Adaptation of a GNSS open-source receiver for analysis of LEO-PNT scenarios

Fran Fabra
IEEC-CERES
Universitat Autònoma de Barcelona
Barcelona, Spain
FranciscoJose.Fabra@uab.cat

José A. López-Salcedo
IEEC-CERES
Universitat Autònoma de Barcelona
Barcelona, Spain
Jose.Salcedo@uab.cat

Gonzalo Seco-Granados
IEEC-CERES
Universitat Autònoma de Barcelona
Barcelona, Spain
Gonzalo.Seco@uab.cat

***Abstract*—With the advent of the first Low Earth Orbit (LEO) mega-constellations, the concept of Low Earth Orbit for Position, Navigation, and Timing (LEO-PNT) has become a hot topic in the Global Navigation Satellite Systems (GNSS) community. In this context, this paper provides a description of the required modifications applied to an open-source GNSS software receiver to work properly under LEO dynamics. In addition, a set of key indications is given to adapt a GNSS simulator to generate a LEO-PNT scenario. Finally, a comparison of the end-to-end performance between GNSS and LEO-PNT is analyzed through a case example, which illustrates the benefits of properly adapting the tracking strategy to each scenario.**

***Index Terms*—LEO-PNT, GNSS, open-source, Skydel, Galileo, Globalstar**

I. Introduction

The use of signals transmitted from Low Earth Orbit (LEO) platforms for Position, navigation, and Timing (PNT) purposes has become a hot topic during the last years [1] [2]. Two main factors are behind this fact. On the one hand, the requirement for faster and more accurate PNT solutions than those provided by current Global Navigation Satellite Systems (GNSS) due to more demanding applications such as, e.g., autonomous car/truck driving, augmented reality in urban environments, accurate beamforming in high-rate communication systems (e.g. 6G) and navigation of drones. On the other hand, the advent of new industrial and business models based on LEO platforms with medium and small sizes have enabled large-scale manufacturing of onboard devices and lower-cost launches, thus opening the door to a wider range of actors in a sector historically limited to governmental projects and big companies. Moreover, such simplicity is also given by GNSS, which enables Orbit-Determination and Time Synchronisation (ODTS) of the LEO satellites.

The key differentiators of LEO-PNT compared against Medium Earth Orbit (MEO) GNSS, which can either serve to enhance current PNT services by increasing measurement diversity and/or to enable new capabilities, are mainly [3] [4]:

- Geometric diversity: when considered as a complementing tool for GNSS, the larger number of satellites provided by LEO-PNT might help to improve satellite-to-user geometry in scenarios with limited GNSS visibility, such as urban environments, deep-valleys or canyons and locations at high latitudes. However, the most significant aspect of LEO transmitters is their faster velocity, which implies larger Doppler values (allowing Doppler-based positioning) and faster decorrelation of observables (multipath whitening effect). In addition, such faster geometric variations will be affected by shorter outages in case of NLOS events.
- Frequency diversity: the use of smaller and simpler payloads allows a more affordable deployment of LEO constellations. This fact encourages the exploration of the use of a potentially wider range of frequencies than GNSS. For example, sub-GHz frequencies such as Ultra-High Frequency (UHF) have a better power budget and larger penetration properties, thus being a very suitable option for achieving indoor positioning or working under canopy environments. Frequencies in the vicinity of GNSS bands allow better interoperability with those systems while increasing robustness and resilience in jamming environments. Finally, higher frequency bands such as K/Ka enable the use of high bandwidth signals that could reach rather accurate performance with code observables. Moreover, better ionospheric corrections with dual-frequency measurements can be obtained by combining signals with larger frequency differences.
- Better link budget: by orbiting the Earth at a lower altitude, LEO signals benefit from a better link budget compared to GNSS signals. Around 20 dB of decrease in free space losses can be expected when comparing a LEO with a MEO. However, such improvement is not directly translated into a 20 dB increment of received power, given that current regulations on power flux density over the Earth's surface might limit the maximum transmitted power depending on the frequency band selected and signal bandwidth.

