SGN – Assignment #2

Samuele Vincenzi, 214867

Disclaimer: The story plot contained in the following three exercises is entirely fictional.

Exercise 1: Uncertainty propagation

The Prototype Research Instruments and Space Mission Technology Advancement (PRISMA) is a technology in-orbit test-bed mission for demonstrating Formation Flying (FF) and rendezvous technologies, as well as flight testing of new sensors and actuator equipment. It was launched on June 15, 2010, and it involves two satellites: Mango (Satellite 1, ID 36599), the chaser, and Tango (Satellite 2, ID 36827), the target.

You have been provided with an estimate of the states of Satellites 1 and 2 at the separation epoch t_{sep} = `2010-08-12T05:27:39.114` (UTC) in terms of mean and covariance, as reported in Table 1. Assume Keplerian motion can be used to model the spacecraft dynamics.

1. Propagate the initial mean and covariance for both satellites within a time grid going from t_{sep} to $t_{sep} + N\,T_1$, with a step equal to T_1, where T_1 is the orbital period of satellite 1 and $N = 10$, using both a Linearized Approach (LinCov) and the Unscented Transform (UT). We suggest to use $\alpha = 0.1$ and $\beta = 2$ for tuning the UT in this case.

2. Considering that the two satellites are in close formation, you have to guarantee a sufficient accuracy about the knowledge of their state over time to monitor potential risky situations. For this reason, at each revolution, you shall compute:

   - the norm of the relative position (Δr), and
   - the sum of the two covariances associated to the position elements of the states of the two satellites (P_{sum})

   The critical conditions which triggers a collision warning is defined by the following relationship:

   $$\Delta r < 3\sqrt{\max(\lambda_i(P_{\text{sum}}))}$$

   where $\lambda_i(P_{\text{sum}})$ are the eigenvalues of P_{sum}. Identify the revolution N_c at which this condition occurs and elaborate on the results and the differences between the two approaches (UT and LinCov).

3. 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 plots of:

   - the time evolution for all three approaches (MC, LinCov, and UT) of $3\sqrt{\max(\lambda_i(P_{r,\text{i}}))}$ and $3\sqrt{\max(\lambda_i(P_{v,\text{i}}))}$, where $i = 1, 2$ is the satellite number and P_r and P_v are the 3x3 position and velocity covariance submatrices.
   - the propagated samples of the MC simulation, together with the mean and covariance obtained with all methods, projected on the orbital plane.

   Compare the results and discuss on the validity of the linear and Guassian assumption for uncertainty propagation.

*Use at least 100 samples drawn from the initial covariance

Table 1: Estimate of Satellite 1 and Satellite 2 states at t_0 provided in ECI J2000.

Parameter	Value
Ref. epoch t_{sep} [UTC]	2010-08-12T05:27:39.114
Mean state $\hat{\boldsymbol{x}}_{0,\text{sat1}}$ [km, km/s]	$\hat{\boldsymbol{r}}_{0,\text{sat1}} = [4622.232026629, 5399.3369588058, -0.0212138165769957]$ $\hat{\boldsymbol{v}}_{0,\text{sat1}} = [0.812221125483763, -0.721512914578826, 7.42665302729053]$
Mean state $\hat{\boldsymbol{x}}_{0,\text{sat2}}$ [km, km/s]	$\hat{\boldsymbol{r}}_{0,\text{sat2}} = [4621.69343340281, 5399.26386352847, -3.09039248714313]$ $\hat{\boldsymbol{v}}_{0,\text{sat2}} = [0.813960847513811, -0.719449862738607, 7.42706066911294]$
Covariance P_0 [km^2, km^2/s, km^2/s^2]	$\begin{bmatrix} +5.6e-7 & +3.5e-7 & -7.1e-8 & 0 & 0 & 0 \\ +3.5e-7 & +9.7e-7 & +7.6e-8 & 0 & 0 & 0 \\ -7.1e-8 & +7.6e-8 & +8.1e-8 & 0 & 0 & 0 \\ 0 & 0 & 0 & +2.8e-11 & 0 & 0 \\ 0 & 0 & 0 & 0 & +2.7e-11 & 0 \\ 0 & 0 & 0 & 0 & 0 & +9.6e-12 \end{bmatrix}$

