4. Orbit propagator

This section aims to explain the functionalities of the novel software developed to provide a customizable constellation, in order to run the simulator presented in [9], with different LLA inputs, other than the "default" 5 days long set provided by *SES S.A.*. The rationale behind this implementation lies on the fundamentals orbital mechanics principles described in Chapter 2.2, where to preserve ease of execution, only the R2BP is performed, and geocentric LLA are obtained, omitting expensive tasks that would not be sound for constellations of thousands of satellites, where high precision localization is not required. The Simulink block of the Orbital Propagator from Aerospace Toolbox has been considered a valid alternative wrt. a manual integration of the routine, especially because of the great *vectorization* power that it provides. This comes in very handy when time performance is crucial, considering that this is a tool to be appended to the mentioned power budget simulator, that is quite computationally intensive by itself. Indeed, the aim of this program is to provide high performance within an acceptable degree of accuracy: the "Periodic cycle mode" stems from the concept used by the LLA sets from SES, i.e. if the simulation needs to prepare annual statistics, it is not required to propagate thousands of satellites for that long. Surely it might be inexpensive to compute the full propagation for a few satellites, in fact it is up to the user to decide, but the renewed design yields a fairly advantageous result. There are multiple setups available, described in the following list.

- `Periodic cycle mode`, as mentioned allows to reduce the propagation time. The period time can be chosen by the user (that is equivalent to choosing a shorter simulation time, similar to the original inputs), or can be automatic. The `Auto` mode relies on the `Max array size` choice, which limits the length of the propagation. This mode uses a simple algorithm to approximate the *period repetition* of any orbit, as presented in [25], without considering the oblateness nor the nodal regression. The groundtrack westward shift is:

$$S = T\omega_{\mathrm{E}} \tag{4.1}$$

where the rotation rate of the Earth $\omega_{\mathrm{E}} = 0.250684454\,\frac{\mathrm{deg}}{\mathrm{min}}$ (as in Eq.:2.2), and P is the satellite's Keplerian period. Hence the satellite trace repetition rational(-izable) parameter:

$$Q = \frac{360}{S} = \frac{N}{D}. \tag{4.2}$$

If Q is an integer, the orbit is *repeating* its *ground-track* after Q revolutions a day, otherwise N and D are integers: N is the number of orbit revolutions until repeat, and D is the number of days until repeat. Therefore, the orbit is built such that the total number of satellites would fit in a 3D-array smaller than the maximum allowed memory size.

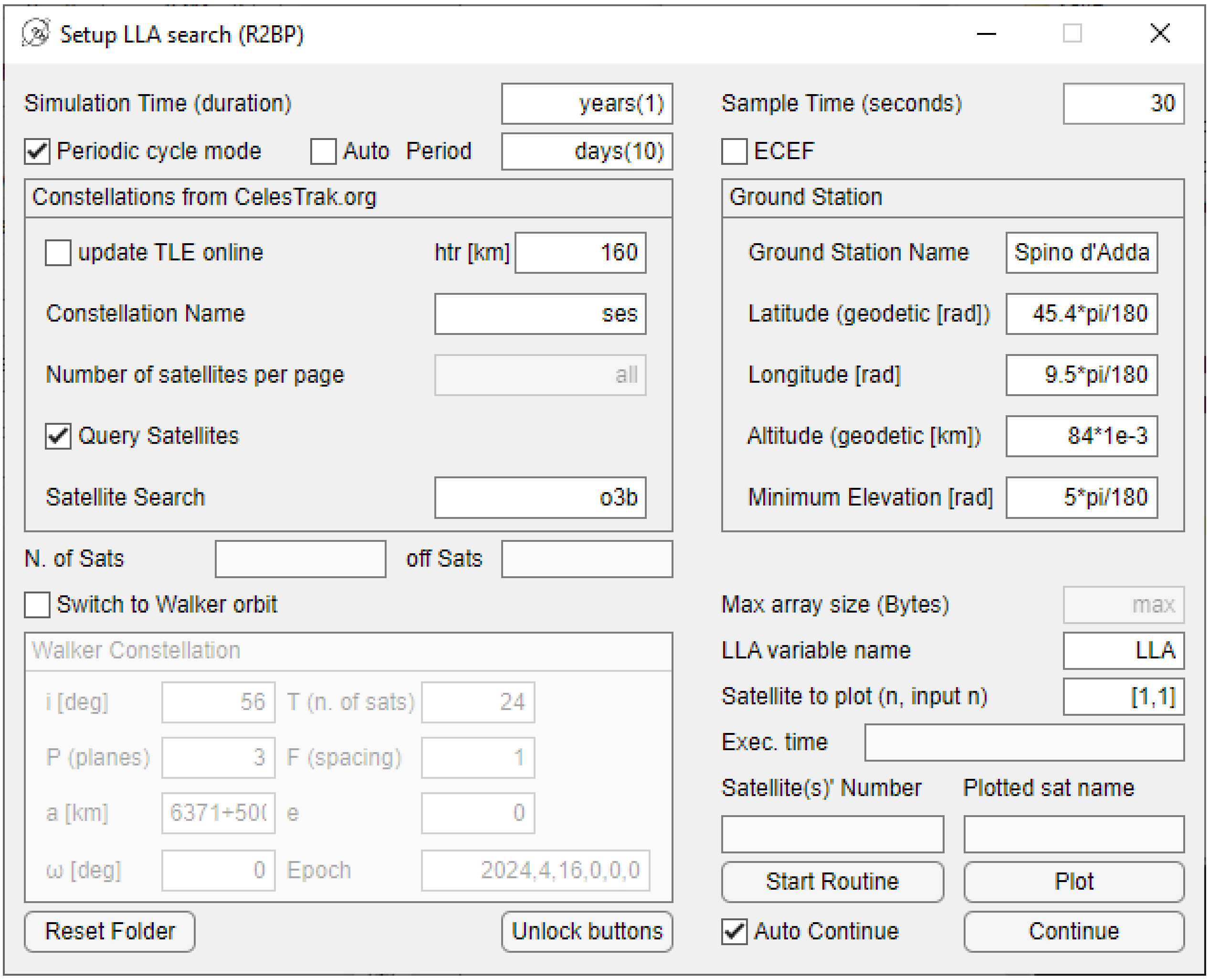


Figure 4.1: The Graphical User Interface of the orbital propagator, developed in App Designer, MATLAB®.

