SGN – Assignment #2

Daniele Paternoster, 10836125

Exercise 1: Uncertainty propagation

You are asked to analyze the state uncertainty evolution along a transfer trajectory in the Planar Bicircular Restricted Four-Body Problem, obtained as optimal solution of the problem stated in Section 3.1 (Topputo, 2013)*. The mean initial state $\mathbf{x}_i$ at initial time t_i with its associated covariance $\mathbf{P}_0$ and final time t_f for this optimal transfer are provided in Table 1.

1. Propagate the initial mean and covariance within a time grid of 5 equally spaced elements going from t_i to t_f, using both a Linearized Approach (LinCov) and the Unscented Transform (UT). We suggest to use $\alpha = 1$ and $\beta = 2$ for tuning the UT in this case. Plot the mean and the ellipses associated with the position elements of the covariances obtained with the two methods at the final time.
2. Perform the same uncertainty propagation process on the same time grid using a Monte Carlo (MC) simulation †. Compute the sample mean and sample covariance and compare them with the estimates obtained at Point 1). Provide the following outputs.
   - Plot of the propagated samples of the MC simulation, together with the mean and the covariance obtained with all methods in terms of ellipses associated with the position elements at the final time.
   - Plot of the time evolution (for the time grid previously defined) for all three approaches (MC, LinCov, and UT) of $3\sqrt{\max(\lambda_i(P_r))}$ and $3\sqrt{\max(\lambda_i(P_v))}$, where P_r and P_v are the 2x2 position and velocity covariance submatrices.
   - Plot resulting from the use of the MATLAB function `qqplot`, for each component of the previously generated MC samples at the final time.

   Compare the results, in terms of accuracy and precision, and discuss on the validity of the linear and Gaussian assumption for uncertainty propagation.

Table 1: Solution for an Earth-Moon transfer in the rotating frame.

Parameter	Value
Initial state $\mathbf{x}_i$	$\mathbf{r}_i$ = [-0.011965533749906, -0.017025663128129] $\mathbf{v}_i$ = [10.718855256727338, 0.116502348513671]
Initial time t_i	1.282800225339865
Final time t_f	9.595124551366348
Covariance $\mathbf{P}_0$	$\begin{bmatrix} +1.041e-15 & +6.026e-17 & +5.647e-16 & +4.577e-15 \\ +6.026e-17 & +4.287e-18 & +4.312e-17 & +1.855e-16 \\ +5.647e-16 & +4.312e-17 & +4.432e-16 & +1.455e-15 \\ +4.577e-15 & +1.855e-16 & +1.455e-15 & +2.822e-14 \end{bmatrix}$

*F. Topputo, "On optimal two-impulse Earth–Moon transfers in a four-body model", Celestial Mechanics and Dynamical Astronomy, Vol. 117, pp. 279–313, 2013, DOI: 10.1007/s10569-013-9513-8.

†Use at least 1000 samples drawn from the initial covariance

1.1 Uncertainty propagation: LinCov and UT

The propagation of the initial mean and covariance presented in Table 1 is here presented and briefly discussed. Both the UT and the Linearized approach are organized in two different functions which allow to propagate the mean and its uncertainty on a specified time grid. In both cases the numerical propagator for the dynamics is the `ode78`, set with both `RelTol` and `AbsTol` to `1e-12`. As suggested, the Unscented Transform have been tuned with $\alpha = 1$ and $\beta = 2$. The plot of the mean and ellipses of covariances associated with the position at time t_f are here reported. The size of the ellipse is relative to 3σ uncertainty in all cases showed in this exercise.

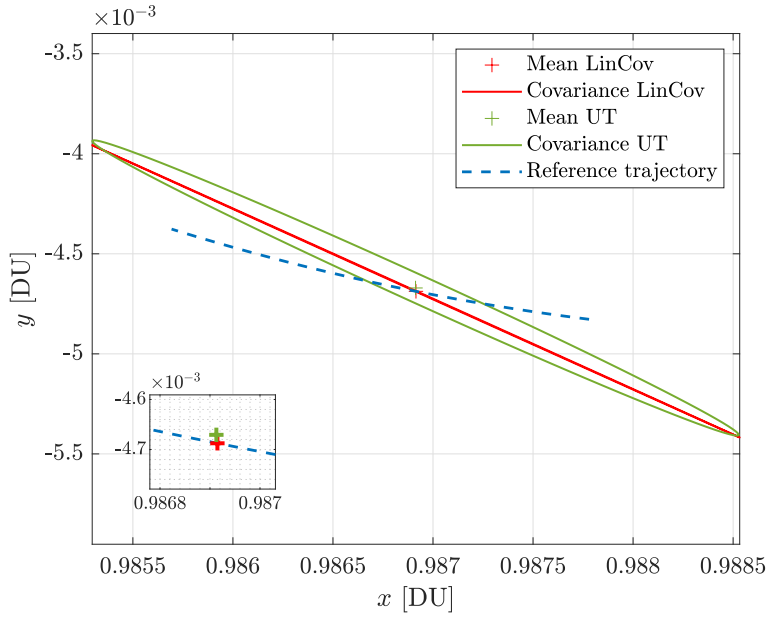

Figure 1: LinCov and UT propgation - EMB-centered rotating frame (adimensional)

Some interesting features can be outlined:

- The reference trajectory of Figure 1 (represented as the dotted blue line) passes through the mean value of the LinCov propagation (see the box at the bottom left of the image). This is expected from the theory since the mean value of the state for the linearized uncertainty propagation *is* the reference trajectory. On the other hand, the mean value at t_f of the UT is slightly off-set from the mean value of LinCov at t_f, this is also coherent with the theory of uncertainty propagation.
- The two ellipses seem to have the same orientation but different centers (as stated in the previous bullet) and clearly different sizes. In particular, the longer dimension seems to be comparable between the two methods, while the smaller dimension is largely different.
- The different performances of the two methods will be evaluated better in the next section, in which the MC simulation will be added. However, an important remark is necessary. From the theory, the UT method can approximate with accuracy up to the third order for Gaussian inputs, and at least to the second-order for non-Gaussian inputs [1]. Thus, it is expected that the covariance (second-order moment) estimated with the UT transform will be more accurate with respect to the one calculated from the LinCov method.

1.2 Uncertainty propagation: a comparison with Monte Carlo simulation

To better evaluate the performance of LinCov and UT uncertainty propagations, a Monte Carlo simulation has been performed. In particular, it is assumed that:

- The distribution of the state at t_i defined in Table 1 is Gaussian.
- The number of samples drawn from that distribution and propagated is $n = 1000$.
- The propagation of the dynamics of the PBRFBP has been performed with the `ode78` numerical integrator with `RelTol` and `AbsTol` set to `1e-12`.

The outcomes of the Monte Carlo simulation in terms of sample mean and sample covariance are represented graphically in Figure 2 and Figure 3. Moreover, the propagated sample points are represented as gray markers.

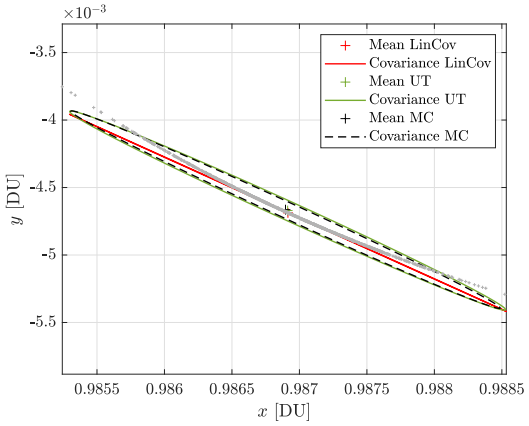

Figure 2: Uncertainty propagated at t_f - EMB-centered rotating frame (adimensional)

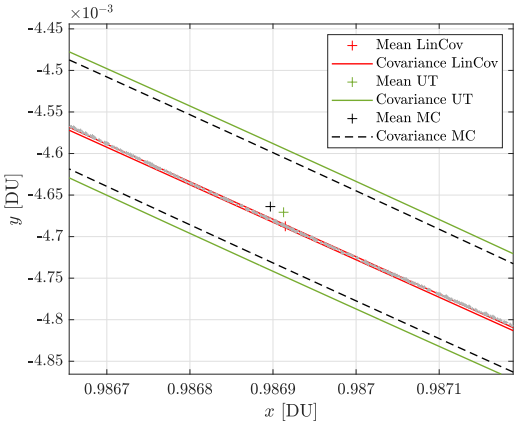

Figure 3: Zoom on propagated mean - EMB-centered rotating frame (adimensional)

Some qualtiative remarks are here mentioned:

- It is clear how the Linearized Method fails to represent the covariance on the position at t_f. This aspect can be qualitatively confirmed by comparison between the covariance ellipses of LinCov and sample covariance of the MC (taken as "truth"). This behavior is expected since the dynamics field is highly non-linear and the propagation time is long.
- From Figure 3 it is also possible to analyze the behaviour of the Linearized Approach near its mean value. The Gaussian approximation seems to work only on some points near the mean value itself which is on the reference trajectory, while looking at the broader picture of Figure 2 it is clear that the Gaussian distribution of the Linearized Method underestimates the more scattered and peculiar distribution of the sample points.
- The true distribution of the propagated sample points coming from the Monte Carlo is the typical "banana-shape", which can be explained by the idea that during the propagation the points will tend to align with the reference trajectory. Due to its complex shape, it is clear that the first and second order moments are incapable of fully representing the pdf of this distribution.
- By looking at Figure 2, it is confirmed the capacity of the UT to catch with great accuracy the first- and second-order moments information, given the Gaussian distribution as input. In fact, the position covariance ellipses are comparable in terms of both orientation and size.

- In conclusion: considering the MC sample mean and sample covariance, it can be stated that the LinCov method is not accurate since it fails at representing the mean value and it is underestimating the dispersion of propagated point due to its narrow covariance. On the other side, the UT method is accurate and precise since its mean value is at least aligned with the direction of maximum uncertainty of the Monte Carlo (on which clearly the sample mean is present) and its sample-weighted covariance is not under estimating the MC sample covariance.

To better evaluate the effect of the uncertainty propagation along time, also the evolution of the estimates of the $3\sigma_r$ and $3\sigma_v$ for the three methods is shown. In particular, the estimate of those quantities is carried out with the *eigenvalues* approach, choosing:

$$3\sigma_r = 3\sqrt{\max\left(\lambda_i(P_r(t_j))\right)} \qquad 3\sigma_v = 3\sqrt{\max\left(\lambda_i(P_v(t_j))\right)} \qquad j = 2, \cdots 5 \tag{1}$$

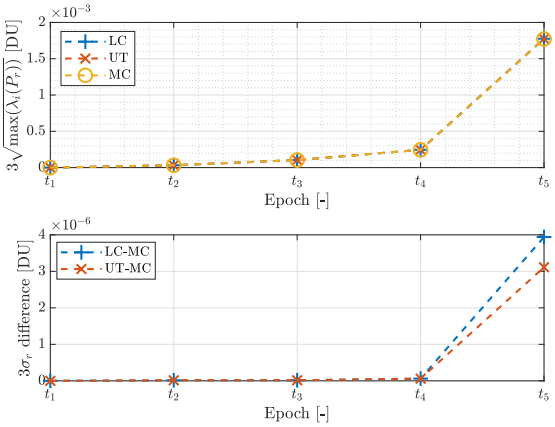

Figure 4: $3\sigma_r$ evolution for the three methods

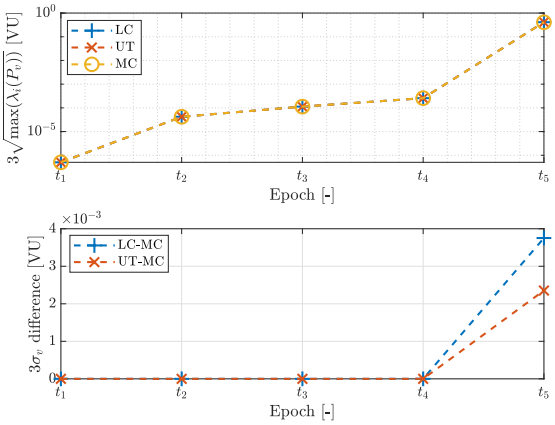

Figure 5: $3\sigma_v$ evolution for the three methods

Some important remarks on Figure 4 and Figure 5 are reported:

- The estimates of the 3σ at each given reference epoch t_j are comparable between the three methods of propagation. This aspect is valid for both position and velocity as seen from Figure 4 and Figure 5, in which also the small difference between methods is represented. In particular, the case of the position uncertainty at the epoch t_f is visible from figure Figure 2. The three ellipses have the largest dimension which is comparable with all the three methods, that specific dimension is associated with the largest eigenvalue of the position covariance sub-matrix.
- The differences on the $3\sigma_r$ and $3\sigma_v$ between the uncertainty propagation methods are more relevant at higher times. This is expected since the time of propagation is a key aspect when considering the quality of an uncertainty propagation estimate. Moreover, the reference trajectory propagated with the given initial condition is a solution of the Moon targeting problem. At times around t_f the solution is nearer to the attractor and the uncertainty has a steeper increase (see appendix).
- In this case, the estimates of the $3\sigma_r$ and $3\sigma_v$ both increase in time if the long term behaviour is considered. However, it is possible that, over shorter periods, the behavior of $3\sigma_r$ and $3\sigma_v$ is not monotonically increasing. Still, it is expected that for long term analysis, the uncertainty grows due to the propagation itself (as reported by both the figures).

- The growth of the uncertainty of the velocity $3\sigma_v$ start much before the growth of the uncertainty of the position $3\sigma_r$. Going from the epoch t_1 to t_2 the value of $3\sigma_r$ is kept at least within the same order of magnitude in adimensional units while for the velocity in the same time step the uncertainty grows of two order of magnitude of VU.