All these aspects represent a set of opportunities and challenges from the point of view of the receiver, which opens

This activity has been partially supported by the Catalan Government in the framework of the New Space Strategy of Catalonia.

979-8-3503-5685-4/24/$31.00 ©2024 IEEE

a new field of research in the GNSS community (e.g. [5], [6], [7], [8]). In particular, open-source Software Defined Radio (SDR) solutions represent a proper means towards standardization [9], given that all the internal details and configuration aspects are publicly available, thus enabling a common framework of analysis and discussion.

In this context, the aim of this work is to provide a means to evaluate LEO-PNT scenarios based on an adaptation of a GNSS open-source receiver. Section II describes the simulation framework, where the first part provides a set of indications on how to adapt Orolia's Skydel simulator for a LEO scenario, and the second part gives the description of the modification done in FGI-GSRx to properly work under high dynamics. Then, Section III shows the results obtained in a case example where signals from the Galileo system are transmitted both from MEO and LEO. The source code and the configuration files employed are available at [10].

II. Simulation framework for LEO-PNT

A. Signal generation based on Skydel

In order to simulate the signals that would be received in a LEO-PNT scenario, one can either generate a desired set of baseband, GNSS signals to then apply a channel emulator to include the impact on range, Doppler, gain, etc., or properly adapt for that purpose the settings of a professional hardware GNSS simulator such as Orolia's GSG-8 based on Skydel Simulation Engine. In this work, we just followed the guidelines from [11] to create and simulate a LEO scenario by turning the Galileo orbit into a LEO and running a simulation with GSG-8. While most of the adaptations are rather straightforward, such as setting the transmitted power to a given value to avoid the default model applied in the original GNSS constellation, the orbit modification in Skydel requires careful attention.

Typically, one can get the orbital parameters of known LEO constellations from the Two Line Elements (TLE) files available at [12]. However, it is worth mentioning that, while the TLE parameters come from the astronomy field, the orbital parameters in Skydel follow the Ephemeris Data definitions (RINEX format) from the Global Positioning System (GPS). A procedure to properly map parameters from a TLE file to RINEX format is described in [13]. Basically, while a mere unit conversion is required for most of the values and rate parameters must be set to zero, the tricky aspect comes when computing the longitude of the ascending node (Ω_0). The corresponding value contained in TLE files (also called the right ascension of the ascending node), Ω_{TLE}, is referred to as an Earth Centered Inertial (ECI) reference frame. However, the Ω_0 injected in Skydel needs to be provided in an Earth Centered Earth Fixed (EFEC) reference frame but referred to the start of the week. To do so, it is only needed to compute the right ascension of the Greenwich meridian at the time epoch corresponding to the beginning of the week, $\Omega(t_{\mathrm{init\_week}})$, with respect to the reference epoch of the TLE file for a given satellite (the previous Sunday at midnight can be taken as $t_{\mathrm{init\_week}}$). Finally, Ω_0 is obtained from:

$$\Omega_0 = \Omega_{\mathrm{TLE}} - \Omega(t_{\mathrm{init\_week}}). \quad (1)$$

The computation of $\Omega_{\mathrm{init\_week}}$ can be made in MATLAB as the output of function "siderealTime" when the input is $t_{\mathrm{init\_week}}$. Another alternative is to use "ecef2eci" function to convert a unitary vector pointing towards the Greenwich meridian at the equatorial plane in the ECEF frame ($[1, 0, 0]$) to its ECI counterpart at $t_{\mathrm{init\_week}}$, to finally compute $\Omega_{\mathrm{init\_week}}$ as the angle of the resultant vector in the XY plane (arctangent of the two first coordinate values).