- The TLE sets can be downloaded from **CELESTRAK**®, the satellites can be selected by the name of the constellation, accessible from [40]. A number of satellites can be chosen (consider the first n sats. of one page) or queried searching by keywords (e.g. name of the single satellite or subconstellation), and those below a certain altitude can be excluded (decommissioned or early off-nominal mission).
- Another option is to build the very flexible *Walker-Delta* orbits: symmetric arrangements of satellites, denoted by the three parameters $T/P/F$: T is the total number of satellites, which has been simplified by making it the number of satellites per plane, P is the number of equally distanced planes ($\Delta\Omega = 360/P$) sharing the same inclination, F is the relative phasing parameter, i.e. relating the satellite positions in one orbital plane to those in an adjacent plane. The remaining parameters to fully define the orbits: a, i, e, ω, Epoch date, see Chapter 2.2. More than one input parameter can be inserted per routine.
- As mentioned above, the LLA triplet (Latitude, Longitude, Altitude) is the actual output, served to the main simulator to provide Attenuations along GSO and NGSO paths, together with the power budget distributions. The LLA results are computed after ECI R2BP propagation, and at the end of the process they are displayed on a Ground-Track planisphere and three dimensional

plots, respectively, where the `start` ○ and the `end` □ points of the orbital paths are marked.

- The last addition is the embedded ECEF propagation, which can be useful to check the correctness of the LLA conversion from ECI at the right epoch (i.e. the whole constellation evolves in the same timeline, but to obtain the starting longitude is no trivial endeavour, as the same epoch has to be fixed for the whole system [26]). This is not recommended for the main iteration of the procedure, simply because of how the fixed-frame propagation is structured in Simulink within the *high precision numerical integration* algorithms, following Eq.: 2.11 and subsequent rotations with *quaternions*, definitely not as efficient as the alternative. In fact, the *Keplerian unperturbed* block implements a *Newton–Raphson* (NR) iterative process [41], an analytical algorithm that works for central body spherical gravity, without perturbations, and is suitable for the inertial frame.

4.1. The *Newton – Raphson* method for Keplerian unperturbed propagation

The NR *universal–variables method* [22] to predict the position and velocity of an object in space is based on Kepler's laws of planetary motion, in particular it is a general method solution of the *Kepler's problem*. Given the initial state vector $\underline{\mathbf{x}}(t_0) = \Big(\underline{\boldsymbol{r}}(t_0), \underline{\boldsymbol{v}}(t_0)\Big)$, the *orbital energy* ξ and $\alpha = \dfrac{1}{a}$ are:

$$\xi = \frac{v_0^2}{2} - \frac{\mu}{r_0} = -\frac{\mu}{2\,a} < 0 \;\text{ for elliptical orbits}\,, \tag{4.3}$$

$$\alpha = -\frac{2\,\xi}{\mu}\,, \tag{4.4}$$

where $\alpha > 0$ for elliptical orbits. The initial guess for the *iterand* χ is:

$$\chi_0 = \sqrt{\mu}\,\alpha\,\Delta t\,, \tag{4.5}$$

where Δt is the propagation step size, 30 seconds, and μ is the Earth gravitational parameter. Performing the NR iteration, while $|\chi_n - \chi_{n-1}| < 10^{-6}$:

$$\chi_{n+1} = \chi_n + \frac{\sqrt{\mu}\,\Delta t - \chi_n^3\, c_{3,n} - \dfrac{\boldsymbol{r}_0 \cdot \boldsymbol{v}_0}{\sqrt{\mu}}\,\chi_n^2\, c_{2,n} - r_0\,\chi_n\Big(1 - \psi_n c_{3,n}\Big)}{r_n}\,, \tag{4.6}$$

where:

$$r_n = \chi_n^2 c_{2,n} + \frac{\boldsymbol{r}_0 \cdot \boldsymbol{v}_0}{\sqrt{\mu}}\,\chi_n\Big(1 - \psi_n c_{3,n}\Big) + r_0\Big(1 - \psi_n c_{2,n}\Big)\,, \tag{4.7}$$

$$\psi_n = \chi_n^2\,\alpha\,, \tag{4.8}$$

which means that $\psi_n > 0$ for elliptical orbits, therefore:

$$c_{2,n} = \frac{1 - \cos\Big(\sqrt{\psi_n}\Big)}{\psi_n}\,, \tag{4.9}$$

$$c_{3,n} = \frac{\sqrt{\psi_n} - \sin\Big(\sqrt{\psi_n}\Big)}{\sqrt{\psi_n^3}}\,, \tag{4.10}$$