The validity of the Gaussian distribution assumption at time t_f can be considered by the q-q plots for each state component of the generated MC samples. The *qqplot* routine from *Matlab* has been exploited to compare the theoretical normal quantiles versus the sample quantiles of the MC.

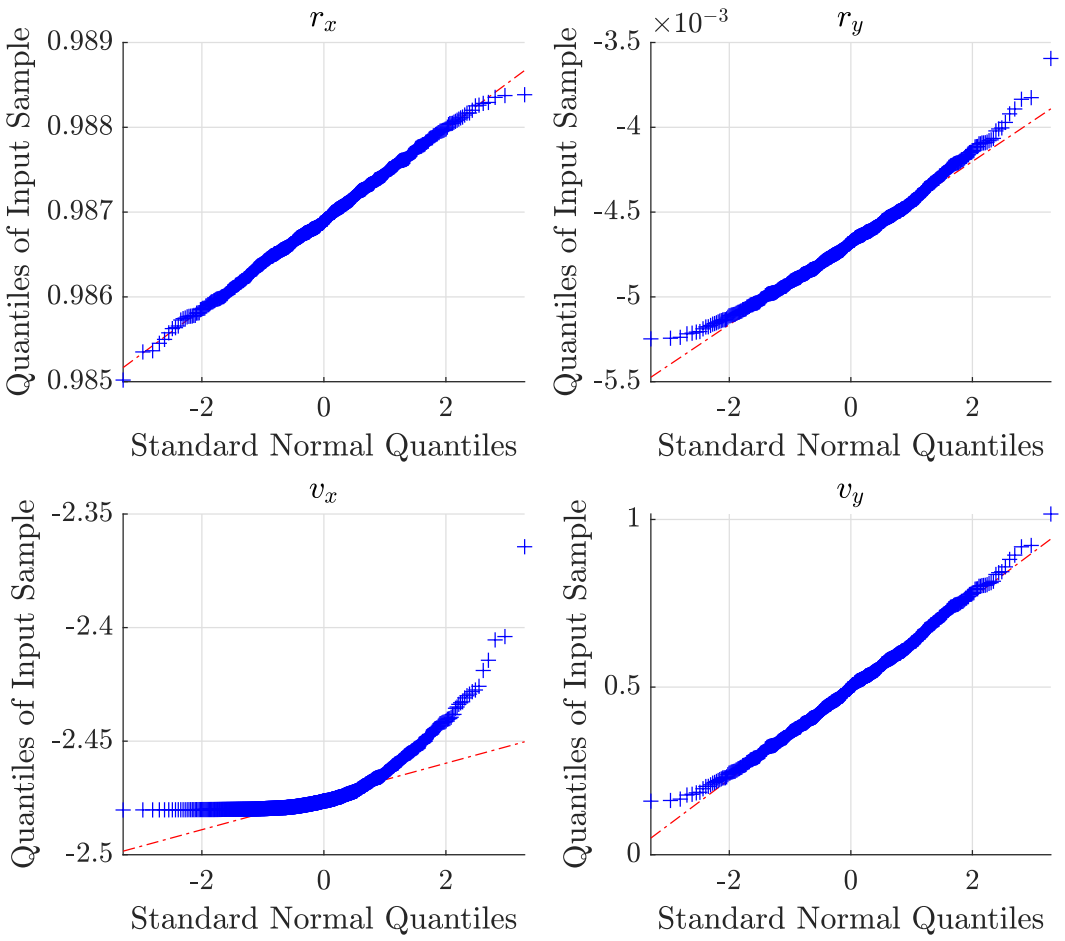

Figure 6: q-q plot - state components MC distribution vs. Normal distribution

From Figure 6 it is clear that the multivariate distribution of the orbital state cannot be claimed to be Gaussian. This is particularly evident from the q-q plot of the v_x and r_y components of the orbital state at t_f. The q-q plots of r_x and v_y seem to be more fitted by a normal distribution, with much less deviations at the edges of the slopes. The upward curvature of both v_x and r_y indicates the asymmetry of the probability distribution function of those components, their deviation with respect to Gaussian distributions is considerable. The results from the q-q plots regarding the distribution of (r_x, r_y) are in line with the qualitative analysis performed in Figure 2 in which the real pdf, which can be though to be represented by the gray markers, is far from being considered Gaussian. Clearly, the assumption that the pdf propagates as a Gaussian distribution (linearized approach) is not valid in this case. On the other side, the best results from the propagation methods in terms of mean and covariances (first order moments) are supplied by the UT, however still incapable of determining the real shape of the pdf.

1.3 Appendix - uncertainty propagation

The values of both $3\sigma_r$ and $3\sigma_v$ calculated as shown before has been plotted for the same interval t_i to t_f but using more elements in the time span, thus allowing to appreciate a less-discretized behavior in time.

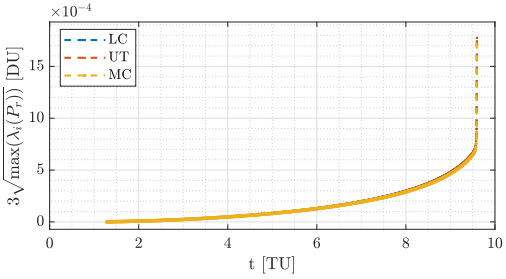

Figure 7: Long term r

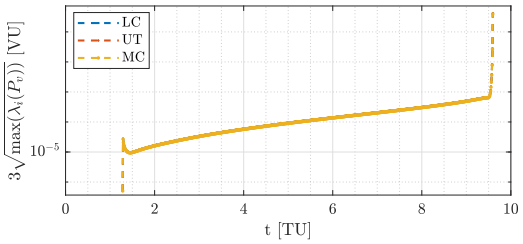

Figure 8: Long term v

The aim of the two plots is to show the finer evolution on the uncertainty of position and velocity. The three propagation methods results in similar values for to the selected metrics (maximum eigenvalue of the correspondent sub matrix). It is clear how the uncertainty both on r and v has a steep growth when near the Moon (at final time). Moreover, from Figure 8 it is clear the non-monotone behavior of the uncertainty in the case of the velocity. It slightly decreases after the departure from the Earth, then it increases until the steep ascent near the moon.

Exercise 2: Batch filters

The Soil Moisture and Ocean Salinity (SMOS) mission, launched on 2 November 2009, is one of the European Space Agency's Earth Explorer missions, which form the science and research element of the Living Planet Programme.

You have been asked to track SMOS to improve the accuracy of its state estimate. To this aim, you shall schedule the observations from the three ground stations reported in Table 2.

1. *Compute visibility windows.* The Two-Line Elements (TLE) set of SMOS are reported in Table 3 (and in WeBeep as 36036.3le). Compute the osculating state from the TLE at the reference epoch t_{ref}, then propagate this state assuming Keplerian motion to predict the trajectory of the satellite and compute all the visibility time windows from the available stations in the time interval from $t_0 =$ `2024-11-18T20:30:00.000` (UTC) to $t_f =$ `2024-11-18T22:15:00.000` (UTC). Consider the different time grid for each station depending on the frequency of measurement acquisition. Report the resulting visibility windows and plot the predicted Azimuth and Elevation profiles within these time intervals.
2. *Simulate measurements.* Use SGP4 and the provided TLE to simulate the measurements acquired by the sensor network in Table 2 by:
   (a) Computing the spacecraft position over the visibility windows identified in Point 1 and deriving the associated expected measurements.
   (b) Simulating the measurements by adding a random error to the expected measurements (assume a Gaussian model to generate the random error, with noise provided in Table 2). Discard any measurements (i.e., after applying the noise) that does not fulfill the visibility condition for the considered station.
3. *Solve the navigation problem.* Using the measurements simulated at the previous point:
   (a) Find the least squares (minimum variance) solution to the navigation problem without a priori information using
      - the epoch t_0 as reference epoch;
      - the reference state as the state derived from the TLE set in Table 3 at the reference epoch;
      - the simulated measurements obtained for the KOROU ground station only;
      - pure Keplerian motion to model the spacecraft dynamics.
   (b) Repeat step 3a by using all simulated measurements from the three ground stations.
   (c) Repeat step 3b by using a J2-perturbed motion to model the spacecraft dynamics.

   Provide the results in terms of navigation solution[‡], square root of the trace of the estimated covariance submatrix of the position elements, square root of the trace of the estimated covariance submatrix of the velocity elements. Finally, considering a linear mapping of the estimated covariance from Cartesian state to Keplerian elements, provide the standard deviation associated to the semimajor axis, and the standard deviation associated to the inclination. Elaborate on the results, comparing the different solutions.
4. *Trade-off analysis.* For specific mission requirements, you are constrained to get a navigation solution within the time interval reported in Point 1. Since the allocation of antenna time has a cost, you are asked to select the passes relying on a budget of 70.000 €. The cost per pass of each ground station is reported in Table 2. Considering this constraint,

[‡]Not just estimated state or covariance

and by using a J2-perturbed motion for your estimation operations, select the best combination of ground stations and passes to track SMOS in terms of resulting standard deviation on semimajor axis and inclination, and elaborate on the results.

5. *Long-term analysis.* Consider a nominal operations scenario (i.e., you are not constrained to provide a navigation solution within a limited amount of time). In this context, or for long-term planning in general, you could still acquire measurements from multiple locations but you are tasked to select a set of prime and backup ground stations. For planning purposes, it is important to have regular passes as this simplifies passes scheduling activities. Considering the need to have *reliable* orbit determination and *repeatable* passes, discuss your choices and compare them with the results of Point 4.

Table 2: Sensor network to track SMOS: list of stations, including their features.

Station name	KOUROU	TROLL	SVALBARD
Coordinates	LAT = 5.25144° LON = -52.80466° ALT = -14.67 m	LAT = -72.011977° LON = 2.536103° ALT = 1298 m	LAT = 78.229772° LON = 15.407786° ALT = 458 m
Type	Radar (monostatic)	Radar (monostatic)	Radar (monostatic)
Measurements type	Az, El [deg] Range (one-way) [km]	Az, El [deg] Range (one-way) [km]	Az, El [deg] Range (one-way) [km]
Measurements noise (diagonal noise matrix R)	$\sigma_{Az,El}$ = 125 mdeg σ_{range} = 0.01 km	$\sigma_{Az,El}$ = 125 mdeg σ_{range} = 0.01 km	$\sigma_{Az,El}$ = 125 mdeg σ_{range} = 0.01 km
Minimum elevation	6 deg	0 deg	8 deg
Measurement frequency	60 s	30 s	60 s
Cost per pass	30.000 €	35.000 €	35.000 €

Table 3: TLE of SMOS.

```
1 36036U 09059A   24323.76060260  .00000600  00000-0  20543-3 0  9995
2 36036  98.4396 148.4689 0001262  95.1025 265.0307 14.39727995790658
```

2.1 Visibility windows

From the TLE of SMOS reported in Table 3, the osculating orbital elements at the reference epoch t_{ref} are retrieved using the *sgp4* folder routines:

a $[km]$	e $[-]$	i $[deg]$	Ω $[deg]$	ω $[deg]$	M $[deg]$
7143.71346	$1.24363 \cdot 10^{-3}$	98.43420	148.46890	69.29293	290.84031

Table 4: Osculating Orbital elements of SMOS @t_{ref}

The steps followed in order to compute the orbital state of SMOS over the time interval prescribed (t_0 to t_f) are the following:

1. The keplerian propagator needs the state expressed in ECI frame, hence the orbital state at t_{ref} has been transformed from TEME frame to ECI frame using the *sgp4* routines.
2. The epochs of interest t_0 and t_f have been converted to ET.
3. The orbital state is propagated from t_{ref} to t_0 with a Keplerian model, all expressed in ECI frame. The numerical propagator used is `ode113` with stringent tolerances (`RelTol` set to `1e-13` and `AbsTol` set to `1e-20`).
4. The highest frequency of acquisition (30s) between the three ground station is considered: a time span between t_0 and t_f with elements spaced of 30.0s is defined.
5. The Keplerian propagation is then performed on the time grid of point 4, with the same numerical propagation used in 3.

The orbital state of SMOS expressed in the ECI frame over the time window of interest (with a frequency of 30.0 seconds between each state) is so obtained. The eventual need of decreasing the acquisition frequency to 60.0 seconds (for KOUROU and SVALBARD) is straightforward. The ideal observation from each ground station is defined as follows:

Antenna data routine:
Input: $\mathbf{r}_{SMOS,ECI}$, t_{span}, GS name.
Loop over the elements of t_{span}:
1. Get $\mathbf{r}_{station,ECI}$ at t_k in ECI (J2000) frame (*spkpos* from spice)
2. Get $\mathbf{r}_{SMOS,ECI} - \mathbf{r}_{station,ECI} = \boldsymbol{\rho}_{ECI}$
3. Get transformation matrix $\boldsymbol{R}$ from ECI to TOPOCENTRIC frame centered at the GS (*pxform* from spice)
4. Get $\boldsymbol{\rho}_{topo} = \boldsymbol{R}\,\boldsymbol{\rho}_{ECI}$
5. Get Az, El, ρ from the topocentric coordinate ρ_{topo} (*reclat* from spice).

Recalling that the TOPOCENTRIC frame of the GS has been defined having the x axis pointing North and the y axis pointing west. The Az angle is calculated as te angle from the (x, z) plane, positive around +z.