1.1 Point 1

Exploiting the conservation of energy and given the position and velocity of Mango at the separation time, its semi-major axis and consequently its orbital period are computed. A time grid of 11 points is created, starting from t_{sep}, until $10 \cdot T_1$. The reference state $\mathbf{x}^*(t)$ is described by the following dynamics:

$$\begin{cases} \dot{\mathbf{x}}^*(t) = \mathbf{f}(\mathbf{x}^*, t) \\ \mathbf{x}^*(t_0) = \hat{\mathbf{x}}_{0,sat} \end{cases} \tag{1}$$

Where $\mathbf{f}(\mathbf{x}^*, t)$ is described by a keplerian dynamical model (2BP). For the propagation of the mean state and covariance, two algorithms are considered:

- LinCov approach: Under the hypothesis of linear approximation, the mean state $\hat{\mathbf{x}}(t)$ is equal to the reference state $\mathbf{x}^*(t)$. The updated covariance $P(t)$ is obtained using the State Transition Matrix between t_0 and t, and the covariance at t_0:

$$\begin{cases} \hat{\mathbf{x}}(t) &= \mathbf{x}^*(t) \\ P(t) &= \Phi(t_0, t) P_0 \Phi^T(t_0, t) \end{cases} \tag{2}$$

  Starting from the initial state $\hat{\mathbf{x}}_{0,sat}$, the LinCov uncertainty propagation is applied inside a for-loop, updating initial time, final time, initial state and initial covariance at each iteration. Both the state and the STM are propagated with keplerian dynamics.

- Unscented transform approach: the method starts from the initial state $\hat{\mathbf{x}}_{0,sat}$.

  1. Sigma points computation at t_{sep}:

$$\begin{cases} \chi_0 = \hat{\mathbf{x}}_{0,sat} \\ \chi_i = \hat{\mathbf{x}}_{0,sat} + (\sqrt{cP_0})_i & \text{for i = 1,...,N} \\ \chi_i = \hat{\mathbf{x}}_{0,sat} - (\sqrt{cP_0})_i & \text{for i = N+1,...,2N} \end{cases} \tag{3}$$

  With $c = \alpha^2(N + k)$ and $N = length(\hat{\mathbf{x}}_{0,sat}) = 6$.

2. Weights computation: $W_0^{(m)}, W_0^{(c)}, W_i^{(m)}, W_i^{(c)}$.
3. Sigma points propagation: this procedure is carried out using the MATLAB function *ode*78. The integration is applied by using the total time grid ([1x11] vector) and the initial samples at t_{sep}, χ_i, as inputs to the function, obtaining the sigma points propagation at each intermediate node of the time grid. Overall, 13 propagations (one for each sigma point) are fulfilled: $\chi_i \rightarrow \gamma_{i,j}$, with $i = 0, .., 12$ (for each sigma point) and $j = 1, .., 11$ (for each node of the time grid).
4. Update of mean state and covariance:

$$\begin{cases} \hat{\mathbf{x}}_j = \sum_{i=0}^{2N} W_i^{(m)} \gamma_{i,j} \quad \text{for j = 1,..,11} \\ \\ P_j = \sum_{i=0}^{2N} W_i^{(c)} (\gamma_{i,j} - \hat{\mathbf{x}}_j)(\gamma_{i,j} - \hat{\mathbf{x}}_j)^T \quad \text{for j = 1,..,11} \end{cases} \tag{4}$$

The final mean states (at the last revolution) of the two satellites are reported in Table 2. Moreover, the time evolution of the norm of the mean state is represented in Figure 2. LinCov

LinCov - Mango	LinCov - Tango	UT - Mango	UT - Tango
$\begin{pmatrix} 4622.232027 \\ 5399.336959 \\ -0.021211 \\ 0.81222112 \\ -0.72151291 \\ 7.42665303 \end{pmatrix}$	$\begin{pmatrix} 4622.324616 \\ 5398.702788 \\ 2.683037 \\ 0.80997157 \\ -0.72410974 \\ 7.42706084 \end{pmatrix}$	$\begin{pmatrix} 4622.232020 \\ 5399.336930 \\ -0.021215 \\ 0.81222105 \\ -0.72151299 \\ 7.42665300 \end{pmatrix}$	$\begin{pmatrix} 4622.324582 \\ 5398.702783 \\ 2.682884 \\ 0.80997167 \\ -0.72410962 \\ 7.42706081 \end{pmatrix}$

Table 2: Final value of the mean state [km;km/s]

and UT solutions are very similar: the norm of the position differs of $10^{-4}[km]$ for Mango and $1.6 \cdot 10^{-4}[km]$ for Tango, while the norm of the velocity differs of $10^{-7}[km/s]$ for Mango and $1.6 \cdot 10^{-7}[km/s]$ for Tango.

