Paper ID: 43360
oral

25th IAA SYMPOSIUM ON SMALL SATELLITE MISSIONS (B4)
Small Satellite Operations (3)

Author: Mr. Artur Scholz
National Cheng Kung University, Germany, artur.scholz@librecube.net

Prof. Jer-Nan Juang
National Cheng Kung University, Taiwan, China, jjuang@mail.ncku.edu.tw

AN OPEN-SOURCE, PYTHON-POWERED WEB FRAMEWORK TO SUPPORT SMALL SATELLITE MISSION OPERATIONS

Abstract

In recent years, constellations of dozens and even hundreds of small satellites are being proposed and implemented. Factors of this development are the low costs in the production of small satellites (in particular CubeSats) and the interest to pursue more complex mission scenarios.

This means that there is a shift from single-groundstation, single-spacecraft to multi-groundstation, multi-satellite architectures, resulting in optimization challenges that are well known to large satellite missions. These challenges include the planning and scheduling of timelines (such as the satellite range scheduling problem) and the need to share data among various stakeholders (for example to the mission scientists in order to plan observations). Independent of how those optimizations are done, what is often needed is a way to conveniently display such information and make it available to the operators.

From analyzing various use cases of multi-satellite, multi-groundstation mission operations planning and monitoring, three generic plot types have stood out: timeline plots, timeseries (parameter) plots, and map plots that provide most tactical view on the current state of operations.

For this reason we have developed an open source web framework that allows mission operations to visualize their mission data using these kinds of plots. Being open-source it allows for full customization and community-driven enhancements. The framework is composed of a flexible database and a web server, which provides for data ingestion, processing, and visualization, as well as an API endpoint for further extensions.

The database and web server have been exclusively written in Python 3 language that provides powerful features while being extremely readable (and thus maintainable). This is a key advantage of our implementation as Python provides a rich ecosystem for data science to substantially boost data ingestion and analysis. Moreover, also the entire visualization pipeline is written in Python using an interactive plotting library for web browsers. This way, programmers do not need to shift languages between data preparation and data visualization.

We demonstrate on several use cases, that such a web-based tool can significantly raise awareness of operators on planning and scheduling, resulting in faster detection of scheduling conflicts and to help investigation of anomalies.

Providing such customizable visualizations of timelines, timeseries data, and geomap locations have already spawned interest in application for other domains, such as fleet operations of drones and rovers.