This routine has been implemented into a *Matlab* function and called for all the ground stations of interest. In particular, for Svalbard and Kourou, the elements have been selected by taking and omitting the next in a recurring sequence in order to fit the lower acquisition frequency of 60.0 seconds. The function *antenna data* just presented in the box have been validated with the private function *antenna pointing* supplied during Lab 4. The ideal observations are then obtained by applying the visibility constraint expressed by the minimum elevation angle, expressed for the different ground stations in Table 2. The found visibility windows are the following:

GS name		**Calendar epoch** (UTC)
KOUROU	t_i	2024-11-18T20:40:00.000
	t_f	2024-11-18T20:49:00.000
TROLL	t_i	2024-11-18T21:02:30.000
	t_f	2024-11-18T21:11:30.000
SVALBARD	t_i	2024-11-18T21:56:00.000
	t_f	2024-11-18T22:06:00.000

Table 5: Ideal visibility windows for the GSs

The Azimuth and Elevation profiles for the three ground stations are:

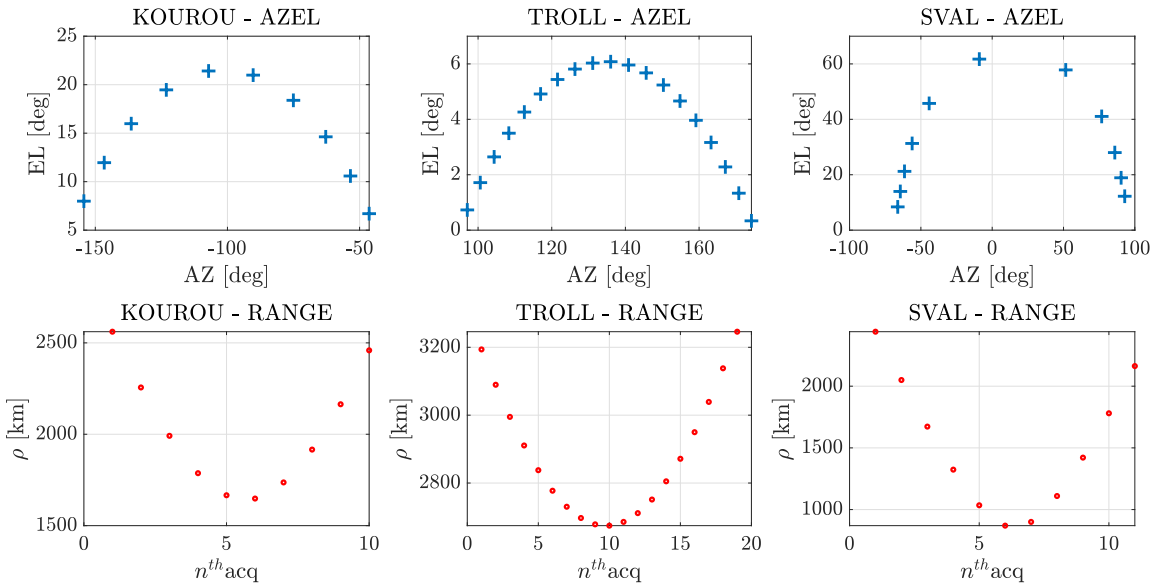

Figure 9: AZEL profile inside the visibility window

- Regarding Figure 9, the values of the azimuth are read as decreasing for Kourou, while for Troll and Svalbard they are increasing.
- From the plots it is noticeable the greater amount of data taken by Troll due to its higher acquisition frequency.
- As one could expect, the AZEL profiles have the typical bell shape due to the fact that the visibility starts from lower values of elevation reaching a peak and then decreasing.
- The horizontal spacing between + markers is minimum at the sides of the AZEL plots while it is larger at the top of the bells. This can be particularly appreciated in the profiles of Kourou and Svalbard. The effect is due to the relative motion between the GS and S/C. At the maximum elevation the S/C is at lower range, hence it is seen faster from the observer and a constant acquisition frequency will be capable of registering less data. The fact that this effect is more visible on Svalbard is due to its lower range to the S/C with respect to the other two stations.

2.2 Simulation of measurements

In order to simulate the measurements taken by the network of ground stations defined in Table 2, a new *Matlab* routine has been developed. The logical steps performed to simulate the measurements of AZEL and range are here reported:

Antenna measurements routine:
Input: $t_{refSMOS}$, t_{vw}, SMOS record, GS struct, nutation parameters
Loop over the elements of t_{vw}:

1. Propagate SMOS motion with *sgp4* from t_{ref} to $t_{vw}(k)$ (*sgp4* routine needed)
2. Convert orbital state at $t_{vw}(k)$ from TEME to ECI (*teme2eci* routine needed)
3. Get $\mathbf{r}_{station,ECI}$ at $t_{vw}(k)$ in ECI (J2000) frame (*spkpos* from spice)
4. Get $\mathbf{r}_{SMOS,ECI} - \mathbf{r}_{station,ECI} = \boldsymbol{\rho}_{ECI}$
5. Get transformation matrix $\boldsymbol{R}$ from ECI to TOPOCENTRIC frame centered at the GS (*pxform* from spice)
6. Get $\boldsymbol{\rho}_{topo} = \boldsymbol{R}\boldsymbol{\rho}_{ECI}$
7. Get μ_{Az},μ_{EL} and μ_{ρ} from the topocentric coordinate ρ_{topo} (*reclat* from spice).
8. Add noise to the mean values of point 7 using *mvnrnd* routine of Matlab.
9. Apply constraint on minimum elevation El_{min}

The *sgp4* model is used in order to simulate the measurements mean value since it contains more perturbations with respect to the Keplerian propagation. Moreover, the validity of the TLE is respected since the propagation of the *sgp4* is far less than 12 hours with respect to the reference epoch t_{ref}. With this approach, it can happen that the last measurement of Troll station sometime is excluded. This is due to the fact that the mean value is slightly lower zero and depending on the run the measured value with noise can be slightly positive and hence enter in the measurement batch. The obtained measurements are here reported graphically:

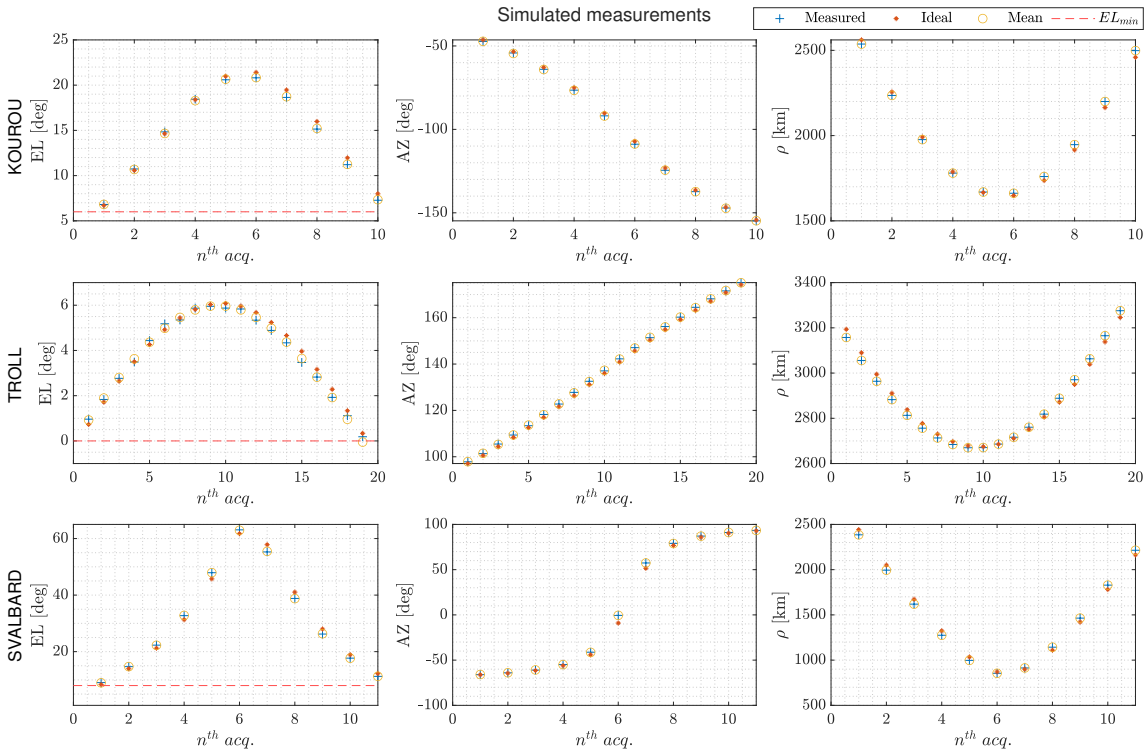

Figure 10: Simulated measurements for all GS

In this run it is evident the exclusion of the last measurements from the Troll station, its mean

is slightly lower than the minimum elevation but even with the random noise the measurement is below the threshold of 0 degrees. In this case, the visibility window of Troll is slightly decreased. The visibility window for Svalbard and Kourou is instead unchanged with respect to what calculated before the simulation.

2.3 Solution of the navigation problem

In this section the simulated measurements coming from the model described in 2.2 are used in the framework of a minimum-variance filter in order to supply an estimate of the S/C state with its covariance expressed both in ECI frame at the epoch t_0. The estimate of the state and its uncertainty at that epoch are then mapped into an uncertainty on semi-major axis and inclination by means of a linearized transformation. The setup of the filter in *Matlab* is briefly recapped in 2.3.1.

2.3.1 Least-squares (minimum variance) filter without a-priori information

The navigation problem requires to find an estimate of the S/C state $\mathbf{x}$ at a certain epoch t_0 given some measurements $\mathbf{Y}_i$. In order to setup properly the filter, the following assumptions are made:

- **Dynamic model:** it this scenario, two models are considered. In particular, a purely keplerian motion around the Earth with fixed parameters or J2-perturbed keplerian motion around the Earth with fixed parameters.

$$\begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ -\mu \dfrac{\mathbf{r}}{r^3} \end{bmatrix} + \mathbf{a}_{J2}$$

- **Measurement model:** the ideal measurements of Az, El and ρ as seen from a particular GS (or a combination of GSs). The measurement equation is implemented in *Matlab* through the already presented routine *Antenna data* described in 2.1

- **Filtering technique:** in this case a batch, minimum-variance filter has been implemented. In particular, the minimum variance condition has been implemented through the formalism of the weighted least squares by choosing the adequate weight matrix. Moreover, the filter has been setup in a non-linear context. In this case, the i-th non-weighted residual is defined as (corresponding to the i-th measure):

$$\boldsymbol{\epsilon}_i = \mathbf{y_{meas,i}} - \mathbf{y_{id}}(\mathbf{x}_i) \tag{2}$$

$$\mathbf{x}_i = \boldsymbol{\phi}(\mathbf{x}_0, t_o; t_i) \tag{3}$$

Without the need to linearize the measurement equations. The total cost function J to be minimized is:

$$J(\mathbf{x}_0) = \sum_{i}^{N_{meas}} \epsilon_i^T \boldsymbol{W} \epsilon_i \tag{4}$$

Where the weigth matrix $\boldsymbol{W}$ is defined as:

$$\boldsymbol{W} = \begin{bmatrix} \dfrac{1}{\sigma_{Az}^2} & 0 & 0 \\ 0 & \dfrac{1}{\sigma_{El}^2} & 0 \\ 0 & 0 & \dfrac{1}{\sigma_{\rho}^2} \end{bmatrix} \tag{5}$$

The σ values describing the noise of the measurements are reported in Table 2.

The cost function J is defined in the *Matlab* environment, then the *lsqnonlin* routine is used to find the state $\mathbf{x_0}$ which minimizes the cost function J. A brief pseudo-code shows the logical steps implemented to calculate the cost function feed in the *lsqnonlin* solver.

Algorithm 1 Cost function $J(\mathbf{x}_0)$

Require: $\mathbf{x}_0$, t_0, $\boldsymbol{Y}$, $\boldsymbol{W}$, t_{meas}, GS name
1: **for** `each i-th measurement epoch` **do**
2: **Propagate** $\mathbf{x}_0$ from t_0 to t_i (measurement epoch) $\rightarrow \mathbf{x}_i$ ▷ Keplerian or J2-Keplerian
3: **Calculate** $\mathbf{y}_{id}$ at epoch t_i using $\mathbf{x}_i$ ▷ Call to *antenna data* routine
4: **Calculate** weight residual $\boldsymbol{\epsilon}_i = \sqrt{\boldsymbol{W}}(\boldsymbol{Y}_i - \mathbf{y}_{id})$
5: **Stack** the residuals $\boldsymbol{\epsilon} = \begin{bmatrix} \boldsymbol{\epsilon} \\ \boldsymbol{\epsilon}_i \end{bmatrix}$
6: **end for**
7: **Output:** $\boldsymbol{\epsilon}$

- The numerical propagator at line 2 is the `ode113` with `RelTol` set to `1e-12` and `AbsTol` set to `1e-20`, with units of km and km/s for $\mathbf{r}$ and $\mathbf{v}$. The *matlab* implementation contains a flag used to call the correct right hand side for the dynamical model considered.
- The *lsqnonlin* routine is used with the `levenberg-marquardt` algorithm. The definition of the cost function is performed in Algorithm 1 as requested by *lsqnonlin*.
- The `central` finite difference scheme has been used, favoring precision to computational effort.
- The generalization of the filter applied to larger batch of measurements (coming from other ground stations), is straight forward. The case of multiple GSs involved in the batch, requires to call the cost function of Algorithm 1 as many times as many GSs are involved and then vectorize the cost function in order to incorporate the residuals calculated on all the measurements from all stations.
- The initial guess for $\mathbf{x}_0$, to be feed into the solver for each run, is the propagated state coming from the TLE, from epoch t_{ref} to t_0 with a purely Keplerian dynamical model.

The solution of the *lsqnonlin* algorithm is the estimated state $\mathbf{x}_0$ expressed in the Earth-Centered J2000 reference frame at the epoch t_0. The covariance $\boldsymbol{P}_{car}$ of this mean value can be estimated as:

$$\boldsymbol{P}_{car} = \frac{\epsilon^*}{m-4}\left(J^t J\right)^{-1} \tag{6}$$

It expresses the uncertainty in the Earth-Centered J2000 reference frame on the estimated state $\mathbf{x}_0$ in terms of position and velocity at the epoch t_0.