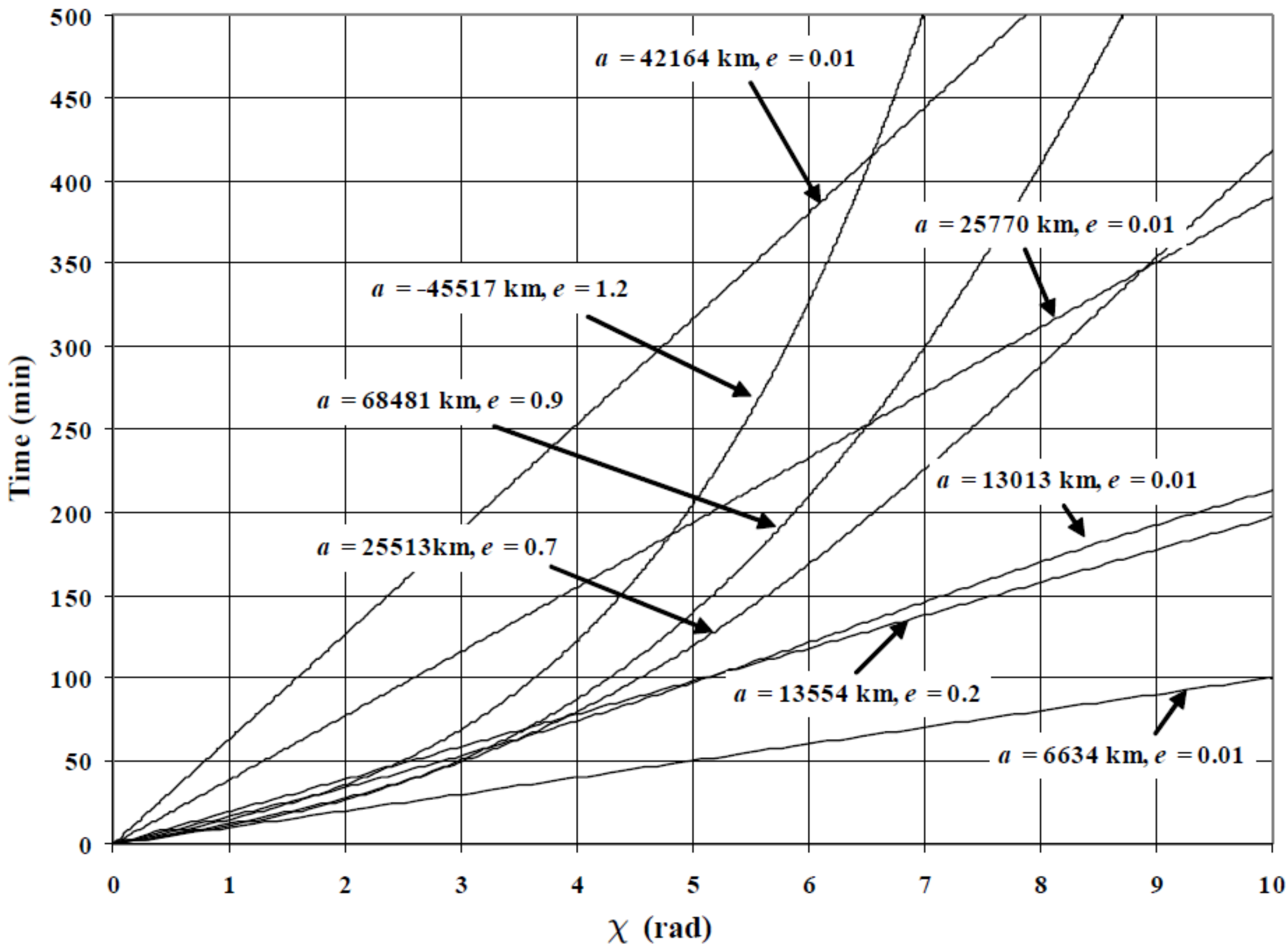


Figure 4.2: Integration *time* versus χ. Elliptical orbits characterized by $e > 0.01$ slightly curve upwards compared to circular orbits, and the slope increases with the eccentricity. This "smooth" nature of the curves suggests that this Newton–Raphson iteration performs much better than regular R2BP integration (from Newton's equation 2.11), for all type of orbits [22].

The universal variables:

$$f_n = 1 - \frac{\chi_n^2}{r_0} c_{2,n} \,, \tag{4.11}$$

$$\dot{f}_n = \frac{\sqrt{\mu}}{r_n \, r_0} \chi_n \left(\psi_n c_{3,n} - 1 \right) \,, \tag{4.12}$$

$$g_n = \Delta t - \frac{\chi_n^3}{\sqrt{\mu}} c_{3,n} \,, \tag{4.13}$$

$$\dot{g}_n = 1 - \frac{\chi_n^2}{r_n} c_{2,n} \,, \tag{4.14}$$

are used to assemble the output state at each time of the propagation in the inertial frame, (after checking that $f_n \dot{g}_n - \dot{f}_n g_n = 1$, which confirms that the angular momentum is non-zero):

$$\underline{\boldsymbol{r}}_n = f_n \underline{\boldsymbol{r}}_0 + g_n \underline{\boldsymbol{v}}_0 \,, \tag{4.15}$$

$$\underline{\boldsymbol{v}}_n = \dot{f}_n \underline{\boldsymbol{r}}_0 + \dot{g}_n \underline{\boldsymbol{v}}_0 \,, \tag{4.16}$$

