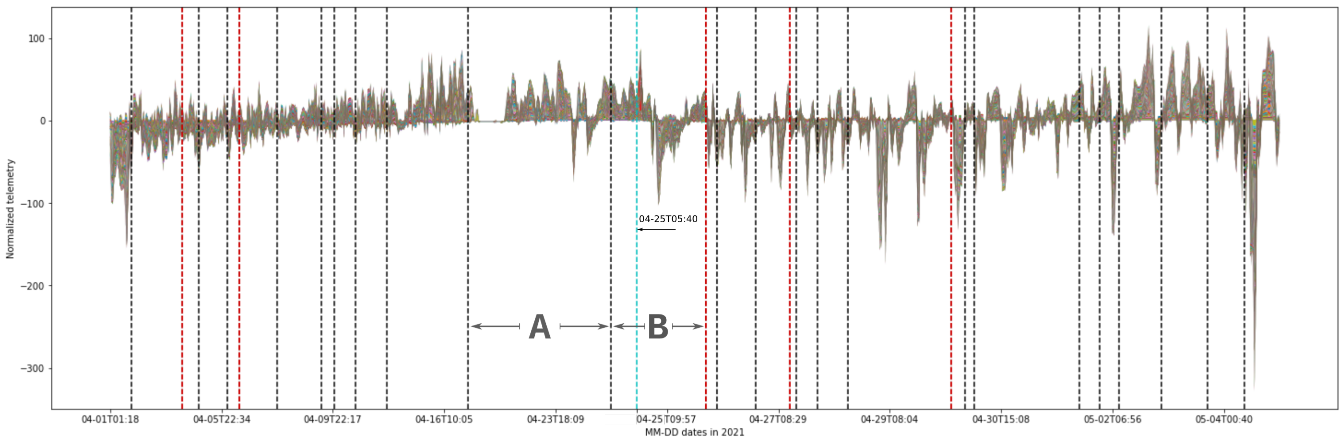


Figure 1: Stacked normalized telemetry view of BOBCAT-1. Parameters: time windows of 25 minutes, stride of 10 minutes and distance threshold of 50% to trigger breakpoints. Black and red dashed bars are BETSI breakpoints; the red dashed bars are the strongest behaviour changes. The cyan dashed bar is the operators' feedback on the critical event where spacecraft entered safe mode and de-tumbling was deactivated. Periods A and B mark periods around the closest breakpoint to the the reported anomaly.

Figure 1 shows all detected events in that period. Some events are close together while others are separated. The idea is that, if consecutive time windows belong to the same behaviour, the model would be able to reproduce its input, window after window, with very similar feature vectors. If a prominent change in the central feature vector is observed, a break point is generated.

Breakpoints are marked by vertical bars in figure 1; they are positioned with a precision depending on the time window size and stride (spaces between time windows). As the bars are placed at the start of each time window, the real breakpoint (if there is one) might happen anywhere later in that time window (in our case in 25 minutes).

A behaviour segment is a space between two breakpoints. In the case of the anomaly reported by the operators at 2021-04-25 05:40:00 (in cyan in figure 1), the closest BETSI breakpoint precedes that anomaly at 2021-04-24 15:38:14 marks a dividing line between two divergent behaviours (notes as A and B in figure 1. Here, we turn to Polaris' dependency graph tools to continue analysis by characterizing behaviour before and after this breakpoint. In order to characterize what happened around this breakpoint we analyze the behaviours space A and B using the Polaris dependency graph tool.

Figures 2a and 2b show the dependency graph calculated on telemetry data belonging to period A and B, respectively. Dependencies between telemetry parameters, are defined by extracting the feature importance of the extreme gradient boosting machine learning technique (XGBoost) as described in [Ceglarek and Boumghar, 2021]. The graph based visualization technique is inspired from previous work on enhancing situation awareness [Boumghar *et al.*, 2018].

The structures of the two graphs, calculated on periods A and B, are strikingly different. After the breakpoint (period B), temperature parameters (in green) tend to have a much more organized or hierarchical influence on each other while before the breakpoint (period A), they were all influencing each other in a disorganized manner.