B. Software receiver based on FGI-GSRx

The other side of the simulation framework consists of a modified version of FGI-GSRx [14] [15], an open-source GNSS software receiver. In order to deal with the higher dynamics produced by LEO transmitters, the acquisition and tracking engines have been properly adapted to work in such challenging environments [7].

In terms of signal acquisition, the most relevant modification has been to include code Doppler compensation in the clean-replica models. This feature, usually neglected in standard GNSS scenarios, might have a significant impact when longer integration times are needed under high dynamics or when using high-rate codes [16]. In addition, Doppler and Doppler-drift aiding capabilities have been included in order to alleviate the computational demands under justified circumstances, such as the simulation of a scenario where a low-energy device has an initial estimation of its position and the LEO constellation.

Regarding tracking, higher-order lock loops have been included with respect to the original version. This is a relevant feature under LEO-PNT, given that the dynamic stress error is proportional to the range derivative of the corresponding order [17], i.e. while second-order loops are sensitive to Doppler drift (second range derivative), third-order loops are sensitive to jerk (third range derivative). In addition, the tracking engine follows a state-machine process that depends on the comparison of frequency and phase lock indicators against configurable thresholds. The standard state sequence would be:

- Pull-in: Initial state with wide loop bandwidths that allows a proper transition from acquisition to tracking or recovery from a loss-of-lock event. Phase not yet tracked.
- Coarse tracking: Transitional state with narrower loop bandwidths where phase starts to be tracked.
- Fine tracking: Final state with very narrow loop bandwidths to achieve high precision in range and frequency/phase estimates. Steady-state conditions are reached at this point.

The modified version of the receiver allows the selection of different lock loop strategies at the different states. Moreover, special care is taken with the memory of the loop filters when doing the transition between coarse and fine tracking states. Both acceleration and velocity accumulators are smoothed using a running window (with a configurable length) at this

stage. This feature, not required in standard GNSS, was found to have a positive impact under high dynamic cases.

III. End-to-end analysis

In order to illustrate the performance of the modified version of the receiver, two scenarios have been simulated: a LEO and a MEO. The purpose is to evaluate the quality of the primary observables obtained after tracking (code-range and Doppler) by comparing them against the reference values given by the signal generator under different tracking strategies.

A. Case example

The orbits selected for the LEO case are the ones from the Globalstar system, at around 1000 km of altitude, while the MEO case is based on the Galileo system, at around 23600 km. TLE files from [12] are employed for both cases to configure the corresponding orbits at Skydel. Despite that this process would not be required for Galileo, given that its orbits are already available at the simulator, it has been done here for validation purposes of the procedure itself (showing good agreement with the expected visibility at the time window selected). The signal employed is Galileo E1B for both scenarios, with only BOC(1,1) modulation for simplicity.

The receiver is located at the coordinates of our laboratory at UAB. The initial satellite visibility for both cases is displayed in Fig. 1, where the different numbers indicate the pseudorandom noise (PRN) sequence transmitted. For the LEO case, this initial visibility has to be checked before running the simulation to properly assign the orbital parameters of the Globalstar satellites in view to different Galileo satellites (associated with a particular PRN).

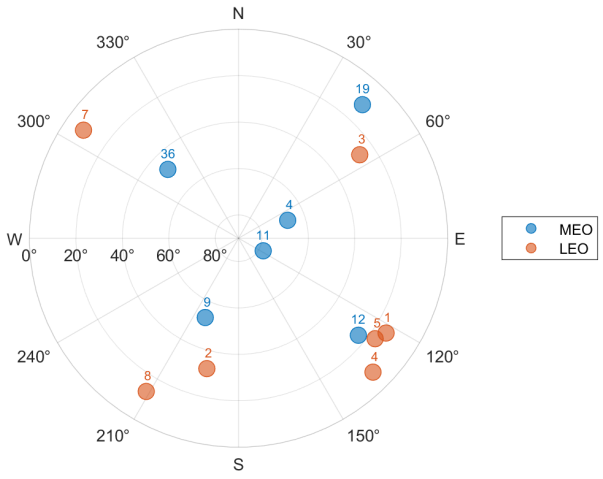

Fig. 1. Satellite visibility under the simulated MEO and LEO scenarios (the numbers indicate the PRN sequences employed).