2.3.2 Cartesian to Keplerian

The transformation from cartesian state to Keplerian parameters is non-linear, however a linearization around the mean value is possible in order to map the uncertainty from the cartesian state to the semi-major axis a and inclination i.

$$\begin{bmatrix} \mathbf{r} \\ \mathbf{v} \end{bmatrix} \rightarrow \mathbf{f} = \begin{bmatrix} a \\ i \end{bmatrix} = \begin{bmatrix} -\dfrac{\mu r}{r v^2 - 2\mu} \\ \arccos \dfrac{(\mathbf{r}\times\mathbf{v})_z}{|\mathbf{r}\times\mathbf{v}|} \end{bmatrix} \tag{7}$$

The Jacobian of $\mathbf{f}$ can be written as:

$$\nabla \mathbf{f} = \begin{bmatrix} \frac{2\mu^2}{(rv^2-2\mu)^2}\frac{\mathbf{r}}{r} & \frac{2\mu r^2}{(rv^2-2\mu)^2}\mathbf{v} \\ \frac{\partial i}{\partial \mathbf{r}} & \frac{\partial i}{\partial \mathbf{v}} \end{bmatrix} \tag{8}$$

Where (see appendix for details on derivatives):

$$\frac{\partial i}{\partial \mathbf{x}} = \frac{\partial i}{\partial h}\frac{\partial h}{\partial \mathbf{x}} + \frac{\partial i}{\partial h_z}\frac{\partial h_z}{\partial \mathbf{x}} = \frac{h_z^2}{h^2\sqrt{1-(h_z/h)^2}} \begin{bmatrix} \frac{\mathbf{v}\times\mathbf{h}}{h} \\ -\frac{\mathbf{r}\times\mathbf{h}}{h} \end{bmatrix} - \frac{1}{h\sqrt{1-(h_z/h)^2}} \begin{bmatrix} v_y \\ -v_x \\ 0 \\ -y \\ x \\ 0 \end{bmatrix} = \cdots \tag{9}$$

$$= \frac{1}{h\sqrt{1-(h_z/h)^2}} \left[(h_z/h)^2 \begin{bmatrix} \frac{\mathbf{v}\times\mathbf{h}}{h} \\ -\frac{\mathbf{r}\times\mathbf{h}}{h} \end{bmatrix} - \begin{bmatrix} v_y \\ -v_x \\ 0 \\ -y \\ x \\ 0 \end{bmatrix} \right] \tag{10}$$

The covariance in terms of (a, i) is then obtained as:

$$P_{kep} = \nabla \mathbf{f} P_{car} \nabla \mathbf{f}^T \tag{11}$$

2.3.3 Numerical results

The numerical results in terms of navigation solution obtained through the implementation of the non-linear least squares, minimum variance filter are here reported for the prescribed combination of measurements and dynamic model. The estimated state $\mathbf{x}_0$ is reported in the Earth centered J2000 reference frame at the epoch t_0, while for representation purposes the state-associated full covariance P_{car} is represented in Appendix 2.6.2. In order to give an idea of the estimated uncertainties found within the filtering process, the standard deviation are reported for each case. In particular, the most relevant information coming from the estimated covariances P_{car} has been shrinked into two parameters calculated as:

$$\sigma_r = \sqrt{\mathrm{Tr}\,(P_r)} \quad [km] \qquad \sigma_v = \sqrt{\mathrm{Tr}\,(P_v)} \quad [km/s]$$

Kourou GS with Keplerian motion

Parameter	Value
Estimated state $\mathbf{x}_0$	$\mathbf{r}_0$ = [3933.265, -1417.104, 5780.823] [km]
	$\mathbf{v}_0$ = [4.88024, -3.76091, -4.23828] [km/s]
Estimated σ_r and σ_v	σ_r = 12.1404 [km]
	σ_v = 1.1549 $\cdot 10^{-2}$ [km/s]

Table 6: Estimated $\mathbf{x}_0$ at t_0 in Earth-Centered J2000 frame - Kourou & pure Keplerian motion

All ground stations with Keplerian motion

Parameter	Value
Estimated state $\mathbf{x}_0$	$\mathbf{r}_0$ = [3926.928, -1411.064,5780.009] [km]
	$\mathbf{v}_0$ = [4.88852, -3.76497, -4.23340] [km/s]
Estimated σ_r and σ_v	σ_r = 1.0535 [km]
	σ_v = 1.0789 $\cdot 10^{-3}$ [km/s]

Table 7: Estimated $\mathbf{x}_0$ at t_0 in Earth-Centered J2000 frame - All GSs & pure Keplerian motion

All ground stations with J2 perturbation

Parameter	Value
Estimated state $\mathbf{x}_0$	$\mathbf{r}_0$ = [3932.753, -1414.906, 5778.500] [km]
	$\mathbf{v}_0$ = [4.87983, -3.76322, -4.23288] [km/s]
Estimated σ_r and σ_v	σ_r = 2.7681 $\cdot 10^{-2}$ [km]
	σ_v = 2.8290 $\cdot 10^{-5}$ [km/s]

Table 8: Estimated $\mathbf{x}_0$ at t_0 in Earth-Centered J2000 frame - All GSs & Keplerian wJ2 perturbed

To better evaluate the performance of the filter on the different scenarios, the just shown estimated standard deviations are reported in a more compact table. Moreover, the full uncertainty on the cartesian state is transformed into uncertainties on both semi-major axis and inclination by means of the linearized transformation described in 2.3.2. In particular, from the covariance of the two elements (a, i), the σ values are extracted from its diagonal by taking the square root. This four parameters allow to condense the most relevant information of the 6×6 estimated covariances P_{car}. The analysis of those parameters allows a comparison between the different proposed scenarios.

Case	$\sqrt{\text{Tr}\,\boldsymbol{P_r}}$ [km]	$\sqrt{\text{Tr}\,\boldsymbol{P_v}}$ [km/s]	σ_a [km]	σ_i [deg]
Kourou-Kep	12.1404	1.1549 $\cdot 10^{-2}$	7.8215	1.2049 $\cdot 10^{-1}$
All-Kep	1.0535	1.0789 $\cdot 10^{-3}$	1.4183 $\cdot 10^{-1}$	2.0551 $\cdot 10^{-3}$
All-KepJ2	2.7681 $\cdot 10^{-2}$	2.8290 $\cdot 10^{-5}$	3.7951 $\cdot 10^{-3}$	5.4022 $\cdot 10^{-5}$

Table 9: Trace and keplerian

Some remarks on the results of Table 9 are reported:

- The measurements acquired by each GSs have the same noise, hence the quality of the observed quantities is similar. However, the concept of observability can lead to have better or worse estimates even with this equal-noise feature. This aspect will be more clear when the trade-off analysis will be carried out and the analysis on GSs couples will be presented.

- It is clear how the estimated uncertainty decreases in all four representations when all stations are considered (compare first and second rows of the table). This is due to the increased number of measurement considered within the estimation process. Going from Kourou measurement to all stations measurements, a gain of around one order of magnitude on position and velocity uncertainty is acquired by adding measurements while the decrease of uncertainty on the two orbital parameters is around two orders of magnitude.
- The second interesting comparison is between the last two rows of Table 9, since those cases have the same batch of measurements. All the uncertainties are decreased by two orders of magnitude by changing the propagation model from Keplerian to J2-perturbed Keplerian. This is a clear advantage and it is due to the more accurate dynamical model used by the filter. This result is expected: since the *sgp4* propagation model used to generate the measurements contains the J2-perturbation effect, a better match is obtained when the filter dynamical model contains more similar features to the measurements nature. This is true in general applications of the filtering process, a model which catches more details clearly allows to achieve more precise estimations.

2.4 Trade-off

This section presents the trade-off analysis carried out in order to obtain a navigation solution for the time interval t_0 to t_f considering a budget of 70 k€. It is assumed that:

- A dynamical model containing the J2 perturbation is used.
- The results are compared in terms of σ_a [km] and σ_i [deg], by means of the linearized transformation seen in 2.3.2.

From the cost constraints it is clear that only couples of GSs (or single GS) of Table 2 can be considered. To carry out a complete analysis, the single GSs and all the possible couples of GSs are issued. Moreover, the result coming from all the ground stations with J2 perturbed motion seen in 2.3.3 is reported for comparison.

Case	σ_a [km]	σ_i [deg]	Cost [k€]
Kourou	7.1067	$9.3157 \cdot 10^{-2}$	30
Troll	7.9874	$1.9868 \cdot 10^{-2}$	35
Svalbard	1.6284	$1.1715 \cdot 10^{-2}$	35
Kou-Sval	$1.2558 \cdot 10^{-2}$	$4.8833 \cdot 10^{-4}$	65
Sval-Troll	$4.5286 \cdot 10^{-2}$	$4.9561 \cdot 10^{-4}$	70
Kou-Troll	1.3832	$2.8753 \cdot 10^{-3}$	65
All	$3.7951 \cdot 10^{-3}$	$5.4022 \cdot 10^{-5}$	100

Table 10: Trade-off table

- In this case, the best performance in terms of reduction on uncertainties which respects the cost constraint is obtained by the Kourou/Svalbard couple of ground stations. Similar performances are achieved with the Svalbard/Troll ground stations, however with a higher price.
- It is interesting to note how Troll station alone, which supplies double the measurements of Kourou or Svalbard, has the largest uncertainty both on σ_a and σ_i. This aspect could be justified by the fact that even though the measurements from Troll are numerous, they are not as significative as the ones taken from Svalbard. From the representation of the simulated measurements in Figure 10 it is clear that the most wide range of values for Az, El and ρ are covered from the Svalbard station, this helps the filter to gain much more

information and eventually obtain a lower uncertainty on the estimated state. In other terms, the observability of the state seems to be lower from Kourou and Troll batch of measurements, while it is higher in the case of the Svalbard batch of measurements.

- The combination of Svalbard measurements with another GS (within the budget cap) leads to the lowest results in terms of uncertainties due to the favorable condition of the collected measurement. In fact, as said before the performance of Troll/Svalbard and Kourou/Svalbard are comparable.
- The best performance is clearly obtained from the collection of all the measurements (last row of Table 10). This scenario is excluded due to budget limitation. As expected, it produces the best results since the noise level of the measurements are the same (Table 2) and moreover different values of AZEL and range are taken from three different locations on the Earth thus making the overall batch of measurement more rich of information.

2.5 Long term analysis

In this section a long term analysis for the coverage of SMOS is considered. In particular, the outcomes of this analysis are the selection of a prime and a backup station for the observation of SMOS. In particular, the continuous coverage is not requested however the satellite should be observed in a regular and periodic fashion in order to ensure reliable and repeatable orbit determination.

The dynamical model used to perform the long term analysis is the Keplerian motion with the J2 perturbation, even though for this analysis is not so relevant to model perturbations. The propagated trajectory has been feed in the *antenna data* routine which simulates ideal observation from a peculiar ground station given a certain trajectory in the ECI frame. The period of propagation has been set to three days, which allows to catch some of the repetition patterns of the ideal observation angles. The most relevant quantity that defines the visibility of an object from a fixed ground station is the elevation El, which has been plotted for the three cases also with its minimum value (from Table 2):

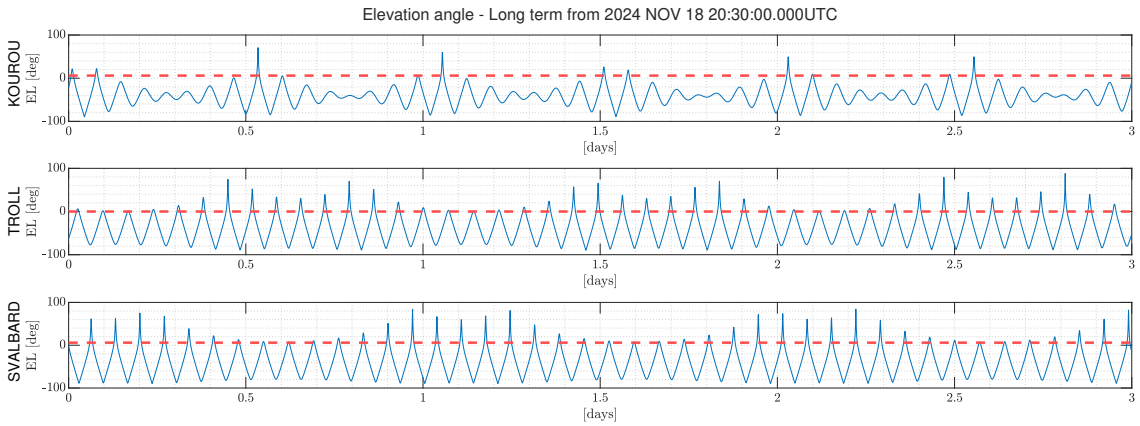

Figure 11: Long term: Elevation - 3 days propagation from 2024 NOV 18 20:30:00.000 UTC

- It is interesting to compare the long term analysis of Figure 11 with the visibility window obtained for the restricted period of the previous subsection 2.4 (short term trade-off). As seen from the elevation profile of Kourou in Figure 11, the presence of a visibility window for Kourou in a 2 hour span is not so common. This lower chance of observation is expected for the Kourou site since its location is approximately on the equator (see Table 2), while the orbit of SMOS is very inclined (almost polar) as seen from Table 4. In terms of visibility, the most relevant frequency for the repetition of the elevation profile

of Kourou is ≈ 1 day. Notice that in this case the repetition doesn't ensure a full range of visibility. Infact, after 1.5 days of propagation the profile seems to become more complex. Hence, this site does not guarantee reliable observation windows.