1.2 Point 2

The uncertainty propagation leads to a worsening of the covariance matrix: the 3σ covariance ellipsoids of position and velocity grow along the propagation. The critical condition that triggers the collision warning occurs when the 3σ position ellipsoids of the two satellites intersect. Starting from the separation time, each revolution is analyzed, applying the following procedure:

$$\begin{cases} P_{sum} = P_{r,sat1} + P_{r,sat2} \\ \Delta r = norm(\hat{\mathbf{r}}_{sat1} - \hat{\mathbf{r}}_{sat2}) \\ residual = \Delta r - 3\sqrt{max(eig(P_{sum}))}; \end{cases} \tag{5}$$

The process is repeated until $residual < 0$. Table 3 highlights the revolution N_C at which the critical condition occurs, pointing out the value of the residual for this situation. Figure 1 shows the time evolution of the two components Δr and $3\sqrt{max(eig(P_{sum}))}$. The intersection happens slightly before the fourth revolution and then again between the ninth and tenth revolutions.

	LinCov	UT
N_C	4	4
$residual$	-0.169777	-0.169531

Table 3: Critical condition analysis

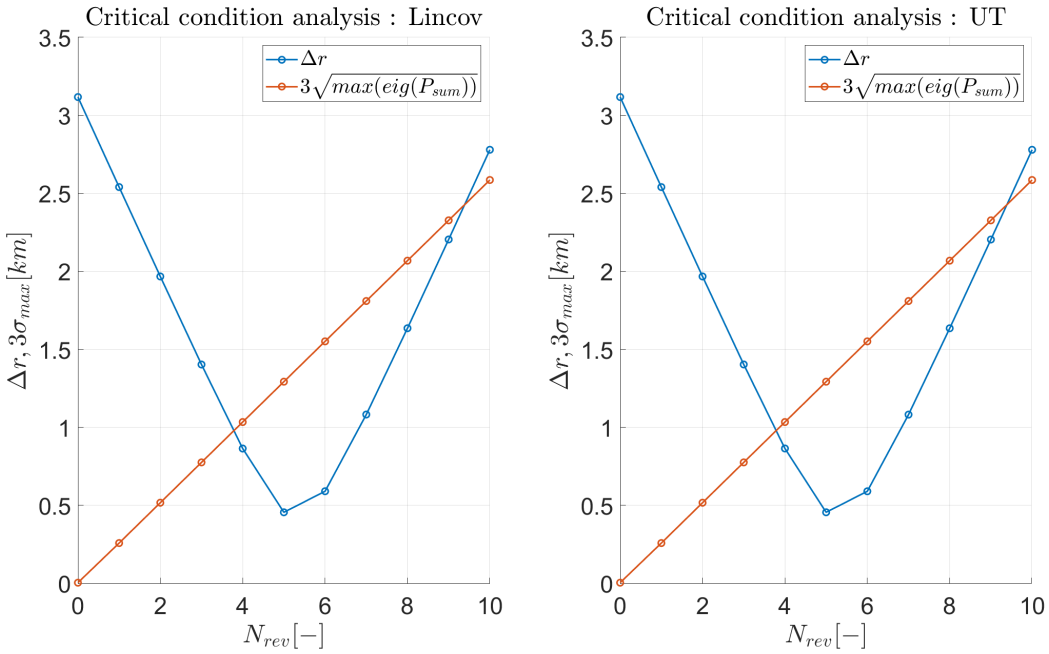

Figure 1: Critical condition analysis

The two images show clearly that the two methods (LinCov and UT) have similar behavior. The LinCov approach is based on the linearization of the flow and dynamical model; it is not suited for highly non-linear systems and long-term propagation. The UT approach is instead based on the sigma points propagation and reconstruction of the mean value and covariance at each step, without applying linearization. Since the two methods provide similar solutions, the linearization employed for the LinCov approach does not affect negatively the results. This is due to the fact that the keplerian dynamics can be well represented by linearized equations if the reference state is not too far from the real value and if the propagation time is not too high.