The time window of the simulation for both scenarios is set to 10 seconds. Tables I and II provide the mean values for Doppler, Doppler drift, and jerk, which are the three derivatives of range as a function of time, for MEO and LEO cases respectively. While the maximum absolute Doppler shows an increase of up to one order of magnitude by lowering the orbit of the transmitters, the impact of LEO dynamics is especially relevant when checking the Doppler derivatives.

TABLE I
Mean dynamics for the MEO case

PRN	Doppler [Hz]	Doppler drift [Hz/s]	Jerk [Hz/s^2]
4	-697	-0.39	0.00009
9	1938	-0.33	-0.00004
11	-493	-0.34	0.00008
12	-2481	-0.19	0.00013
19	-2776	-0.08	-0.00024
36	1901	-0.19	0.00003

TABLE II
Mean dynamics for the LEO case

PRN	Doppler [Hz]	Doppler drift [Hz/s]	Jerk [Hz/s^2]
1	2282	-59.6	-0.029
2	21202	-46.4	-0.261
3	-23155	-16.9	0.080
4	9008	-45.5	-0.073
5	3322	-63.1	-0.045
7	28885	-4.9	-0.046
8	-13488	-46.2	0.121

B. Results obtained

Once the datasets have been generated, the modified version of the software receiver processes the raw samples to provide code-range and Doppler estimations under both scenarios. The first step consists of applying the acquisition engine to find the signals hidden under the noise floor and obtain a preliminary estimation of code and Doppler positions over the search grid. The size of the grid area is adapted to the length of the codes employed (4092 chips at 1.023 Mcps) in the time domain, and to the expected maximum absolute Doppler in the frequency domain (4 kHz for MEO and 40 kHz for LEO). On the other hand, the resolution of the individual cells is defined by the inverse of the sampling frequency (25 MHz) in the time domain and the inverse of the coherent time of integration (set to 8 ms) in the frequency domain. The acquisition process works fine in both scenarios under analysis without requiring any aiding (all satellites were acquired without false events). Moreover, the modifications done at the receiver for LEO dynamics, which can be disabled, do not have a significant impact on the results given that no longer integrations are required. However, the high computational cost for the LEO case (cold start mode) as a result of an extended search area, would require special addressing in a real-time application.

After signal acquisition, the receiver starts the tracking process. In order to evaluate the impact of LEO dynamics, two different tracking strategies are applied in both scenarios depending on the orders of the frequency lock loops (FLL) and phase lock loops (PLL) employed:

- **Strategy #1**: a first-order FLL is used during pull-in, then it is changed to a second-order PLL aided with a first-

order FLL during coarse tracking to end up with a second-order PLL alone at fine tracking. This option represents a traditional choice in standard GNSS where there is a transition from a purely FLL to a purely PLL to reach accurate precision from phase estimations.
- **Strategy #2**: a third-order PLL aided with a second-order FLL is employed during all the tracking stages. This option is adapted to achieve good performance while keeping robustness under higher dynamics by means of loops with higher order and the permanent aiding of the FLL.

In both strategies, the delay lock loop (DLL) is of second order and it is aided by the FLL/PLL. For the sake of simplicity, the same loop bandwidth values are set for both strategies: 15 Hz during pull-in, 7 Hz during coarse tracking, and 2 Hz (FLL/PLL) and 1 Hz (DLL) during fine tracking.