4.2. The Greenwich sidereal time

As part of the initialization of the propagation, it is required to provide a common epoch date for all the satellites, regardless of the method adopted. For the realistic sets from **CELES**TRAK®, the earliest epoch among the selected TLE dates is used, and this is enough for the embedded ECI and ECEF propagations. On the other hand, in order to obtain the LLA from the ECI orbital state (more efficient procedure than obtaining the ECEF state), there would be a discrepancy if the longitude of Greenwich at the initial epoch date was not adjusted. Therefore the following steps have been implemented in the conversion, to obtain $\lambda_{\mathrm{G}}(t_0)$ of Eq.:2.2. The Julian day number J_0 at 0h UT (Universal Time) [26]:

$$\text{Epoch date} = \left[Y, M, D, h, m, s\right], \tag{4.17}$$

i.e. year, month, day, hour, minute, second, where:

$$\begin{gathered} 1901 \leq Y \leq 2099 \\ 1 \leq M \leq 12 \\ 1 \leq D \leq 31 \end{gathered} \tag{4.18}$$

$$J_0 = 367\,Y - \mathtt{fix}\left(\frac{7}{4}\left(Y + \mathtt{fix}\left(\frac{M+9}{12}\right)\right)\right) + \mathtt{fix}\left(\frac{275\,M}{9}\right) + D + 1,721,013.5 \qquad \text{[days]} \tag{4.19}$$

where $\mathtt{fix}(\cdot)$ means to retain only the integer portion, rounding toward zero. The Julian epoch adopted is the $J2000$, defined at 12h UT on January 1, 2000. The time T_0 between the Julian day J_0 and $J2000$ is, considering that a Julian year is 365.25 days, and JD at $J2000$ epoch is 2,451,545 :

$$T_0 = \frac{J_0 - 2,451,545}{36,525} \qquad \text{[Julian centuries]} \tag{4.20}$$

Then the Greenwich sidereal time θ_{G_0} at 0 h UT:

$$\theta_{\mathrm{G}_0} = \mathtt{mod}\Big(100.4606184 + 36,000.77004\,T_0 + 0.000387933\,T_0^2 - 2.583 \times 10^{-8}\,T_0^3\Big) \qquad \text{[deg]} \tag{4.21}$$

where $\mathtt{mod}(\cdot)$ is the *modulus* operator, so to keep the value in the range $0 \leq \theta_{\mathrm{G}_0} \leq 360\,\text{deg}$. The Greenwich sidereal time $\theta_{\mathrm{G}} = \lambda_{\mathrm{G}}(t_0)$ is found using the UT in hours:

$$UT = h + \frac{m}{60} + \frac{s}{3600} \qquad \text{[hours]} \tag{4.22}$$

$$\theta_{\mathrm{G}} = \mathtt{mod}\left(\theta_{\mathrm{G}_0} + 360.98564724\,\frac{UT}{24}\right) \qquad \text{[deg]} \tag{4.23}$$

where the coefficient of the second term is the number of degrees the Earth rotates in 24 h, i.e. during a solar day [Figure: 2.22].

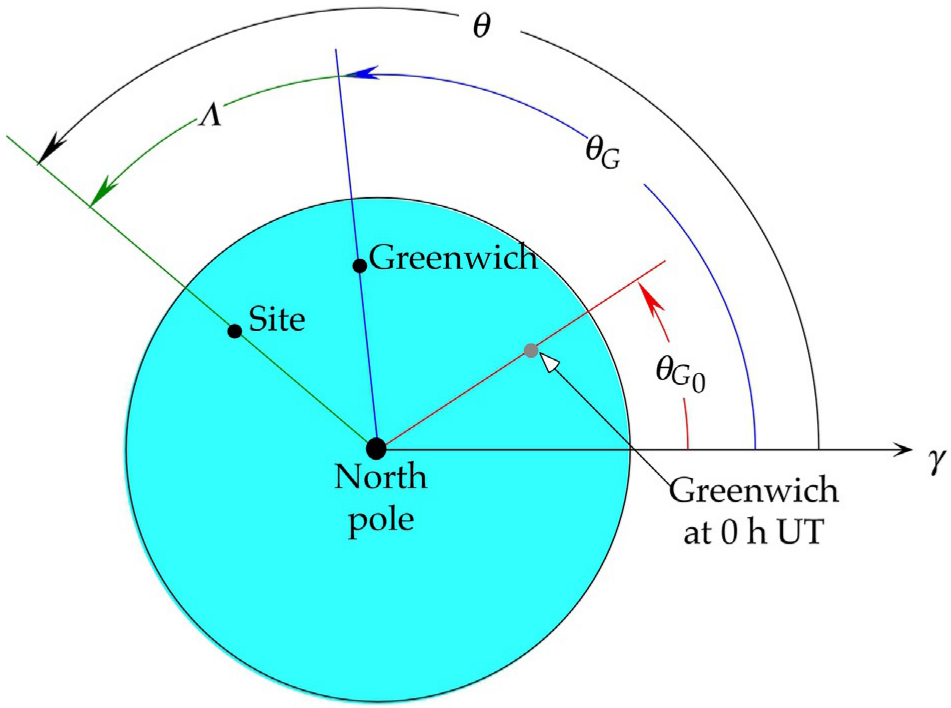


Figure 4.3: Sidereal time schematics: Λ is the east Longitude of the observed spacecraft at current epoch date, θ is the Local Sidereal Time (S.T.), θ_{G_0} is Greenwich meridian S.T. at the Epoch date day, and θ_{G} refers to the position of Greenwich at the any later UT of the day, measured from the vernal equinox direction γ [26].

4.3. Notes and examples of the implementation

Backtracking on the initial Epoch date selected to start the simulation, the `satellite` MATLAB® Aerospace Toolbox function [42] supplies reliable resources to analyze the TLE files, even though it hides some forced procedures that may disrupt the algorithm. It is used to propagate all the satellites from their respective TLE epoch to the earliest date among the epochs. The setup is configured as `2-body-keplerian`, but this is not quite what happens, in fact the SGP4 routine (see 2.2) is performed regardlessly when the keplerian period $T < 225$ minutes, exploiting the first and second *derivatives of mean motion*, along with the `B_star` air drag coefficient from the TLE sets (Figure: 2.21). This may result in a complete change of orbital features if the backward propagation is significantly long, whereas it is negligible if only a few satellites with similar epochs are studied together. In an extreme scenario, the lowest satellites (tested with altitudes $\ll 200$ km) may impact with Earth and completely arrest the script execution, due to the air drag perturbing effects, the implementation of which cannot be prevented. Therefore, the solution adopted is to zero out these three parameters from the TLE files for each satellite, so that the `satellite` function can correctly interpret the data and only perform an actually effective unperturbed 2BP integration. By doing so, the correct initial state of each satellite is obtained at the same date, and subsequently transferred to the Simulink's Keplerian Orbit Propagator block.

