

# State of Polaris, more visualization in using ML for space operations

---



Adithya Venkateswaran

December 10, 2021

The Polaris Project

Libre Space Foundation

Open Source CubeSat Workshop 2021

# INTRODUCTION

---

## ABOUT US (POLARIS)

---

- Automatic, machine learning based open source tools to help operators monitor the health of their satellites.
- Global team from Canada to India.
- Supported by and part of the Libre Space Foundation.
- Started as an idea at Open Source CubeSat Workshop 2018.

# OPEN SOURCE TOOLS WE USE!

---

We depend on several open source projects for delivering Polaris:


- XGBoost - Finding dependencies between telemetry parameters
- Tensorflow - Extracting behaviour for detecting anomalies
- ReactJS, ThreeJS - Visualizing our results
- SatNOGS, NOAA/SWPC - Our major data source
- poliastro - For propagating orbits

# WHAT WE PRESENTED LAST YEAR

- Polaris generated dependency graph and what it means
- Collaboration with the BOBCAT-1 team
- Watch it at <https://youtu.be/SxYeLhbY21U>

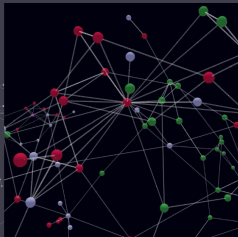
## The Force of the Data

A 3D Graph for Dependencies




**Polaris**  
RESEARCH LEARNING NETWORK

Item	Meaning
Nodes	Telemetry/feature
Links	Dependency
Dots	Degree of dependency
Number of connections	Level of significance
Color	Group (manual)



LightSail-2 Dependency Graph

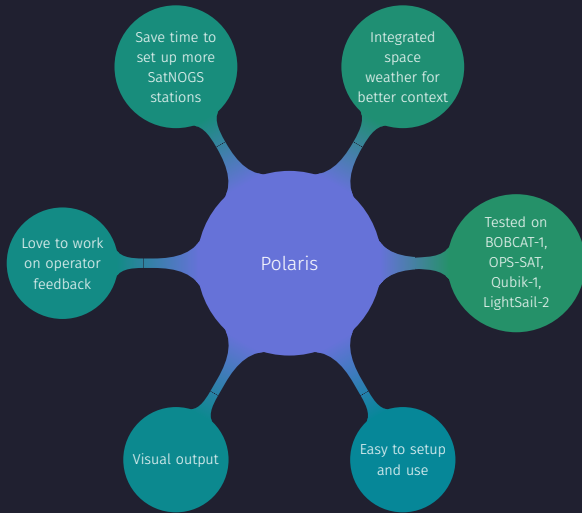


OSCW 2020

WHY POLARIS?

---

# WHY USE POLARIS:



## SIMPLE COMMAND FOR ROBUST VISUALIZATION

### ALL OF POLARIS IN FEW COMMANDS!

- `polaris fetch [SAT] normalized_frames.json`
- `polaris learn normalized_frames.json graph.json`
- `polaris viz graph.json`

(OR)

- `polaris behave normalized_frames.json anomaly.json`
- `polaris report anomaly.json`



# ANOMALY DETECTION VISUALIZATION

---

# ANOMALY DETECTION

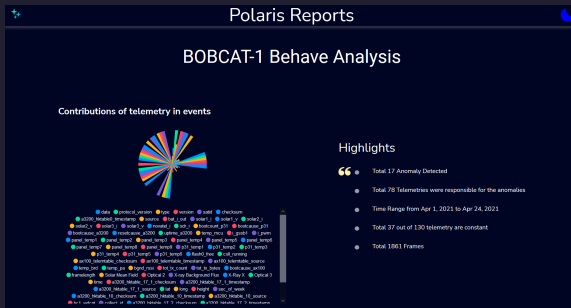
---

- Data is ordered by time and grouped into "windows".
- A neural network model (auto-encoder) tries to reproduce these "windows" of data.
- The internal representation of data are compared to find maximum deviation (behavioural breakpoints).

# VISUALIZATION OF ANOMALY REPORTS I

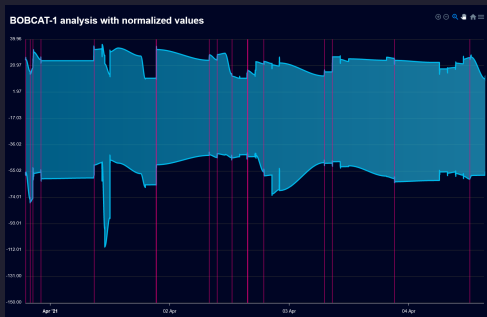
The output of *polaris report* is an interactive webpage:

Figure 1: Main screen showing statistics, with toggle for light and dark mode



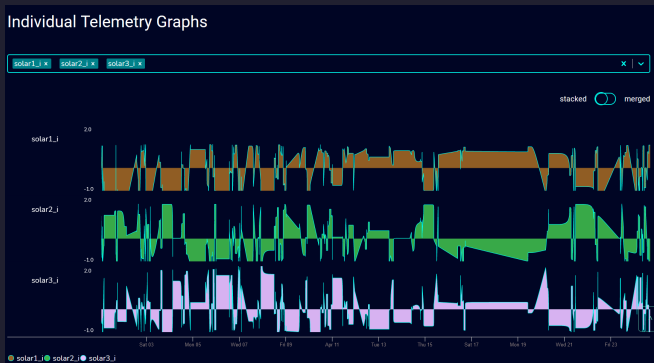
## VISUALIZATION OF ANOMALY REPORTS II

**Figure 2:** Stacked normalized values of telemetry. Red lines are detected breakpoints. Bold red line is the largest behavioural change detected.



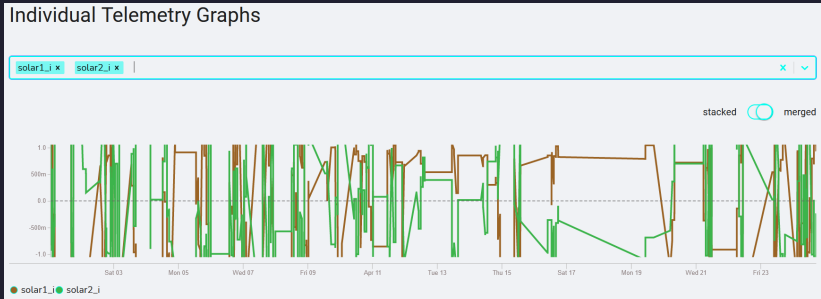
# VISUALIZATION OF ANOMALY REPORTS III

Figure 3: Visualize any field in your telemetry



# VISUALIZATION OF ANOMALY REPORTS IV

Figure 4: Merge the plots to get a single one (and light mode)



# CONCLUSION

---

# WHAT WE WANT FROM YOU!

## TEST US OUT

- Install using `pip install polaris-ml`
- Check out our code, demo and documentation at `polarisml.space`
- Join us at `app.element.io/#/room/#polaris:matrix.org`

We want your feedback on:

- features you want implemented
- Polaris has helped your team
- what didn't work so well

so that Polaris becomes an integral part in managing your satellites!



# THANK YOU

To all the contributors to Polaris.

- Xabi Crespo
- Hugh Brown
- Red Boumghar
- Jan-Peter Ceglarek
- Julien Flawinne
- Ayush Bansal
- and many more

To the Open Source CubeSat Workshop team and Libre Space Foundation.

To the audience and all satellite operators who have helped us get so far.