- On the other hand, the elevation patterns of Troll and Svalbard are very similar in terms of shape but shifted in time, for example they both touch higher values of elevation angles with higher frequency. This basically implies similar periodic repetition of the visibility window of SMOS. The short term trade-off of the orbit determination carried out in subsection 2.4 showed a worst performance in terms of precision for Troll. This is due to the limited span of values that were observed during that particular visibility window of interest (also appreciated from the low maximum elevation reached at the first times of the propagation in Figure 11). It is expected that on the long term, Troll can provide observation batches of measurements similar to Svalbard due to their comparable EL profile. The more frequent visibility from these sites with respect to Kourou is justified by their latitudinal positioning (near the poles), hence allowing more repeatable observations of SMOS.

The long term analysis revealed the exclusion of the Kourou observation site due to its position near the equator, which is not suitable with the observation of the polar orbit of SMOS. This aspect has been confirmed by the elevation profile plotted for that site for three days. The angle showed some peaks with a frequency of one day interspersed with two low peaks (these last both visible at 0.05 days and 1.05 days of propagation Figure 11). Conversely, both Svalbard and Troll can provide very similar observation profile (in terms of elevation) due to their location on the Earth that is favorable to observation of polar orbits. The peaks of elevation angles are contained in large interval of ≈ 0.5 days for both sites, and the frequency of the peak is slightly less 0.1 days (which is approximately a peak every 2.5 hours for a half of a day). This reveals that the selection of the two ground stations will discard Kourou and consider Troll and Svalbard.

2.6 Appendix - Batch filters

2.6.1 Jacobian calculation: inclination component

The gradient of the inclination i with respect to the orbital state $\mathbf{x}$ can be calculated using the chain rule, the various components are retrieved as follows:

$$i = \arccos \frac{h_z}{h} \qquad \frac{\partial i}{\partial h} = \frac{h_z^2}{h^2\sqrt{1-(h_z/h)^2}} \qquad \frac{\partial i}{\partial h_z} = \frac{1}{h\sqrt{1-(h_z/h)^2}} \tag{12}$$

$$\frac{\partial h}{\partial \mathbf{x}} = \frac{\partial h}{\partial \mathbf{h}}\frac{\partial \mathbf{h}}{\partial \mathbf{x}} = \frac{\mathbf{h}}{h}\begin{bmatrix} -[\mathbf{v}\times] & [\mathbf{r}\times] \end{bmatrix} = \begin{bmatrix} -[\mathbf{v}\times]^t \\ [\mathbf{r}\times]^t \end{bmatrix}\frac{\mathbf{h}}{h} = \begin{bmatrix} [\mathbf{v}\times] \\ [-\mathbf{r}\times] \end{bmatrix}\frac{\mathbf{h}}{h} \tag{13}$$

Where $[\mathbf{v}\times]$ and $[\mathbf{r}\times]$ are the skew-symmetric matrices that formalize the cross product as a matrix product. The skew symmetric matrix A is such that $A^t = -A$. The transposition is necessary in order to have a column vector as the final result. On the other hand, the derivative of the h_z component is carried out by explicitly calculating $h_z = xv_y - yv_x$ and derive it component by component:

$$\frac{\partial h_z}{\partial \mathbf{x}} = \begin{bmatrix} v_y & -v_x & 0 & -y & x & 0 \end{bmatrix} \tag{14}$$

2.6.2 Batch filter estimated state and covariances

The navigation solutions for the three cases of the state estimation processes with the minimum variance non-linear least squares filter are here reported. The state $\mathbf{x}_0$ is the full orbital state of SMOS expressed in the Earth centered J2000 frame (equatorial) at the epoch $t_0 =$ `2024-11-18T20:30:00.000`. The associated covariance is the full, symmetric 6×6 matrix. Even though the full matrix it is less significative with respect to σ_r or σ_v due to the 36 numbers inside, it contains all the necessary information to reconstruct different metrics of evaluation of the uncertainty.

Kourou GS with Keplerian motion

Parameter	Value
Estimated state $\mathbf{x}_0$	$\mathbf{r}_0 =$ [3933.265, -1417.104, 5780.823] [km]
	$\mathbf{v}_0 =$ [4.88024, -3.76091, -4.23828] [km/s]
Covariance $\mathbf{P}_{car,r}$	$\begin{bmatrix} 65.1534 & 32.8665 & -15.9815 \\ & 62.0996 & -35.0837 \\ \text{Sym.} & & 20.1368 \end{bmatrix}$ [km^2]
Covariance $\mathbf{P}_{car,rv}$	$\begin{bmatrix} -0.0621 & -0.0410 & -0.0228 \\ -0.0174 & -0.0490 & 0.0168 \\ 0.0070 & 0.0265 & -0.0114 \end{bmatrix}$ [(km^2/s)^2]
Covariance $\mathbf{P}_{car,v}$	$\begin{bmatrix} 6.345\cdot10^{-5} & 3.029\cdot10^{-5} & 3.034\cdot10^{-5} \\ & 4.422\cdot10^{-5} & -2.906\cdot10^{-6} \\ \text{Sym.} & & 2.571\cdot10^{-5} \end{bmatrix}$ [(km/s)^2]

All ground stations with Keplerian motion

Parameter	Value
Estimated state $\mathbf{x}_0$	$\mathbf{r}_0 =$ [3926.928, -1411.064,5780.009] [km]
	$\mathbf{v}_0 =$ [4.88852, -3.76497, -4.23340] [km/s]
Covariance $\mathbf{P}_{car,r}$	$\begin{bmatrix} 0.6490 & -0.3921 & -0.1086 \\ & 0.3782 & 0.0980 \\ \text{Sym.} & & 0.0827 \end{bmatrix}$ [km^2]
Covariance $\mathbf{P}_{car,rv}$	$\begin{bmatrix} -6.514\cdot10^{-4} & 2.211\cdot10^{-4} & -4.063\cdot10^{-4} \\ 4.459\cdot10^{-4} & -1.124\cdot10^{-4} & -2.925\cdot10^{-4} \\ 7.532\cdot10^{-5} & -3.034\cdot10^{-5} & 1.072\cdot10^{-4} \end{bmatrix}$ [(km^2/s)^2]
Covariance $\mathbf{P}_{car,v}$	$\begin{bmatrix} 7.408\cdot10^{-7} & -1.862\cdot10^{-7} & 4.124\cdot10^{-7} \\ & 1.222\cdot10^{-7} & -1.353\cdot10^{-7} \\ \text{Sym.} & & 3.011\cdot10^{-7} \end{bmatrix}$ [(km/s)^2]

Table 11: Estimated $\mathbf{x}_0$ at t_0 in Earth-Centered J2000 frame - All GSs & pure Keplerian motion

All ground stations with J2 perturbation

Parameter	Value
Estimated state $\mathbf{x}_0$	$\mathbf{r}_0$ = [3932.753, -1414.906,5778.500] [km]
	$\mathbf{v}_0$ = [4.87983, -3.76322, -4.23288] [km/s]
Covariance $\mathbf{P}_{car,r}$	$\begin{bmatrix} 4.467 \cdot 10^{-4} & -2.706 \cdot 10^{-4} & -7.534 \cdot 10^{-5} \\ & 2.621 \cdot 10^{-4} & 6.805 \cdot 10^{-5} \\ \text{Sym.} & & 5.739 \cdot 10^{-5} \end{bmatrix}$ [km^2]
Covariance $\mathbf{P}_{car,rv}$	$\begin{bmatrix} -4.477 \cdot 10^{-7} & 1.521 \cdot 10^{-7} & -2.795 \cdot 10^{-7} \\ 3.077 \cdot 10^{-7} & -7.739 \cdot 10^{-8} & 2.019 \cdot 10^{-7} \\ 5.223 \cdot 10^{-8} & -2.099 \cdot 10^{-8} & 7.429 \cdot 10^{-8} \end{bmatrix}$ [(km^2/s)^2]
Covariance $\mathbf{P}_{car,v}$	$\begin{bmatrix} 5.089 \cdot 10^{-10} & -1.276 \cdot 10^{-10} & 2.834 \cdot 10^{-10} \\ & 8.413 \cdot 10^{-11} & -9.293 \cdot 10^{-11} \\ \text{Sym.} & & 2.073 \cdot 10^{-10} \end{bmatrix}$ [(km/s)^2]

Table 12: Estimated $\mathbf{x}_0$ at t_0 in Earth-Centered J2000 frame - All GSs & pure Keplerian motion

Exercise 3: Sequential filters

An increasing number of lunar exploration missions will take place in the next years, many of them aiming at reaching the Moon's surface with landers. In order to ensure efficient navigation performance for these future missions, space agencies have plans to deploy lunar constellations capable of providing positioning measurements for satellites orbiting around the Moon.

Considering a lander on the surface of the Moon, you have been asked to improve the accuracy of the estimate of its latitude and longitude (considering a fixed zero altitude). To perform such operation you can rely on the use of a lunar orbiter, which uses its Inter-Satellite Link (ISL) to acquire range measurements with the lander while orbiting around the Moon. At the same time, assuming the availability of a Lunar Navigation Service, you are also receiving measurements of the lunar orbiter inertial position vector components, such that you can also estimate the spacecraft state within the same state estimation process.

To perform the requested tasks you can refer to the following points.

1. *Check the visibility window.* Considering the initial state $\mathbf{x}_0$ and the time interval with a time-step of 30 seconds from t_0 to t_f reported in Table 13, predict the trajectory of the satellite in an inertial Moon-centered reference frame assuming Keplerian motion. Use the estimated coordinates given in Table 14 to predict the state of the lunar lander. Finally, check that the lander and the orbiter are in relative visibility for the entire time interval.

2. *Simulate measurements.* Always assuming Keplerian motion to model the lunar orbiter dynamics around the Moon, compute the time evolution of its position vector in an inertial Moon-centered reference frame and the time evolution of the relative range between the satellite and the lunar lander. Finally, simulate the measurements by adding a random error to the spacecraft position vector and to the relative range. Assume a Gaussian model to generate the random error, with noise provided in Table 13 for both the relative range and the components of the position vector. Verify (graphically) that the applied noise level is within the desired boundary.

3. *Estimate the lunar orbiter absolute state.* As a first step, you are asked to develop a sequential filter to narrow down the uncertainty on the knowledge of the lunar orbiter absolute state vector. To this aim, you can exploit the measurements of the components of its position vector computed at the previous point. Using an Unscented Kalman Filter (UKF), provide an estimate of the spacecraft state (in terms of mean and covariance) by sequentially processing the acquired measurements in chronological order. To initialize the filter in terms of initial covariance, you can refer to the first six elements of the initial covariance $\mathbf{P}_0$ reported in Table 13. For the initial state, you can perturb the actual initial state $\mathbf{x}_0$ by exploiting the MATLAB function `mvnrnd` and the previously mentioned initial covariance. We suggest to use $\alpha = 0.01$ and $\beta = 2$ for tuning the UT in this case. Plot the time evolution of the error estimate together with the 3σ of the estimated covariance for both position and velocity.

4. *Estimate the lunar lander coordinates.* To fulfill the goal of your mission, you are asked to develop a sequential filter to narrow down the uncertainty on the knowledge of the lunar lander coordinates (considering a fixed zero altitude). To this aim, you can exploit the measurements of the components of the lunar orbiter position vector together with the measurements of the relative range between the orbiter and the lander computed at the previous point. Using an UKF, provide an estimate of the spacecraft state and the lunar lander coordinates (in terms of mean and covariance) by sequentially processing the acquired measurements in chronological order. To initialize the filter in terms of initial covariance, you can refer to the initial covariance $\mathbf{P}_0$ reported in Table 13. For the initial state, you can perturb the actual initial state, composed by $\mathbf{x}_0$ and the latitude

and longitude given in Table 14, by exploiting the MATLAB function `mvnrnd` and the previously mentioned initial covariance. We suggest to use $\alpha = 0.01$ and $\beta = 2$ for tuning the UT in this case. Plot the time evolution of the error estimate together with the 3σ of the estimated covariance for both position and velocity.

Table 13: Initial conditions for the lunar orbiter.

Parameter	Value
Initial state $\mathbf{x}_0$ [km, km/s]	$\mathbf{r}_0$ = [4307.844185282820, -1317.980749248651, 2109.210101634011] $\mathbf{v}_0$ = [-0.110997301537882, -0.509392750828585, 0.815198807994189]
Initial time t_0 [UTC]	2024-11-18T16:30:00.000
Final time t_f [UTC]	2024-11-18T20:30:00.000
Measurements noise	σ_p = 100 m
Covariance $\mathbf{P}_0$ [km^2, km^2/s^2, rad^2]	`diag([10,1,1,0.001,0.001,0.001,0.00001,0.00001])`

Table 14: Lunar lander - initial guess coordinates and horizon mask

Lander name	MOONLANDER
Coordinates	LAT = 78° LON = 15° ALT = 0 m
Minimum elevation	0 deg

3.1 Check the relative visibility

The state $\mathbf{x}$ of the lunar orbiter during the time interval of interest is computed through a Keplerian propagation from $\mathbf{x}_0$, using a 30 seconds time span from t_0 to t_f. The equation of motion for the orbiter in the Moon centered J2000 frame is:

$$\begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ -\mu \dfrac{\mathbf{r}}{r^3} \end{bmatrix} \tag{15}$$

The propagation is carried out in *Matlab* with *ode113*, using `RelTol` set to `1e-13` and `AbsTol` set to `1e-20` for both position ([km]) and velocity ([km/s]). As requested, also the state of the lunar lander $\mathbf{x}_L$ can be retrieved, which can be expressed in the Moon centered J2000 inertial frame by means of a coordinate transformation. In particular:

1. The initial guess of the Lunar Lander coordinates expressed in Table 14 are converted to rectangular coordinates, using *reclat* or *pggrec* routines from *spice*. The position vector that will be found is the position of the lander in rectangular coordinates expressed in the Moon centered - Moon fixed (MCMF) reference frame: `IAU-MOON`.
2. Define the velocity of the lander in the MCMF frame as a 3×1 vector of zeros.
3. Transform the state at each time epoch of interest from the MCMF frame to the Moon centered J2000 inertial frame, using the transformation obtained from *sxform* from *spice*.