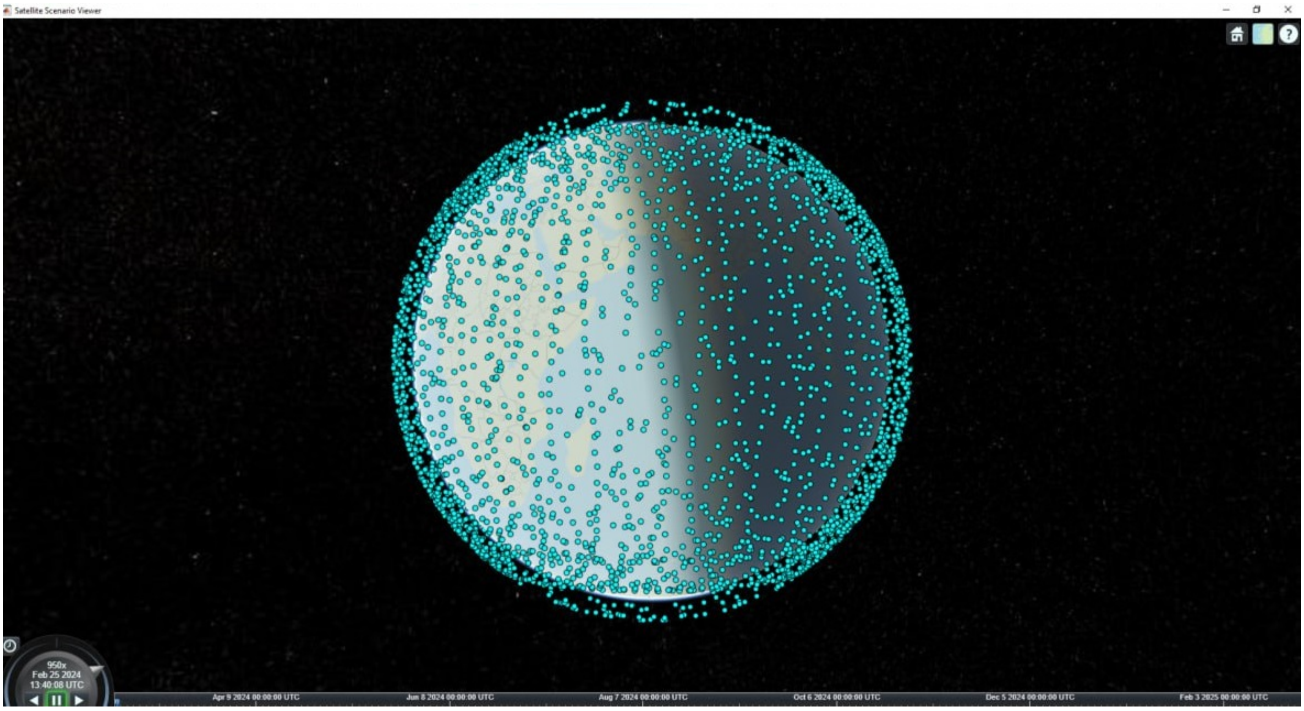


Figure 4.4: `Satellite scenario viewer` (MATLAB® Aerospace Toolbox) displaying positions of $\sim$ 5500 Starlink satellites from the available TLE sets, as of late february 2024, after initialization at the same common epoch. The state of each spacecraft could be propagated by this software in the body of the simulator app, but the lack of computational efficiency makes it unaffordable for the purposes of this work [43].