3 Graph Analysis

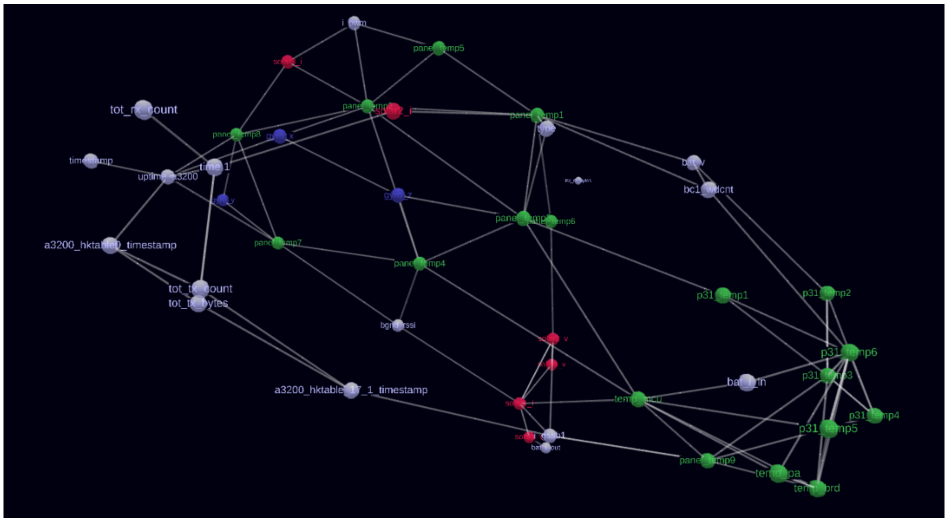
The clear structural differences between the two dependency graphs calculated on period A and B can help operators bringing insightful hypothesis of what changes are remarkable during this critical change. We note that not all breakpoints are anomalous; a graph comparison based on data before and after each breakpoint can help categorizing breakpoints as nominal or abnormal changes.

In our case, a more detailed analysis of the difference between the two graphs shown in figures 2a and 2b highlights the following changes:

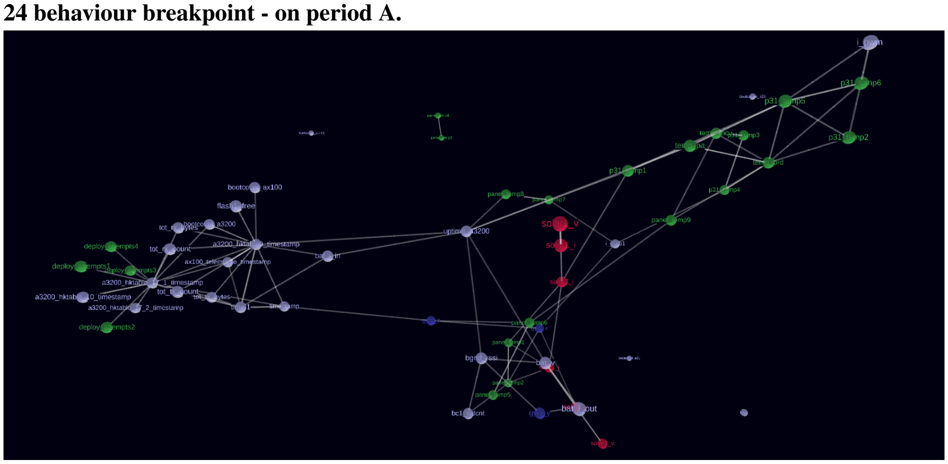
- 43 nodes connections have consequently changed values with 21 of them being a reinforcement of their mutual influence
- 41 new pairs have been connected from telemetry parameters (nodes) already active in the graph
- 12 new connections have been established from 4 newly active telemetry parameters

The term *active telemetry* refers to a graph activity, in opposition to the real activity of the parameter. A telemetry is considered active in the graph when it has influence on others. This influence is calculated with respect to all other influences, so one telemetry might be considered inactive because others are much more influential.

Thanks to semantics attached to each node we can extract information to interpret what is happening during a behaviour change. In our case, 19 of consequent changes and 22 of new pairs seem to be directly related to temperature parameters, proving 48% of the behaviour change is expressed in temperature relations inside the spacecraft. The temperature measurements that seems to have been impacted are from the panels, board and processing unit. Temperatures effects are greatly driven by orbit conditions, illumination conditions, but also by how electronic components are involved in different satellite tasks.