Finally, the visibility of the orbiter from the lander can be checked by computing the elevation profile of the spacecraft seen from the guess position of the lander site. Several ways can be adopted in order to retrieve the AZEL & range profiles, but all methods converge then into transforming from rectangular to spherical the coordinates of the vector $\boldsymbol{\rho}$ from the observation site to the orbiter expressed in the site-topocentric frame. This vector could be retrieved exploiting the just calculated orbiter trajectory in MCI frame and the lander state in MCI frame, and defining a transformation from MCI to Lander-centric coordinates.

In this case, another strategy has been used in order to get familiarity with the generation of ground station *kernels* using the *pinpoint* routine of *spice*. The kernel definition (called `lander_guess`) uses the guess coordinates in Table 14, the topocentric frame is defined with the x-axis towards north and the y-axis towards west. After that, a *Matlab* function called *antenna data* is created in order to calculate the ideal observed quantities (AZEL and range), seen from a specific position on the Moon (defined by the kernel). The routine *antenna data* works as described by the following procedure:

Antenna data routine:
Input: $\mathbf{r}_{ORB,ECI}$, t_{span}, Site name.
Loop over the elements of t_{span}:
1. Get $\mathbf{r}_{land,MCI}$ at t_k in MCI -J2000 frame (*spkpos* from spice)
2. Get $\mathbf{r}_{ORB,MCI} - \mathbf{r}_{land,MCI} = \boldsymbol{\rho}_{MCI}$
3. Get transformation matrix $\boldsymbol{R}$ from J2000 to TOPOCENTRIC frame centered at the lander site (*pxform* from spice)
4. Get $\boldsymbol{\rho}_{topo} = \boldsymbol{R}\,\boldsymbol{\rho}_{MCI}$
5. Get Az, El, ρ from the topocentric coordinate $\boldsymbol{\rho_{topo}}$ (*reclat* from spice).

The results of the ideal observations of the lunar orbiter calculated from the lander guess site using the *antenna data* function along the specified time span are represented graphically in Figure 12.

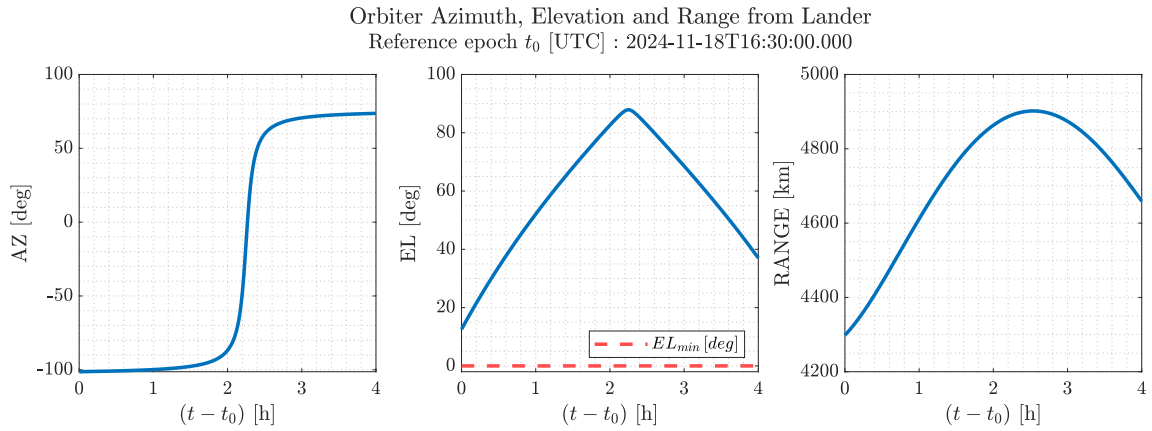

Figure 12: Orbiter AZEL range profiles during the epoch: t_0 to t_f specified in Table 13

As seen in the second plot of Figure 12, the visibility during the interested time period is satisfied. Moreover, the range value increases and with a maximum around 2.5 hours after the initial epoch t_0. This is due to the elliptical shape of the orbit and its apogee location visible from the lander site. This fact can be graphically appreciated from the trajectory of the orbiter in Figure 13, plotted in the Moon centered J2000 frame. The lunar lander position has been plotted only at t_0 since the movement due to Moon rotation is negligible in that time window (only for representation purposes). From the elevation plot it is also noticeable the fact that the visibility window is a little bit larger than the specified t_0 to t_f time span (considering that the observation site on the Moon surface is not moving too much in four hours time span).

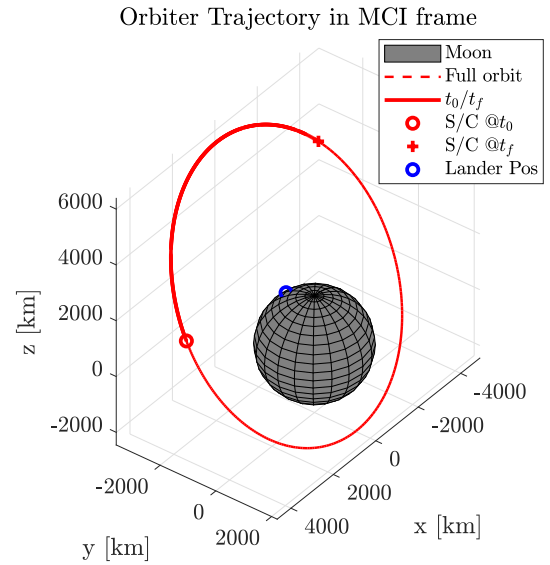

Figure 13: Trajectory in MCI frame - side 1

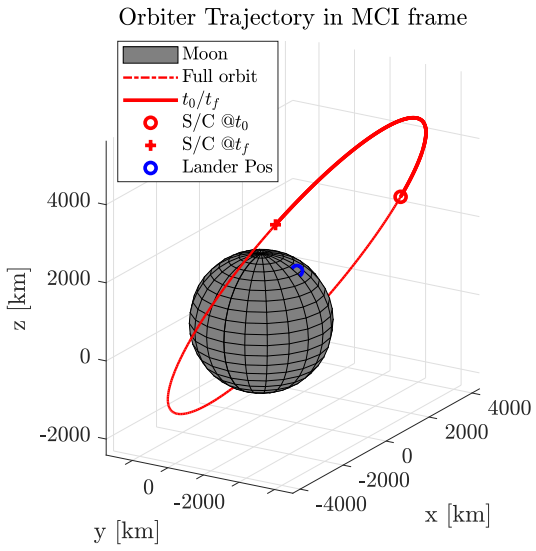

Figure 14: Trajectory in MCI frame - side 2

Since the inertial frame MCI is Moon centered J2000, this frame has **not** the +z axis aligned with the Moon rotation axis (because J2000 is based on the Earth equator). Since the Moon has not a high flatness, it is simple represented as a sphere recalling that the top points of those spheres in Figure 13 and Figure 14 are not exactly the poles.

3.2 Simulate measurement of inertial orbiter position and relative range

The simulation of the measurements have been performed for:

- The **inertial position of the orbiter**, assuming that this quantity can be measured by the Lunar Navigation Service. This measurement has been simulated by taking as mean value the already propagated Keplerian inertial position over the period t_0 to t_f. Then, Gaussian noise has been added using the *mvnrnd* function of *Matlab* by using a diagonal covariance matrix with equal elements ($\sigma^2 = 10^{-2}$ km^2).
- The **relative range** from the orbiter to the S/C acquired by the ISL present on the S/C. This measurement can be simulated by considering the following procedure:

  1. Use the **ground-truth** kernel defined as `moon_lander`, to find the *true* position of the lander in the MCI frame $\mathbf{r}_{land,MCI}$
  2. Calculate the relative position between the orbiter and the lander in MCI frame as:

     $$\boldsymbol{\rho}_{MCI} = \mathbf{r}_{ORB,MCI} - \mathbf{r}_{land,MCI}$$

     Which is the difference between the mean of the measurement of the S/C position and the true position of the lander in MCI.
  3. The mean value of the range measurements is defined as:

     $$\mu_\rho = |\boldsymbol{\rho}_{MCI}|$$

     The actual measurements are defined by adding Gaussian noise to μ_ρ using the *mvnrnd* function and setting the covariance to $\sigma^2 = 10^{-2}$ km^2 (in this case is a scalar covariance).

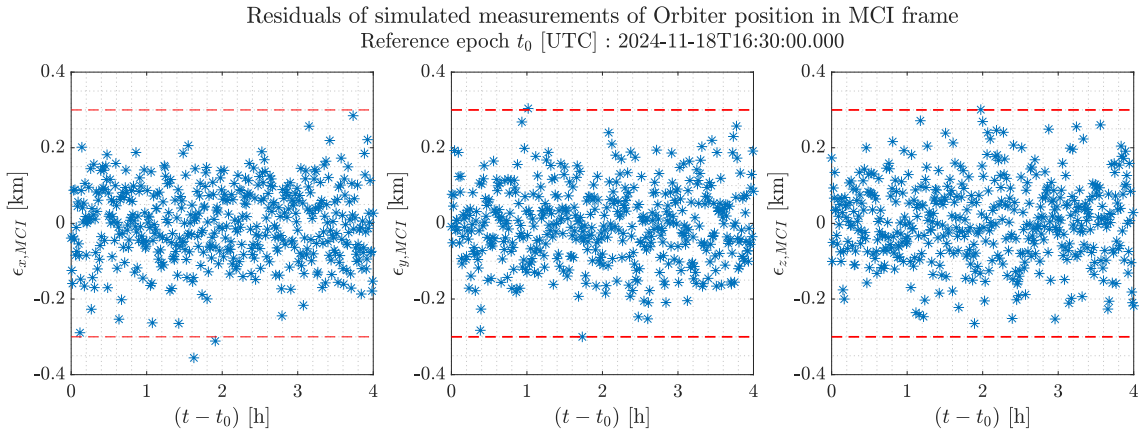

Figure 15: Residuals for the simulated orbiter position in MCI frame

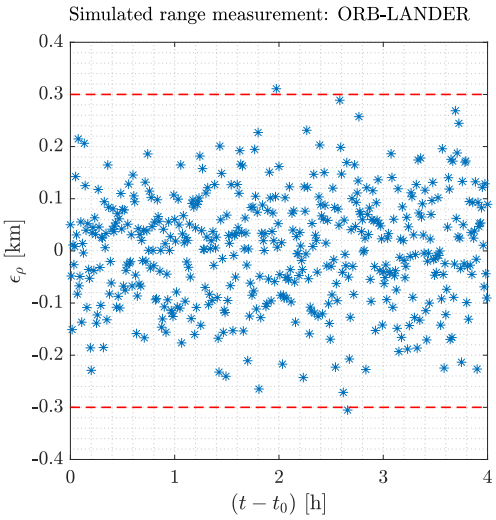

Figure 16: Residuals for the simulated orbiter range

Each plot represents the difference between the simulated quantity and its corrispondent mean value as a blue star marker, while the 3σ bounds are represented by red dotted lines. Due to the Gaussian nature of the simulated measurements, it is expected that 99.7% of the samples fall within the 3sigma boundary. All five plots contains exactly 481 simulated quantities, 0.3% of them (approximately 1.4) can happen to be outside the dotted region.

3.3 Estimate lunar orbiter state

This section addresses the implementation of an UKF with the aim of estimating the orbiter state in the Moon-Centered J2000 frame during the time span from t_0 to t_f, in which measurements are sequentially supplied. The following hypothesis are made on the estimation process:

- **Dynamical model:** the dynamical model used inside the filter is the keplerian dynamics around the Moon, no perturbations are added. Process noise is not considered. The dynamics non-linear function is:

$$\boldsymbol{f}(\mathbf{x}) = \begin{bmatrix} \mathbf{v} \\ -\mu \dfrac{\mathbf{r}}{r^3} \end{bmatrix}$$

- **Measurement model:** the measurement equation used inside the filter is a transformation which selects the position components of the MCI state.

$$\mathbf{x}_{MCI} \xrightarrow{\boldsymbol{h}(\mathbf{x})} \mathbf{r}_{MCI}$$

  Additive noise in considered in the measurement model, with covariance defined by the diagonal matrix $\mathbf{R}$ with σ_p^2 elements as specified in Table 13

- The Unscented Transform used in the UKF is tuned with $\alpha = 0.01$ and $\beta = 2$, as specified by the text.

- The simulated measurements used are the one produced as specified in subsection 3.2.

The procedure of the UKF for the state estimation is here presented:

- **Initialize the filter:** the initialization has been performed by taking the state of the orbiter $\mathbf{x}_0$ and perturbed with Gaussian noise with covariance the 6x6 upper-left submatrix

of $\mathbf{P}_0$ from Table 13, using *mvnrnd* of *Matlab*. The UT weights have been calculated as:

$$W_0^{(m)} = \frac{\lambda}{n+\lambda} \qquad W_i^{(m)} = \frac{1}{2(n+\lambda)} \qquad i = 1 \cdots 2n \tag{16}$$

$$W_0^{(c)} = \frac{\lambda}{n+\lambda} + (1 - \alpha^2 + \beta) \qquad W_i^{(c)} = \frac{1}{2(n+\lambda)} \qquad i = 1 \cdots 2n \tag{17}$$

Given that n represents the number of elements in the state vector to be estimated and $\lambda = (\alpha^2 - 1)n$

- **Loop** over the measurements: from $k = 1 \cdots N_m$
- **Prediction step:**
  1. **Define** χ points for each epoch t_{k-1}

$$\boldsymbol{\chi}_{0,k-1} = \hat{\mathbf{x}}_{k-1}^{+} \quad \boldsymbol{\chi}_{i,k-1} = \hat{\mathbf{x}}_{k-1}^{+} \pm \left(\sqrt{(n+\lambda)\hat{\boldsymbol{P}}_{\boldsymbol{k-1}}^{+}} \right)_{i,col} \qquad i = 1, \cdots 2n$$

  2. **Apply** the propagation function $\boldsymbol{f}$ to get the propagated sigma points $\chi_{i,k}$ at the epoch of interest t_k

$$\boldsymbol{\chi_{i,k}} = \boldsymbol{f}\left(\boldsymbol{\chi}_{i,k-1}\right) \qquad i = 0, \cdots 2n$$

  The propagation is performed in *Matlab* with `ode113`, with `RelTol` set to `1e-13` and `AbsTol` set to `1e-20` (for position [km] and velocity [km/s]).