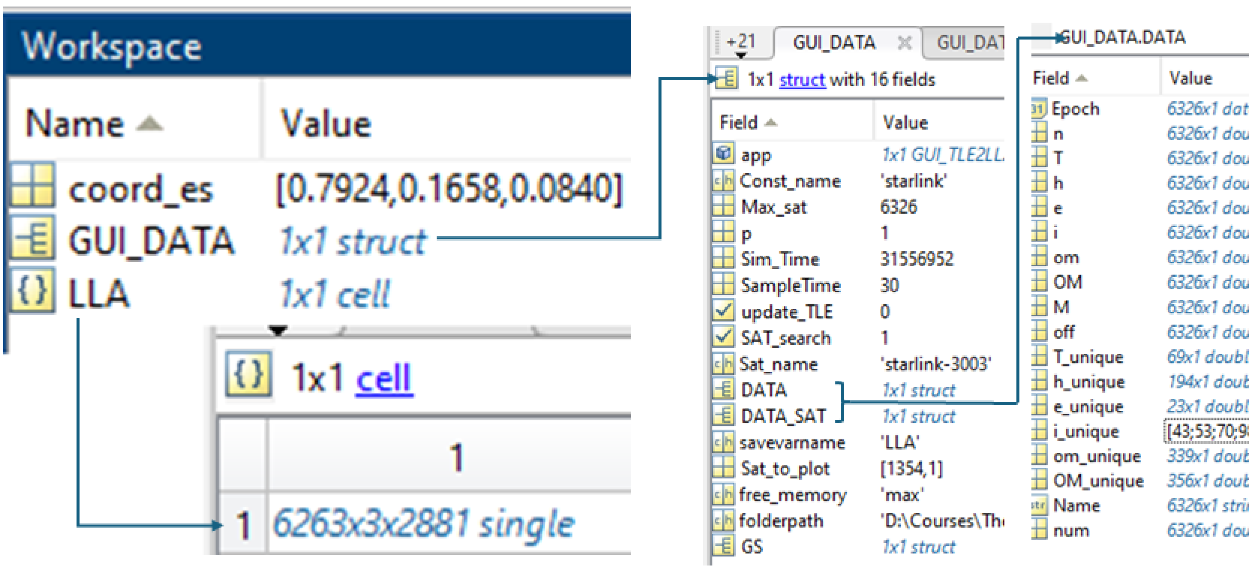
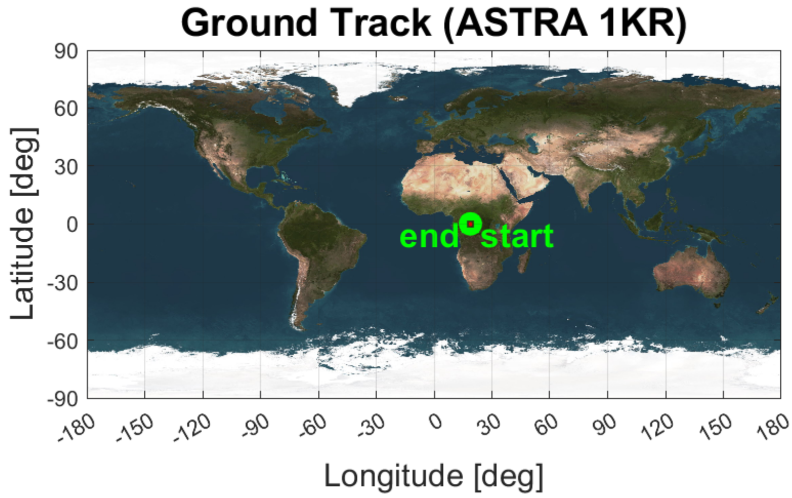
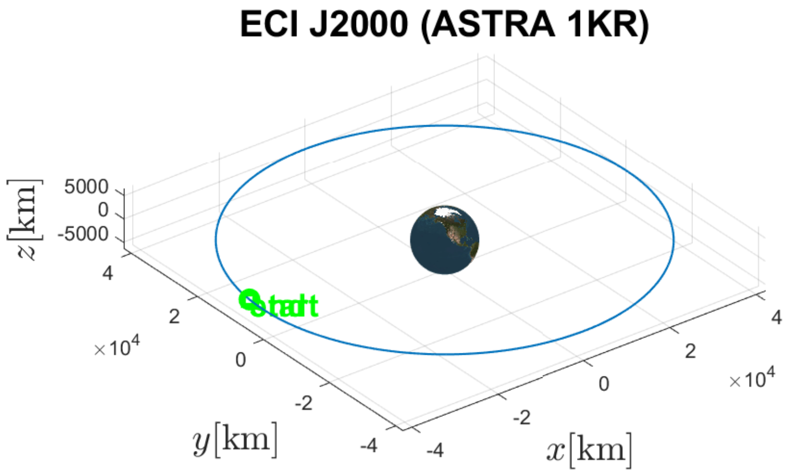


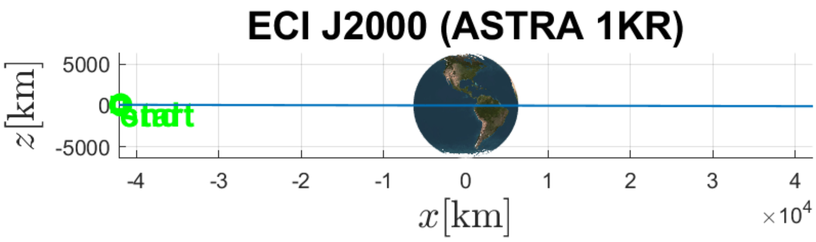
Figure 4.5: An example of the output of the satellite orbit simulator. The LLA file can provide a 3D-array $n \times 3 \times m$, with the three coordinates for n satellites by m time evaluations. The data structure provides information about the propagation settings and the keplerian parameters of each satellite of the chosen constellation, if the TLE search is performed. Unique (rounded) values yield compact information, so to quickly access the variability of the data, before browsing the full set.



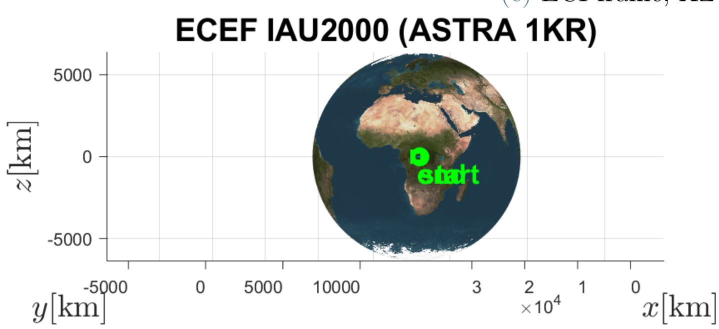
(a) Ground-Track. The GSO satellite remain fixed wrt. the observer from Earth: inclination is zero, so the track does not move along the N-S direction.



(b) ECI frame propagation.