By working with an integration time of 4 ms, up to 10 s of data are processed under each scenario and tracking strategy. The range and Doppler estimations obtained are then compared against the reference values provided by Skydel during the simulations. The error values are computed as the standard deviation (1 σ) of the difference between estimations and models (the resultant bias is negligible in all cases). The time interval considered for these computations is the last 5 s to account only for steady-state conditions and avoid the initial fluctuations due to acquisition to tracking transition. Tables III and IV provide the results obtained under the MEO scenario. As could be expected, the use of tracking strategy #2 does not provide any improvement in this case. On the contrary, the contribution of the FLL limits the performance that could be reached by the PLL alone, without barely any impact at code-range level. The overall results mainly depend on the evolution of Carrier to Noise density ratio (CN0), whose estimation is also given.

TABLE III
RESULTS FOR THE MEO CASE USING TRACKING STRATEGY #1

PRN	CN0 [dB-Hz]	Range error [m]	Doppler error [Hz]
4	39.9	0.21	0.04
9	35.4	0.31	0.06
11	45.9	0.25	0.04
12	33.2	0.82	0.10
19	39.2	0.52	0.11
36	26.7	0.79	0.24

TABLE IV
RESULTS FOR THE MEO CASE USING TRACKING STRATEGY #2

PRN	CN0 [dB-Hz]	Range error [m]	Doppler error [Hz]
4	39.9	0.21	0.08
9	35.4	0.31	0.10
11	45.9	0.25	0.07
12	33.2	0.82	0.18
19	39.2	0.52	0.19
36	26.7	0.80	1.18

Tables V and VI provide the results obtained under the LEO scenario. In this case, the adoption of higher-order loops with permanent FLL aiding from tracking strategy #2 clearly benefits the overall performance, achieving similar results as in the corresponding MEO case. A closer view of the results obtained with strategy #1 reveals that the receiver cannot keep the fine tracking state for any satellite, thus disabling the benefits of using a narrower loop bandwidth against the contribution from thermal noise. The impact of dynamic stress error is the main contributor behind such behavior and therefore the loop bandwidth at this state should be properly adapted to achieve the best performance. It is worth mentioning that, while third-order loops are sensitive to jerk, second-order loops are affected by Doppler drift, and a comparison of both variables in absolute terms reveals a two-order magnitude difference for this particular case (values given in Table II). The dynamic stress error is directly proportional to these terms [17] and thus becomes the major contributor to the total error when using strategy #1 at the LEO scenario.

TABLE V
RESULTS FOR THE LEO CASE USING TRACKING STRATEGY #1

PRN	CN0 [dB-Hz]	Range error [m]	Doppler error [Hz]
1	41.3	1.01	1.91
2	39.4	0.88	1.46
3	38.2	1.10	1.37
4	32.6	1.58	1.62
5	29.2	2.95	2.94
7	32.5	2.66	1.69
8	33.7	1.57	1.56

TABLE VI
RESULTS FOR THE LEO CASE USING TRACKING STRATEGY #2

PRN	CN0 [dB-Hz]	Range error [m]	Doppler error [Hz]
1	41.3	0.35	0.10
2	39.4	0.33	0.07
3	38.2	0.31	0.08
4	32.6	0.53	0.16
5	29.2	1.57	0.81
7	32.5	0.96	0.31
8	33.7	0.62	0.13

IV. DISCUSSION

The tools and indications provided in this paper, which are publicly available in [10], can be employed to explore a wide variety of scenarios in LEO-PNT, an emerging field of research nowadays. Regarding the case example analyzed, it is important to point out that this paper aims to illustrate some of the limitations that standard GNSS receivers may find when trying to work under the higher dynamics that are intrinsic to the transmitters in LEO-PNT. However, these effects have a major impact at higher frequencies than L-band, so way more challenging scenarios can be analyzed. In such cases, the modifications included in the acquisition engine could play a more relevant role or external aiding might eventually be a mandatory requirement. On the other hand, a better power budget could be expected for most LEO cases, which somehow could balance the impact of the dynamic stress error. In any

case, the work presented here provides a valid framework to explore all these ways of research.