  3. **Project** the sigma points onto the measurements subspace to get the basis of measurements t_k.

$$\gamma_{i,k} = \boldsymbol{h}(\chi_{\mathbf{i,k}}) \quad i = 0, \cdots 2n$$

  4. **Calculate** predicted quantities at t_k through a linear combination:

$$\hat{\mathbf{x}}_k^{-} = \sum_i W_i^{(m)} \boldsymbol{\chi_{i,k}} \qquad \hat{\mathbf{y}}_k^{-} = \sum_i W_i^{(m)} \boldsymbol{\gamma_{i,k}}$$

$$\boldsymbol{P}_k^{-} = \sum_i W_i^{(c)} (\boldsymbol{\chi}_{i,k} - \hat{\mathbf{x}}_k^{-})(\boldsymbol{\chi}_{i,k} - \hat{\mathbf{x}}_k^{-})^T$$

$$\boldsymbol{P}_{ee,k} = \sum_i W_i^{(c)} (\boldsymbol{\gamma}_{i,k} - \hat{\mathbf{y}}_k^{-})(\boldsymbol{\gamma}_{i,k} - \hat{\mathbf{y}}_k^{-})^T + \boldsymbol{R}$$

$$\boldsymbol{P}_{xy,k} = \sum_i W_i^{(c)} (\boldsymbol{\chi}_{i,k} - \hat{\mathbf{x}}_k^{-})(\boldsymbol{\gamma}_{i,k} - \hat{\mathbf{y}}_k^{-})^T$$

  The innovation covariance matrix $\boldsymbol{P}_{ee,k}$ contains the additive covariance matrix noise due to additivity of noise in the measurement model.

- **Update step** at epoch t_k:
  1. **Calculate** Kalman gain as

$$\boldsymbol{K}_k = \boldsymbol{P}_{xy,k} \boldsymbol{P}_{ee,k}^{-1} \tag{18}$$

  2. **Update** the estimate of the state and its associated covariance as:

$$\hat{\mathbf{x}}_k^{+} = \hat{\mathbf{x}}_k^{-} + \boldsymbol{K}_k (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^{-}) \tag{19}$$

$$\boldsymbol{P}_{\boldsymbol{k}}^{+} = \boldsymbol{P}_{\boldsymbol{k}}^{-} - \boldsymbol{K}_k \boldsymbol{P}_{ee,k} \boldsymbol{K}_k^T \tag{20}$$

  Where $\boldsymbol{y}_k$ is the measurement at time t_k.

- **Repeat** from **prediction** step by putting $k = k + 1$

3.3.1 UKF state estimation: numerical results

The results obtained from the implementation of the UKF to estimate the orbital state of the spacecraft in the Moon centered J2000 frame, from t_0 to t_f are reported in Figure 17. In particular, the estimated error is computed as the norm of the difference for both $\mathbf{r}$ and $\mathbf{v}$ by taking as truth the Keplerian propagated state. In this context this *true* value is meaningful since the measurements are simulated over the purely keplerian motion. The 3σ bounds for the position and velocity are obtained as:

$$3\sigma_r = 3\sqrt{\max[eig(\boldsymbol{P}_r)]} \qquad 3\sigma_v = 3\sqrt{\max[eig(\boldsymbol{P}_v)]}$$

calculated at each time instant t_k of measurements.

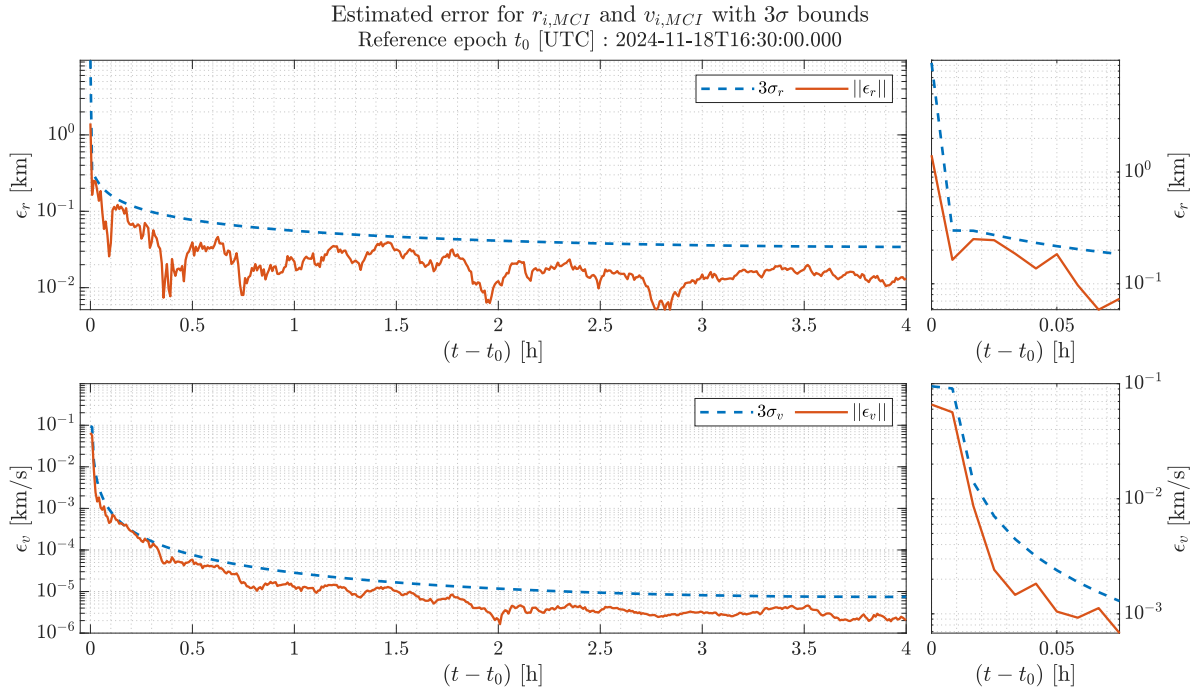

Figure 17: UKF results: Estimated error for $\mathbf{r}$ and $\mathbf{v}$ with 3σ bounds

The convergence of the filter is different on the two quantities, this fact can be appreciated by the plots in Figure 17 (the right graphs are zoomed regions of the corresponding left in a time frame of some minutes). The position convergence is much steeper and reaches a plateau of around $\approx 3 \cdot 10^{-2} km$ at the end of the estimation process. On the other side, the convergence on the velocity starts at the acquisition of the *second measurement* and it is less steep than the position. The minimal decrease of the uncertainty on the velocity within the first measurement is due to the fact that the filter has not acquired enough information from the measurements (which are relative only to the position). In 0.05h $\approx$ 3 minutes (which are 6 measurements) the uncertainty decreases approximately of one order of magnitude, the plateau is reached farther in time. At the end of the acquisition campaign, the uncertainty on the velocity is around 10^{-5} km/s.

3.4 Estimate lunar lander coordinates and orbiter state

This section aims at presenting the implementation of a UKF which estimates the position of the lander and the orbital state of the spacecraft. The measurements are the inertial position of the orbiter and its relative range between the lander site, simulated in subsection 3.2. The

extension of the estimation of parameters (latitude and longitude of the lander site) in the UKF needs a slight modification to that presented in subsection 3.3, since the filter has to estimate some parameters within the same step. Estimation of the latitude and longitude of the lander is possible since the range measurement is affected by the location of the lander site.

- **Augmented state:** the estimation of some parameters within the filtering process requires to define an augmented state:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x} \\ \mathbf{p} \end{bmatrix} \qquad \text{where:} \quad \mathbf{x} = \begin{bmatrix} \mathbf{r}_{MCI} \\ \mathbf{v}_{MCI} \end{bmatrix} \qquad \mathbf{p} = \begin{bmatrix} \phi \\ \theta \end{bmatrix}$$

  The state $\mathbf{x}$ is the orbital state of the spacecraft in MCI frame, while the vector $\mathbf{p}$ contains the latitude and longitude respectively of the lander site.

- **Dynamical model:** within the framework of the UKF, a dynamical model must be defined for the augmented state $\mathbf{X}$. Since the parameters of latitude and longitude are constant, the augmented state dynamics become:

$$\boldsymbol{f}(\mathbf{x}) = \begin{bmatrix} \mathbf{v} \\ -\mu \dfrac{\mathbf{r}}{r^3} \\ \mathbf{0}_{2\times 1} \end{bmatrix}$$

- **Measurement model:** the measurement equation used inside the filter is also slightly modified from the previous implementation. Infact, a measurement which depends on the parameters must be added. The relative range is a measure which depends both on the orbiter state and the lander site position.

$$\boldsymbol{h}(\mathbf{X}) = \begin{bmatrix} \mathbf{r}_{MCI} \\ \rho \end{bmatrix}$$

  Similarly to the state-estimation UKF, the first component of the measurement equation is straightforward once the orbital state is given. On the other hand, the computation of the relative range ρ is less trivial. A brief explanation of how it can be retrieved is here given:

  Antenna data routine:
  Input: $\mathbf{p} = (\phi, \theta)$, t_k, $\boldsymbol{r}_{MCI}$
  1. From the parameter vector $\mathbf{p}$ which contains the lander position, retrieve its position in the Moon centered Moon fixed (MCMF) reference frame in rectangular coordinates. This step requires the *pggrec* spice routine, it is assumed that the altitude of the site is exactly 0 meters.
  2. The position of the orbiter $\mathbf{r}_{MCI}$ can be expressed in the MCMF reference. This step requires the use of the *pxform* spice routine to define the transformation between Moon centered J2000 frame to the MCMF frame.
  3. Calculate the range ρ as the norm of the position vector from lander site to orbiter, expressed in MCMF frame.

  The measurement model assumes additive noise defined by the 4×4 diagonal covariance matrix $\boldsymbol{R}$ with σ_p^2 elements as specified in Table 13.

- The Unscented Transform used in the UKF is tuned with $\alpha = 0.01$ and $\beta = 2$, as specified by the text.

- The simulated measurements used, are produced as specified in subsection 3.2 relative to the orbiter inertial position and relative range over the time window from t_0 to t_f

The procedure of the UKF for the state and parameters estimation is here presented:

- **Initialize the filter:** the initialization of the filter augmented state $\mathbf{X}_0$ has been defined using as mean values the ones reported in Table 13 (latitude and longitude converted to radians) and adding Gaussian noise with covariance defined by the 8×8 matrix $\mathbf{P}_0$ from Table 13. The *Matlab* function *mvnrnd* has been used for the purpose. The UT weights have been calculated as:

$$W_0^{(m)} = \frac{\lambda}{n+\lambda} \qquad W_i^{(m)} = \frac{1}{2(n+\lambda)} \qquad i = 1 \cdots 2n \tag{21}$$

$$W_0^{(c)} = \frac{\lambda}{n+\lambda} + (1 - \alpha^2 + \beta) \qquad W_i^{(c)} = \frac{1}{2(n+\lambda)} \qquad i = 1 \cdots 2n \tag{22}$$

  Given that n=8 represent the number of elements in the augmented state and $\lambda = (\alpha^2 - 1)n$

- **Loop** over the measurements: from $k = 1 \cdots N_m$

- **Prediction step:**

  1. **Define** χ points for the epoch of t_{k-1} with $k = 1, \cdots n_{meas}$

$$\boldsymbol{\chi}_{0,k-1} = \hat{\mathbf{X}}_{k-1}^{+} \quad \boldsymbol{\chi}_{i,k-1} = \hat{\mathbf{X}}_{k-1}^{+} \pm \left(\sqrt{(n+\lambda)\hat{\boldsymbol{P}}_{\boldsymbol{k-1}}^{+}} \right)_{i,col} \qquad i = 1, \cdots 2n$$

     As it happens for the augmented state $\mathbf{X}$, also the sigma points $\boldsymbol{\chi}_{i,k}$ can be split into two components relative to orbiter state and parameters:

$$\boldsymbol{\chi}_{i,k} = \begin{bmatrix} \boldsymbol{\chi}_{i,k}^{s} \\ \boldsymbol{\chi}_{i,k}^{p} \end{bmatrix}$$

  2. **Apply** the dynamics function $\boldsymbol{f}$ to get the propagated sigma points $\chi_{i,k}$ at the epoch of interest t_k

$$\boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{s}} = \boldsymbol{f}_{\boldsymbol{kep}} \left(\boldsymbol{\chi}_{i,k-1}^{s} \right) \qquad i = 1 \cdots 2n$$
$$\boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{p}} = \boldsymbol{\chi}_{\boldsymbol{i,k-1}}^{\boldsymbol{p}} \qquad i = 1 \cdots 2n$$

     Notice that the last two elements of $\boldsymbol{f}$ represent a static dynamics for the parameters. Hence, the *Matlab* implementation requires only the propagation of the first 6 elements of each $\boldsymbol{\chi}_{i,k-1}$. The propagation is performed in *Matlab* with `ode113`, with `RelTol` set to `1e-13` and `AbsTol` set to `1e-20`.