(a) 3D visualization of the inter-influences of telemetry parameters in BOBCAT-1 **before 2021-04-24 behaviour breakpoint - on period A.**



(b) 3D visualization of the inter-influences of telemetry parameters in BOBCAT-1 **after 2021-04-24 behaviour breakpoint - on period B.**

Figure 2: 3D visualization of inter-influences during period A and B. The different colors of nodes indicate the kind of telemetry they represent: green for temperatures, red for solar panels voltage and intensity, dark blue for gyrometers.

The graphs have been generated from two time periods of about 7.8 days and 4.5 days respectively; this means many orbits were completed during these periods. This reinforces the fact that the observed differences between the two graphs are due to a real change in the operational mode of the satellite instead of impact from external factors.

The other 51% of changes in the graphs show presence of influence from the acceleration measurement, along with, but not directly related to, changes between solar panels activity and battery voltage and current. With the information that the satellite started tumbling, we can hypothesise that changes in rotation (angular acceleration) might have impacted the way solar panels were illuminated, thus being a precursor behaviour for anomalies.

4 Conclusions

BETSI has demonstrated that insights are to be gained from the examination of time series telemetry via deep learning. Through our analysis of BOBCAT-1 telemetry, we have demonstrated that automated detection of behavioural changes is possible, and matches well with satellite operators' manual approaches to identifying and diagnosing anomalies.

In its current form, BETSI can identify these breakpoints and provide them to operators in a form suitable either for display using standard libraries, or as input for future analysis. The goal is to be able to provide meaningful reports for data scientists or analysts who do not have an operations background. BETSI, licensed under the LGPLv3, can also serve as a core library for other applications.

4.1 Future work and vision

While BETSI serves as a strong foundation for detecting behavioural changes, we wish to integrate it further into Polaris in order to complement existing Polaris tools and to expand its capabilities. To that end, our next milestone will be two-fold: easy analysis of data with BETSI by invoking it from the Polaris command line; and interactive, automated display of time series data with identified breakpoints clearly marked, helping operators to make the most of this deep learning tool.

Future milestones include:

- improving our graph analysis by introducing the time factor. This will help visualize the dynamic changes in the graph in real or accelerated time;
- studying the sequencing of behaviours to detect if some behaviour changes are precursor to anomaly events.

The fact that not all breakpoints are marked by operators does not exclude the fact that they could mark anomalous behaviour changes. The novelty of our approach resides in the fact that we can learn from every breakpoint by analysing the dependency graphs of each surrounding period, as well as studying the sequencing of behaviours. This sequencing would help operators understand which behaviour normally or usually follow up another one, thus building a real knowledge base of how the whole spacecraft behave.

Additionally, we are currently investigating the feasibility of using the model built by BETSI in resource-constrained environments for real-time detection and flagging of behavioural changes. Our goal is to demonstrate that even modest cubesat hardware can successfully run state-of-the-art machine learning models, supplying operators with automated insight into changes of behaviour that need attention and paving the way for in-orbit, spacecraft-initiated anomaly corrections.

References

[Boumghar *et al.*, 2018] Redouane Boumghar, José da Silva, Ioannis Angelis, Jonathan Schulster, and Alessandro Donati. Enhanced awareness in space operations using multipurpose dynamic network analysis. In *2018 SpaceOps Conference*. American Institute of Aeronautics and Astronautics, may 2018.

[Ceglarek and Boumghar, 2021] Jan-Peter Ceglarek and Redouane Boumghar. Polaris: A machine learning tool for telemetry data exploration. *2021 SpaceOps Conference*, 2021.

[Foundation, 2021a] Libre Space Foundation. BETSI component for the Polaris Project. https://gitlab.com/librespacefoundation/polaris/betsi, 2021. [Online; accessed 13-August-2021].

[Foundation, 2021b] Libre Space Foundation. Polaris Project. https://libre.space/projects/polaris/, 2021. [Online; accessed 13-August-2021].

[Hundman *et al.*, 2018] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Söderström. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding. *CoRR*, abs/1802.04431, 2018.

[Lee *et al.*, 2018] Wei-Han Lee, Jorge Ortiz, Bongjun Ko, and Ruby Lee. Time series segmentation through automatic feature learning. *arXiv preprint arXiv:1801.05394*, 2018.

[Schlag *et al.*, 2018] Leonard Schlag, Corey OMeara, and Martin Wickler. Numerical analysis of automated anomaly detection algorithms for satellite telemetry. In *2018 SpaceOps Conference*, 2018.