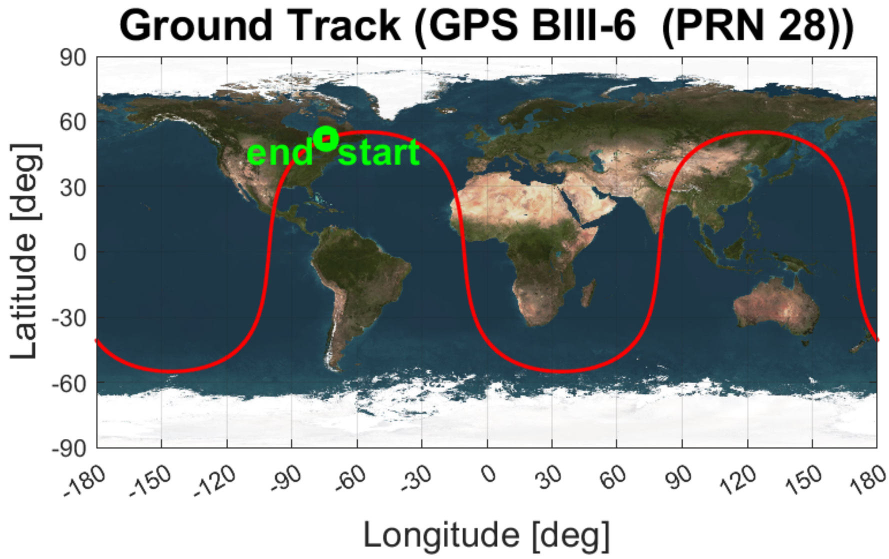


(c) ECI frame, XZ view.

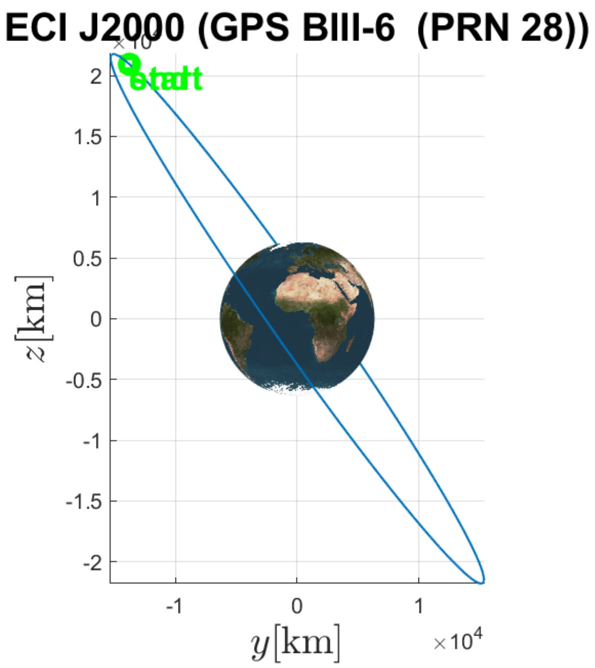


(d) ECEF frame. View from the satellite towards the center of Earth.

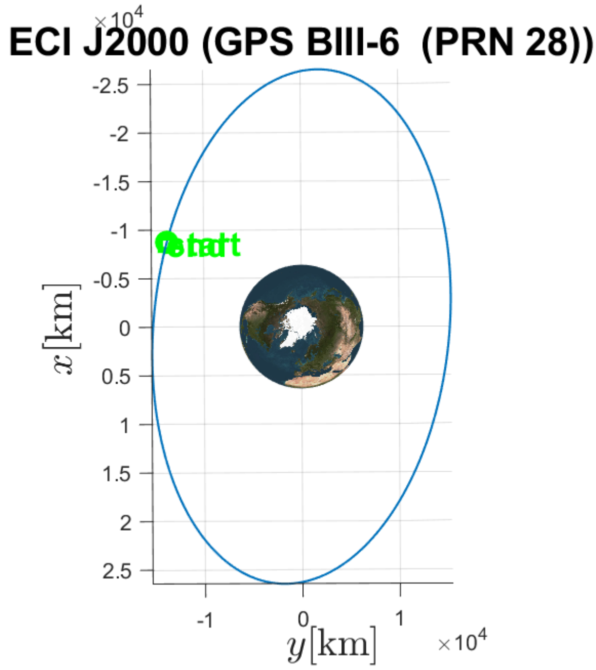
Figure 4.6: GEO orbit propagation example, TLE retrieved from SES satellites page. ECEF can be compared with the Ground-Track to confirm the correct implementation of the procedure.



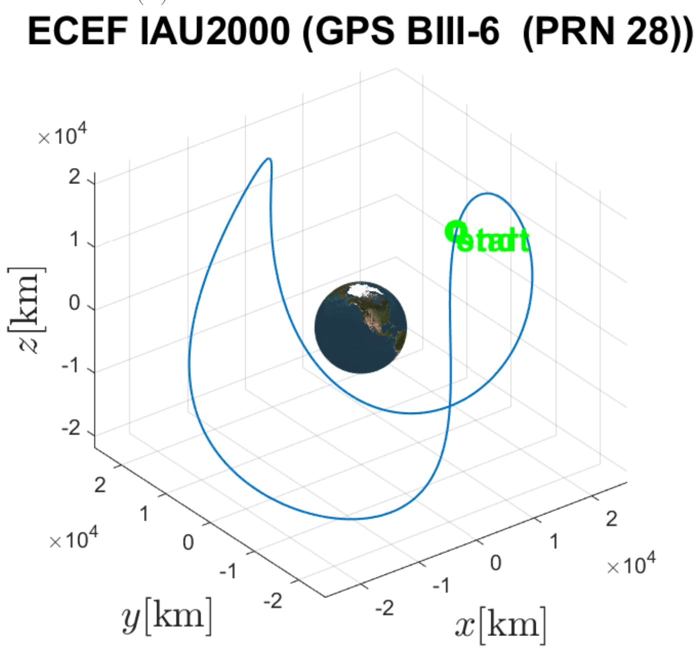
(a) Ground-Track of GPS.



(b) ECI frame orbit: YZ view.