  3. **Project** the sigma points onto the measurements subspace to get the basis of measurements t_k.

$$\gamma_{i,k} = \boldsymbol{h}(\boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{p}}, \boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{s}}) \quad i = 0, \cdots 2n$$

     Which includes both ideal measurements of position of the orbiter in MCI and relative range. To this last aim, apply the procedure implemented before to simulate the relative range measurement (the antenna data routine presented in the box of section. 3.4)

  4. **Calculate** predicted quantities at t_k through a linear combination:

$$\hat{\mathbf{X}}_k^{-} = \begin{bmatrix} \sum_i W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{s}} \\ \sum_i W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} \sum_i W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{s}} \\ \hat{\mathbf{p}}_{k-1}^{+} \end{bmatrix}$$

Where the equality $\sum_i W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{p}} = \hat{\mathbf{p}}_{k-1}^{+}$ is due to the absence of dynamics in the context of the parameters propagation of uncertainty. The equality is also explained in the appendix 3.5.1. The predicted mean of the measurements is instead:

$$\hat{\mathbf{y}}_k^- = \sum_i W_i^{(m)} \boldsymbol{\gamma}_{\boldsymbol{i,k}}$$

$$\boldsymbol{P}_k^- = \sum_i W_i^{(c)} (\boldsymbol{\chi}_{i,k} - \hat{\mathbf{X}}_k^-)(\boldsymbol{\chi}_{i,k} - \hat{\mathbf{X}}_k^-)^T$$

$$\boldsymbol{P}_{ee,k} = \sum_i W_i^{(c)} (\boldsymbol{\gamma}_{i,k} - \hat{\mathbf{y}}_k^-)(\boldsymbol{\gamma}_{i,k} - \hat{\mathbf{y}}_k^-)^T + \boldsymbol{R}$$

$$\boldsymbol{P}_{Xy,k} = \sum_i W_i^{(c)} (\boldsymbol{\chi}_{i,k} - \hat{\mathbf{X}}_k^-)(\boldsymbol{\gamma}_{i,k} - \hat{\mathbf{y}}_k^-)^T$$

The innovation covariance matrix $\boldsymbol{P}_{ee,k}$ contains the additive covariance matrix noise due to additivity of noise in the measurement model.

- **Update step** at epoch t_k:
  1. **Calculate** Kalman gain as

$$\boldsymbol{K}_k = \boldsymbol{P}_{Xy,k} \boldsymbol{P}_{ee,k}^{-1} \tag{23}$$

  2. **Update** the estimate of the state and its associated covariance as:

$$\hat{\mathbf{X}}_k^+ = \hat{\mathbf{X}}_k^- + \boldsymbol{K}_k(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^-) \tag{24}$$

$$\boldsymbol{P}_{\boldsymbol{k}}^+ = \boldsymbol{P}_{\boldsymbol{k}}^- - \boldsymbol{K}_k \boldsymbol{P}_{ee,k} \boldsymbol{K}_k^T \tag{25}$$

- **Repeat** from **prediction** step by putting $k = k + 1$

3.4.1 UKF parameters and state estimation: numerical results

This section presents the numerical results of a single run for the state and parameter estimation UKF. Given that the true parameters of the lander latitude and longitude are present in the kernel `moon_lander`, the convergence process of the estimated parameters is shown in Figure 18.

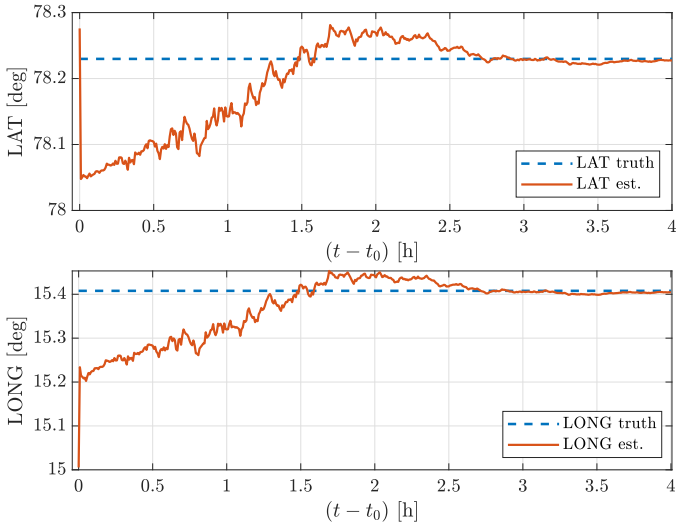

Figure 18: Convergence of parameters - UKF

A more accurate representation of the state and parameters estimation process of the UKF is given in Figure 19. The graphs show respectively:

- The error on the state estimation for both position and velocity in MCI frame along the t_0 to t_f interval is calculated as the norm of difference between the estimated state by the UKF and the purely keplerian propagated state in MCI taken as the truth (which was used to construct the measurements). Moreover, the 3σ value for both position and velocity is calculated as:

$$3\sigma_r = 3\sqrt{max(eig(\boldsymbol{P}_r))}$$
$$3\sigma_v = 3\sqrt{max(eig(\boldsymbol{P}_v))}$$

  On the right side there is a zoom to better appreciate the first phases of action of the UKF.

- The last horizontal graph represent the evolution of the error on the parameter estimation process calculated as the absolute value of the difference between the estimated parameter and the true parameter taken from the `moon_lander` kernel. The 3σ is calculated for both parameters as:

$$3\sigma_p = 3\sqrt{max(eig(\boldsymbol{P}_\mathbf{p}))} \tag{26}$$

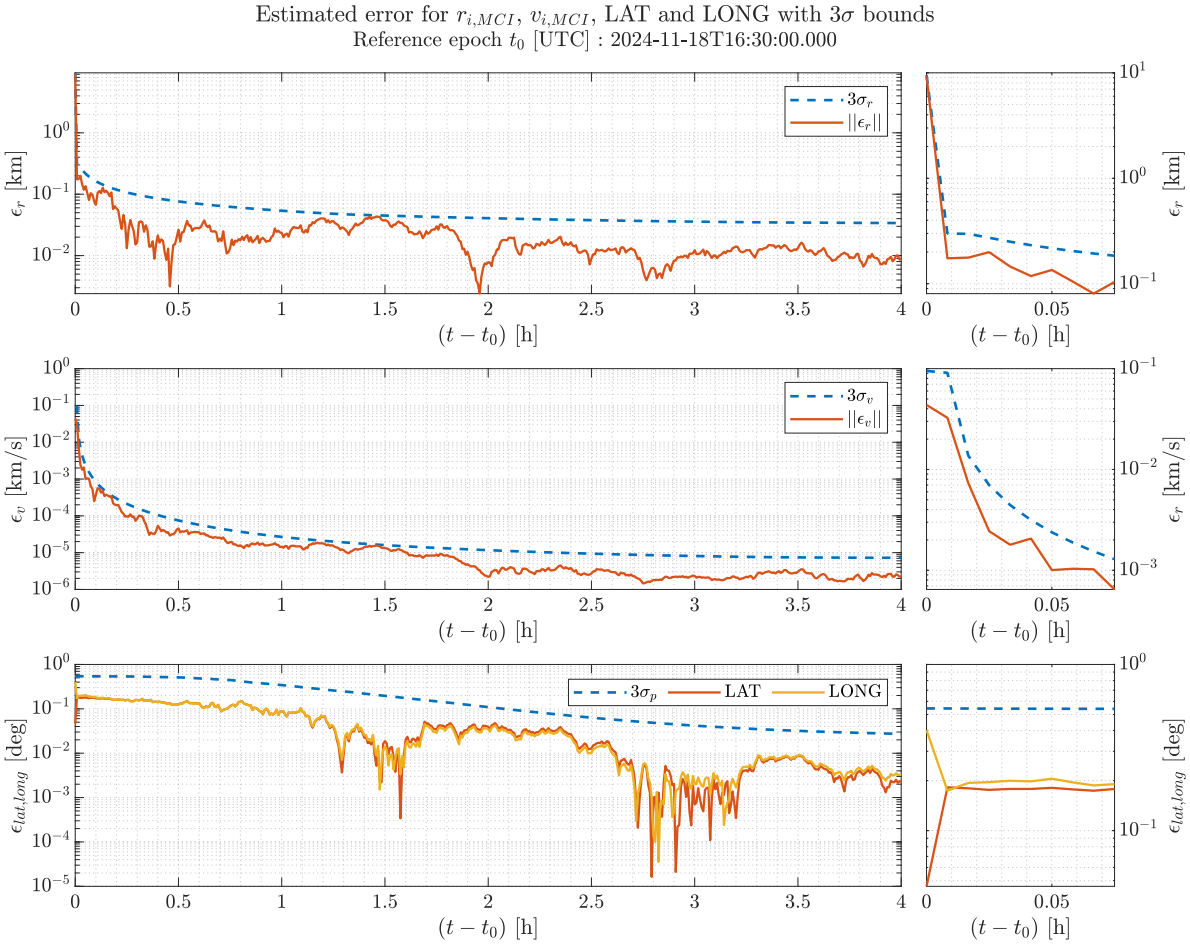

Figure 19: State and parameters estimation - UKF

Some comments on the performance of the UKF for the state and parameters estimation are here reported:

- The convergence process of the uncertainty of state of the orbiter is not much altered by the introduction of the parameter dynamics. Similar results are found with respect to the purely-state estimation UKF with only inertial position measurements. In particular, convergence values of the 3σ values are around $3 \cdot 10^{-2}$ km and 10^{-5} km/s.
- On the other side, the convergence of the uncertainty on the parameters is slower, eventually at the end of the measurement campaign the uncertainty has been narrowed down to $3 \cdot 10^{-2}$ deg for both longitude and latitude.
- From the right plot of the last row of graphs in Figure 19 it can be appreciated a zoom on the first iteration of the UKF filter on the parameters estimated error. In particular, it is noticeable how the estimated error on one of the two parameters grows while the other one decreases in the same step. After that, similar behavior of both errors on the parameters can be observed on the long term. This indicates a statistical correlation between the latitude and longitude parameters. This correlation can be verified by expressing the covariance submatrix relative to the parameters during the UKF process, in the correlation matrix form using the *corrcov* function from *Matlab*. The high correlation present in the $\boldsymbol{P}_{\mathbf{p}}$ covariance matrix (extradiagonal term is > 0.9 for the whole estimation process) is the reason why it has been decided to use the eigenvalue estimator for the 3σ as shown in Equation 26 as opposite to simply the square root of the diagonal elements $\boldsymbol{P}_{\mathbf{p}}$ for each parameter. In this way the cross-correlation uncertainty (which in this case is high) is also accounted for and the 3σ estimator is more conservative.

In conclusion, the UKF is capable of reducing the uncertainties on both orbiter state in MCI and latitude/longitude angular positioning of the lander during the time span t_0 to t_f.

3.5 Appendix

3.5.1 UKF with parameters estimation details

$$\hat{\mathbf{X}}_k^- = \begin{bmatrix} \sum_i W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{s}} \\ \sum_i W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} \sum_i W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{s}} \\ \hat{\mathbf{p}}_{k-1}^+ \end{bmatrix}$$

Where the equality

$$\sum_{i=0}^{2n} W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{p}} = \hat{\mathbf{p}}_{k-1}^+ \tag{27}$$

Can be demonstrated recalling that

$$\boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{p}} = \boldsymbol{\chi}_{\boldsymbol{i,k-1}}^{\boldsymbol{p}} \qquad i = 0 \cdots 2n \tag{28}$$

From the definition of the sigma points:

$$\boldsymbol{\chi}_{\boldsymbol{0,k-1}}^{\boldsymbol{p}} = \hat{\mathbf{p}}_{k-1}^+$$

$$\boldsymbol{\chi}_{\boldsymbol{i,k-1}}^{\boldsymbol{p}} = \hat{\mathbf{p}}_{k-1}^+ \pm \left(\sqrt{(n+\lambda)\hat{\boldsymbol{P}}_{\boldsymbol{k-1}}^+} \right)_i \qquad i = 1 \cdots 2n$$

Where only the last two rows of the square root matrix are taken (the values corresponding to the parameter vector $\mathbf{p}$). Since Equation 28 holds due to lack of parameters dynamics, then Equation 27 can be rewritten using $\boldsymbol{\chi}_{\boldsymbol{i,k-1}}^{\boldsymbol{p}}$ as:

$$\sum_{i=0}^{2n} W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i,k}}^{\boldsymbol{p}} = \sum_{i=0}^{2n} W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i,k-1}}^{\boldsymbol{p}} = W_0^{(m)} \hat{\mathbf{p}}_{k-1}^+ + W_i^{(m)} \sum_{i=1}^{2n} \left[\hat{\mathbf{p}}_{k-1}^+ \pm \left(\sqrt{(n+\lambda)\hat{\boldsymbol{P}}_{\boldsymbol{k-1}}^+} \right)_i \right] \tag{29}$$

Where $W_i^{(m)}$ is constants for $i = 1 \cdots 2n$ hence it is outside the sum. As can be seen, inside the sum the terms after $\pm$ cancel out since it is sum of n positive terms and n negative terms with same magnitude. What is left out, expliciting the all the weights $W_i^{(m)}$ for $i = 0, \cdots 2n$ is:

$$\sum_{i=0}^{2n} W_i^{(m)} \boldsymbol{\chi}_{\boldsymbol{i},\boldsymbol{k}}^{\boldsymbol{p}} = \frac{\lambda}{n+\lambda} \hat{\mathbf{p}}_{k-1}^{+} + \frac{2n}{2(n+\lambda)} \hat{\mathbf{p}}_{k-1}^{+} = \frac{2\lambda + 2n}{2(n+\lambda)} \hat{\mathbf{p}}_{k-1}^{+} = \hat{\mathbf{p}}_{k-1}^{+}$$

References

[1] Eric A. Wan Vander Merwe. *The uncented Kalman Filter for Nonlinear estimation.* URL: `https://ieeexplore.ieee.org/document/882463`.