1.3 Point 3

For the Montecarlo uncertainty propagation, the following algorithm is applied:

1. Samples computation: a Gaussian distribution is supposed and, given the initial mean value $\hat{\mathbf{x}}_{0,sat}$ and covariance P_0, $N = 100$ samples are calculated using the MATLAB function *mvnrnd*.
$$\chi = mvnrnd(\hat{\mathbf{x}}_{0,sat}, P_0) \tag{6}$$
Where χ is a [6xN] matrix containing all the samples.

2. Samples propagation: the same procedure applied for the propagation of the sigma points with the UT approach is employed. Overall, 100 propagations (one for each sample) are fulfilled: $\chi_i \rightarrow \gamma_{i,j}$, with $i = 1, .., N$ and $j = 1, .., 11$.

3. Update of mean state and covariance:
$$\begin{cases} \hat{\mathbf{x}}_j = \dfrac{1}{N}\sum_{i=1}^{N}\gamma_{i,j} \quad \text{for j = 1,..,11} \\ \\ P_j = \dfrac{1}{N-1}\sum_{i=1}^{N}(\gamma_{i,j} - \hat{\mathbf{x}}_j)(\gamma_{i,j} - \hat{\mathbf{x}}_j)^T \quad \text{for j = 1,..,11} \end{cases} \tag{7}$$

Figure 2 shows the time evolution of the norm of the mean value propagated with Lincov, UT and Montecarlo, while in Figure 3 the time evolution of $3\sqrt{max(eig(P))}$ is represented for position and velocity and for both the satellites.