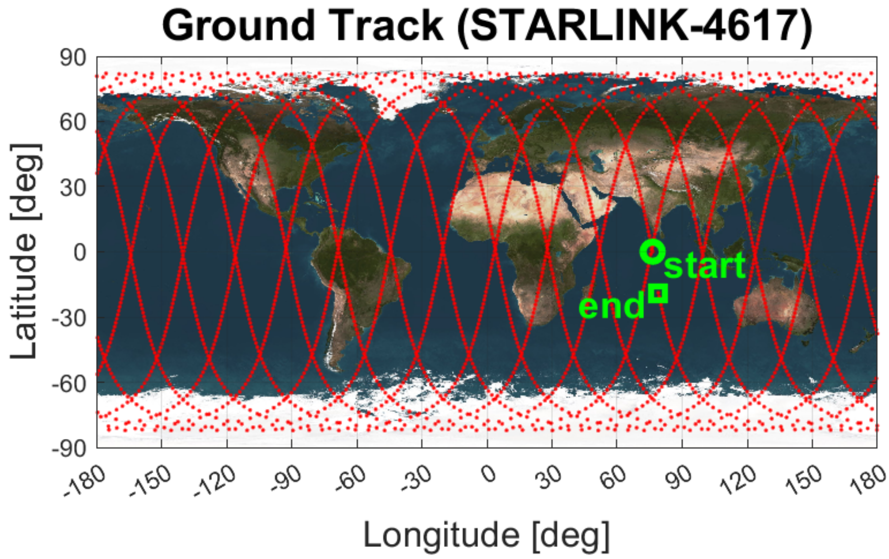


(c) ECI frame orbit: XY view.

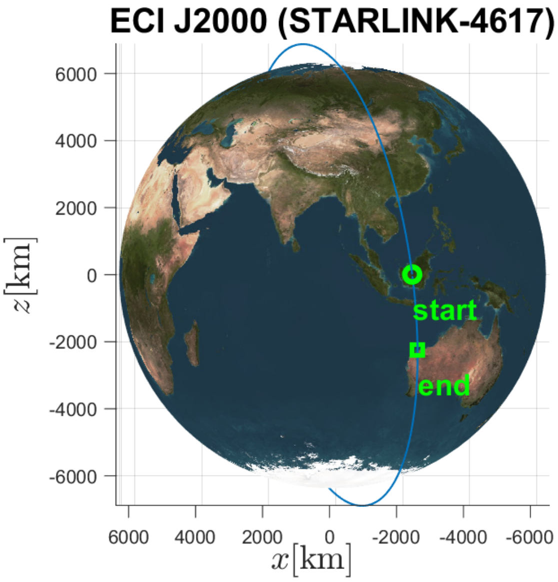


(d) ECEF frame orbit.

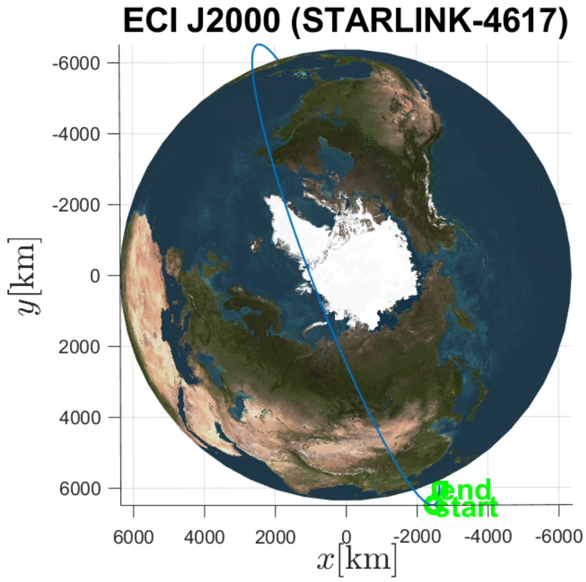
Figure 4.7: GPS orbit propagation, from GPS-OPS database. This circular MEO fleet features 12-h revolution period and of groundtrack repetition time.



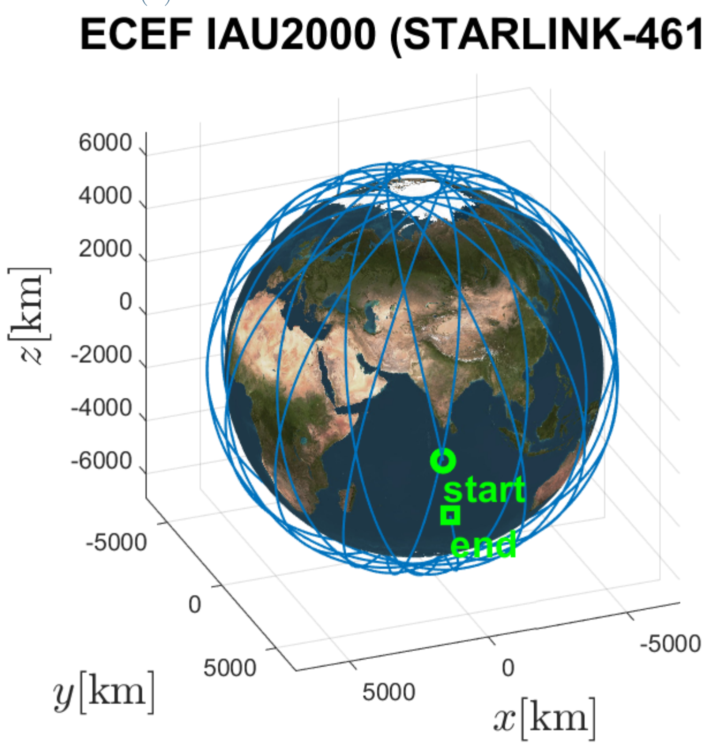
(a) Ground-Track of Starlink polar LEO.



(b) ECI frame orbit: XZ view.



(c) ECI frame orbit: XY view.



(d) ECEF frame orbit.

Figure 4.8: LEO circular orbit propagation, from Starlink database. This orbit is retrograde, i.e. $i > 90$ deg.