SatNOGS: Towards a Modern, Crowd Sourced and Open Network of Ground Stations

Manolis Surligas[1,2], Matthaios Papamatthaiou[1], Ilias Daradimos[1], Vasilis Tsiligiannis[1], Pierros Papadeas[1], Agisilaos Zisimatos[1], Fredy Damkalis[1], Kostis Triantafillakis[1], George Vardakis[1,2], Nestoras Sdoukos[1,2], Nikos Karamolegkos[1,2]
[1] Libre Space Foundation, Athens Greece, info@libre.space
[2] Computer Science Department, University of Crete

Abstract

Over the last years the launching cost of a payload in space has been significantly reduced and this trend is expected to continue, as the interest for space applications is increasing. The reduced launch cost and the advancements in technology, gave the opportunity to small satellites to revolutionize access to space.

The majority of the small satellites missions are targeting the Low Earth Orbit (LEO). Due to the nature of this particular orbit, communication with a satellite is possible only for a few minutes per day for a given location. This raises the need for multiple ground stations in several geographic locations. Although such an infrastructure is possible, most of the times it is both complicated and expensive for research or educational entities to obtain. Given the fact that each ground station exhibits a small per day utilization for a specific satellite, the idle time can be used for reception of other missions.

SatNOGS is an open source software and open hardware project that addresses this problem by interconnecting all participating ground stations, offering their idle time to other users of the SatNOGS network.

1 Introduction

SatNOGS is a global network of satellite ground stations, designed as an open source participatory project that receives data from Low Earth Orbit (LEO) satellites. This particular orbit provides the ground station with a reception window that is limited to a few minutes. Therefore, a ground station remains underutilized and idle for most of the time. On the other hand, using a single ground station means that there are only a few communication windows to send or receive data from a LEO satellite each day.

The concept of SatNOGS, is to use the deployed ground stations around the world in an efficient manner so that both the underutilization of them as well as the coverage problems can be resolved with a single solution. Aground station operator joining the SatNOGS network, provides the idle time of their ground station to other users, while at the same time they can take advantage of ground stations at various locations of the earth to schedule observations.

2 Architecture

The SatNOGS ecosystem consists of several components operating interchangeably. The SatNOGS Network infrastructure orchestrates the scheduling of each ground station, based on the trajectory of the targeted satellites and the online available ground stations, while allowing the owner of a ground station to have complete control over her hardware. The SatNOGS Database (DB)[1] is responsible for keeping the orbital elements and communication related information of each satellite. It also holds decoded frames from the deployed ground station network. The Data Warehouse, visualizes graphically the gathered data from each satellite using a web interface and the Grafana visualization framework. Last but not least, the SatNOGS Ground Station is the necessary software and hardware, that allows satellite tracking, the control of the RF front-end, signal reception and demodulation of possible data frames.

2.1 The network

The SatNOGS Network[2] infrastructure is the backbone of the SatNOGS ecosystem and instruments essential functionalities for the flawless, efficient and effective operation of the entire project. The network infrastructure services can be divided into two main categories, the back-end and the front-end. At the back-end, the SatNOGS Network keeps track of the available online ground stations and the list of the observation jobs that each one has been assigned. Moreover, it receives observation data from the deployed ground stations. The received data are either stored in the network infrastructure to be used through the web interface or are submitted to the SatNOGS DB[1] for fur-

ther analysis. The front-end of the SatNOGS Network provides a set of different web interfaces. For the ground station operators, a web-based control panel is available in order to control and configure remotely their stations. It also allows operators to schedule

2.2 Rotator hardware

SatNOGS Rotator is the mechanism that allows tracking of satellites in both azimuth and elevation axis. By design, the goal of the rotator is to keep the cost low, by using widely available materials of common sizes and 3D printed parts. For the users that cannot afford a rotator, SatNOGS can still operate without one, using a less directional antenna.

2.3 Client software

The SatNOGS Client[4] is a Python program that runs on the ground station computer. The client is responsible to retrieve observation jobs from the network and execute them. When a new job is received from the network, it is placed in an execution queue, sorted in chronological order based on the start time of the satellite pass. The client constantly monitors the local time of the ground station and the starting time of the first observation at the queue. When the timing is proper, the client removes the observation job from the queue and prepares to execute it. To do so, it initializes the SDR front-end with the RF parameters described by the job and then executes the appropriate GNU Radio script provided by the gr-satnogs OOT module[3]. Meanwhile, the client controls the rotator, so the antennas can track the trajectory of the targeted satellite. When the observation job is finished, the client instructs the rotator to place the antennas in a park position. Then, all the resulting files generated by the grsatnogs OOT[3] are uploaded back to the network. The client continues to operate, waiting for the execution of the next job in the queue.

2.4 Gnuradio OOT module

Each ground station is equipped with an SDR device for the signal reception. The architecture is modular enough, so it can support a wide range of different SDR hardware, depending on the target cost of the station. For example, there are stations using an RTL SDR dongle costing about 15USD, whereas others utilize a high-end device like the USRP B210 with a cost of 2000USD. For the signal analysis and demodulation, the GNU Radio Out-of-Tree (OOT) module called gr-satnogs [3] is used. The software on its core is written in C++, using also some Python bindings. The first task of this module is to receive the signal from the SDR front-end, apply coarse filtering and re-sample it. The later is crucial in order to reduce the sampling rate of the signal originating from the SDR device, so the processing can be performed in CPU limited devices like the Raspberry Pi 3. Afterwards and based on the satellite trajectory, the gr-satnogs compensates the Doppler effect which introduces a constantly changing center frequency offset. Then, the Doppler corrected signal passes additional and more fine grained filtering stages. Finally, the module tries to automatically demodulate the resulting signal in real-time, based on the coding/modulation scheme of the targeted satellite. At the same time, the module produces a waterfall spectrum analysis plot and an audible representation of the spectrum. The waterfall plot is an excellent tool, for immediate and visual spotting of satellite transmissions, nearby interference, possible RF performance issues or misconfiguration at the station setup. The audible transformation of the spectrum, is a technique quite popular in the amateur community and many amateur signal analysis tools utilize it. For each observation, the decoded frames, a waterfall spectrum analysis plot and the audio file are uploaded back to the SatNOGS Network, for visualization and further analysis. Currently, gr-satnogs provides automated demodulators decoders that cover a wide range of satellite missions. From beacons to weather pictures.

3 Conclusion

The use of gnuradio enable an unmatched agility in a groundstation network. The ability to demodulate virtually any kind of transmission is fully future proof. It only requires automated software update of the client. It is also the perfect match with low cost SDR receiver like rtl-sdr.

References

[1] *SATNOGS Database :* `https://db.satnogs.org/`

[2] *SATNOGS network :* `https://network.satnogs.org/`

[3] *gr-satnogs :* `https://gitlab.com/librespacefoundation/satnogs/gr-satnogs`

[4] *SATNOGS client :* `https://gitlab.com/librespacefoundation/satnogs/satnogs-client`