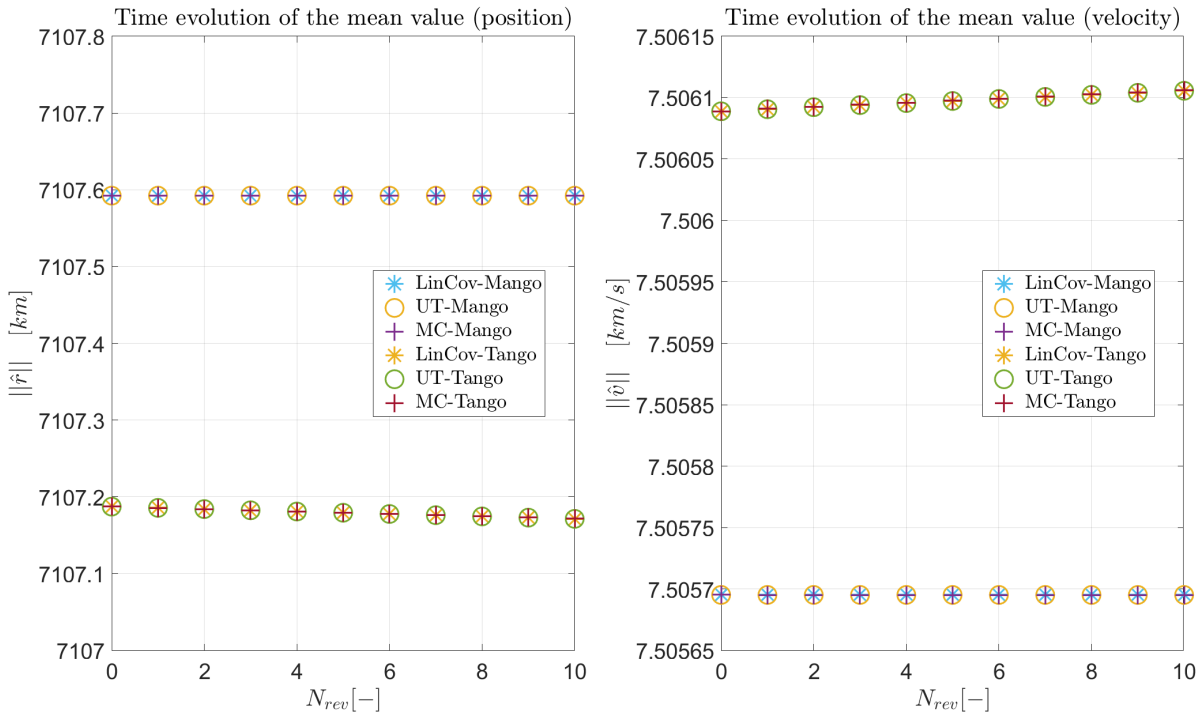

Figure 2: Time evolution of the norm of the mean value

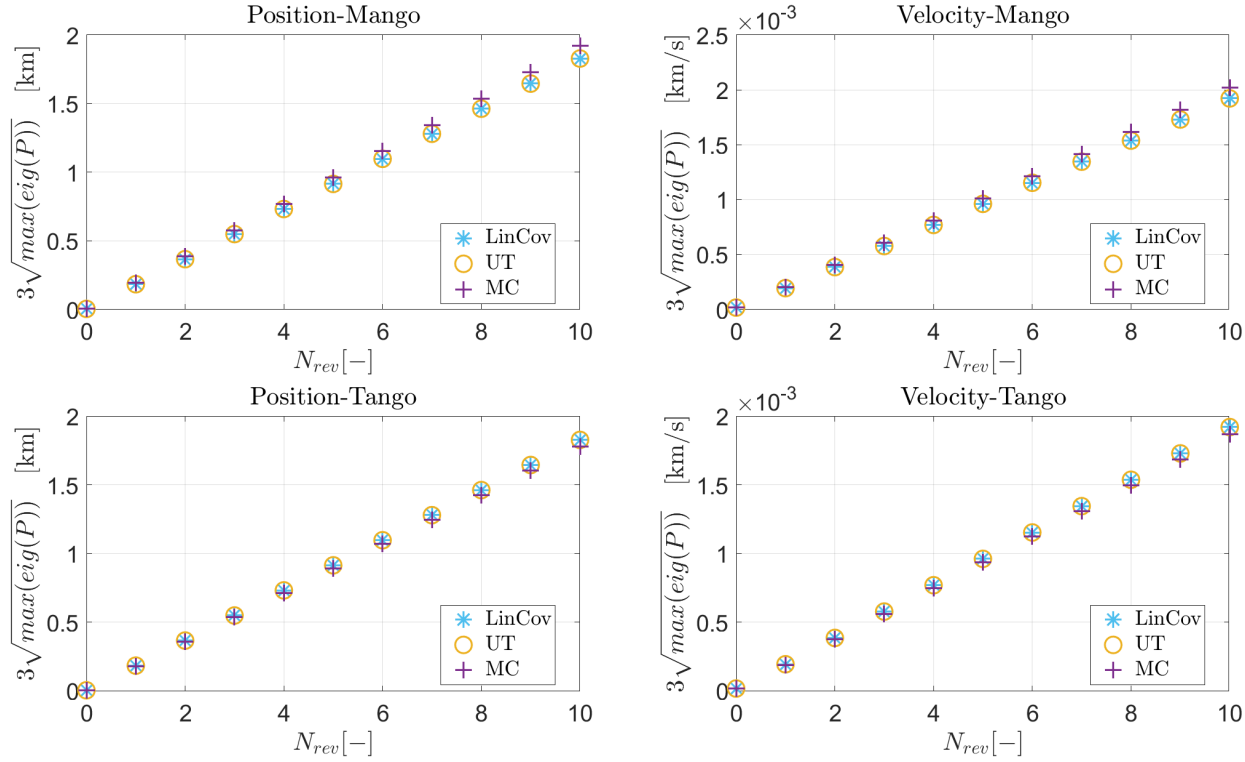

Figure 3: 3σ time evolution

All three approaches show a similar behavior. LinCov and Unscented transform solutions almost coincide, while Montecarlo values slightly differ. The latter is strongly related to the number of random samples chosen for the propagation and computation of the sample mean and covariance: the higher the number of samples the more precise is the Montecarlo solution. As explained in subsection 1.2, the LinCov solution is proven to be well-suited for the case of

study, providing the same result as the one obtained through the unscented transform (with a very small difference). Another proof of this statement comes from the fact that the 3σ grows linearly. In Table 4 the computational effort requested by each approach is reported, showing clearly that the LinCov method is less computationally expensive.