REFERENCES

[1] W. Stock, R. T. Schwarz, C. A. Hofmann and A. Knopp, "Survey On Opportunistic PNT With Signals From LEO Communication Satellites," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2024.3406990.

[2] T. Janssen, A. Koppert, R. Berkvens and M. Weyn, "A Survey on IoT Positioning Leveraging LPWAN, GNSS, and LEO-PNT," in IEEE Internet of Things Journal, vol. 10, no. 13, pp. 11135-11159, 1 July, 2023, doi: 10.1109/JIOT.2023.3243207.

[3] L. Ries et al., (2023) "LEO-PNT for Augmenting Europe's Space-based PNT Capabilities," in IEEE/ION Position, Location and Navigation Symposium (PLANS), pp. 329-337. doi: 10.1109/PLANS53410.2023.10139999.

[4] F. S. Prol et al., (2022) "Position, Navigation, and Timing (PNT) Through Low Earth Orbit (LEO) Satellites: A Survey on Current Status, Challenges, and Opportunities," IEEE Access, vol. 10, pp. 83971-84002, doi: 10.1109/ACCESS.2022.3194050.

[5] P. A. Iannucci and T. E. Humphreys, "Fused Low-Earth-Orbit GNSS," in IEEE Transactions on Aerospace and Electronic Systems, doi: 10.1109/TAES.2022.3180000.

[6] De Bast, S.; Sleewaegen, J.-M.; De Wilde, W. "Analysis of Multipath Code-Range Errors in Future LEO-PNT Systems". Eng. Proc. 2023, 54, 34. https://doi.org/10.3390/ENC2023-15453

[7] F. Fabra, J. A. Lopez-Salcedo, G. Seco-Granados, "Analysis on baseband algorithms for LEO PNT", Proc. European Navigation Conference (ENC), Jun 02, 2023.

[8] R. Morales Ferre, J. Praks, G. Seco-Granados and E. S. Lohan, "A Feasibility Study for Signal-in-Space Design for LEO-PNT Solutions With Miniaturized Satellites," in IEEE Journal on Miniaturization for Air and Space Systems, vol. 3, no. 4, pp. 171-183, Dec. 2022, doi: 10.1109/JMASS.2022.3206023.

[9] T. Pany et al., "GNSS Software-Defined Radio: History, Current Developments, and Standardization Efforts", NAVIGATION: Journal of the Institute of Navigation, Mar 2024, 71 (1) navi.628; DOI: 10.33012/navi.628

[10] FGI-GSRx for LEO-PNT. Available online: https://gitlab.com/spcomnav_public/fgi-gsrx-leo-pnt

[11] Creating and Simulating LEO Constellations with Skydel. Available online: https://safran-navigation-timing.com/document/creating-leo-constellations-with-skydel/

[12] Celestrak: NORAD GP Element Sets. Available online: https://celestrak.org/NORAD/elements/

[13] M. Garcia-Fernandez, "Mapping TLE orbital parameters to GNSS ephemeris for LEO PNT mega-constellation orbit simulations and visibility analysis", arXiv:2401.17767, Jan 31, 2024.

[14] FGI-GSRx Software Receiver. Available online: https://github.com/nlsfi/FGI-GSRx

[15] Borre K, Fernández-Hernández I, López-Salcedo JA, Bhuiyan MZH, eds. *GNSS Software Receivers.* Cambridge University Press; 2022.

[16] Foucras, M.; Julien, O.; Macabiau, C.; Ekambi, B.; Bacard, F. Assessing the Performance of GNSS Signal Acquisition: New Signals and GPS L1 C/A Code. Inside GNSS, Inside GNSS Media LLC. 2014; pp. 68–79.

[17] Kaplan, E.D.; Hegarty, C.J. *Understanding GPS: Principles and Applications*, 2nd ed.; Artech House: Boston, MA, USA, 2006.