	Lincov	UT	MC
CPU time [s]	0.24515	0.31352	2.17347

Table 4: CPU time

Figure 4, Figure 5 and Figure 6 contain the mean values, the covariance with all the three approaches, plus the Montecarlo samples respectively for $N_{rev} = 0, 5, 10$. The quantities are projected on the perifocal frame of Mango's mean state at t_{sep}: the rotation matrix from the ECI to the perifocal frame is calculated and used to rotate and project the mean values, the 3σ covariance ellipsoids, and the Montecarlo samples. Therefore, the covariance ellipsoids are converted into ellipses. Starting from the shape of Figure 4 the 3σ covariance ellipses of position are stretched during the uncertainty propagation, resulting in an increasing value of the covariance along one direction. As expected from the previous analysis all three methods show a similar behavior for what concerns mean value and covariance. The Montecarlo simulation may slightly differ depending on the number of random samples. Montecarlo's samples are distributed inside the ellipses following their shape and being bounded inside the 3σ value, which represents the region where 99.7% of the Gaussian random samples fall.

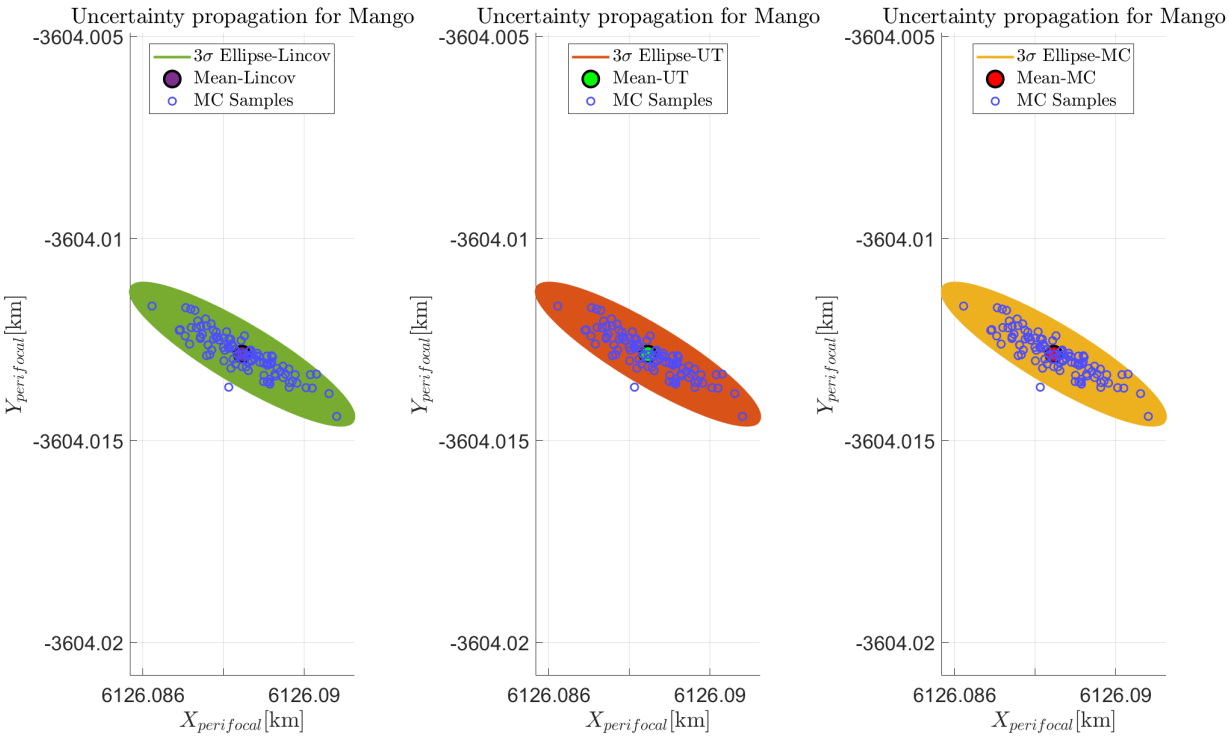

Figure 4: Position mean values, covariance and Montecarlo samples at t_{sep}

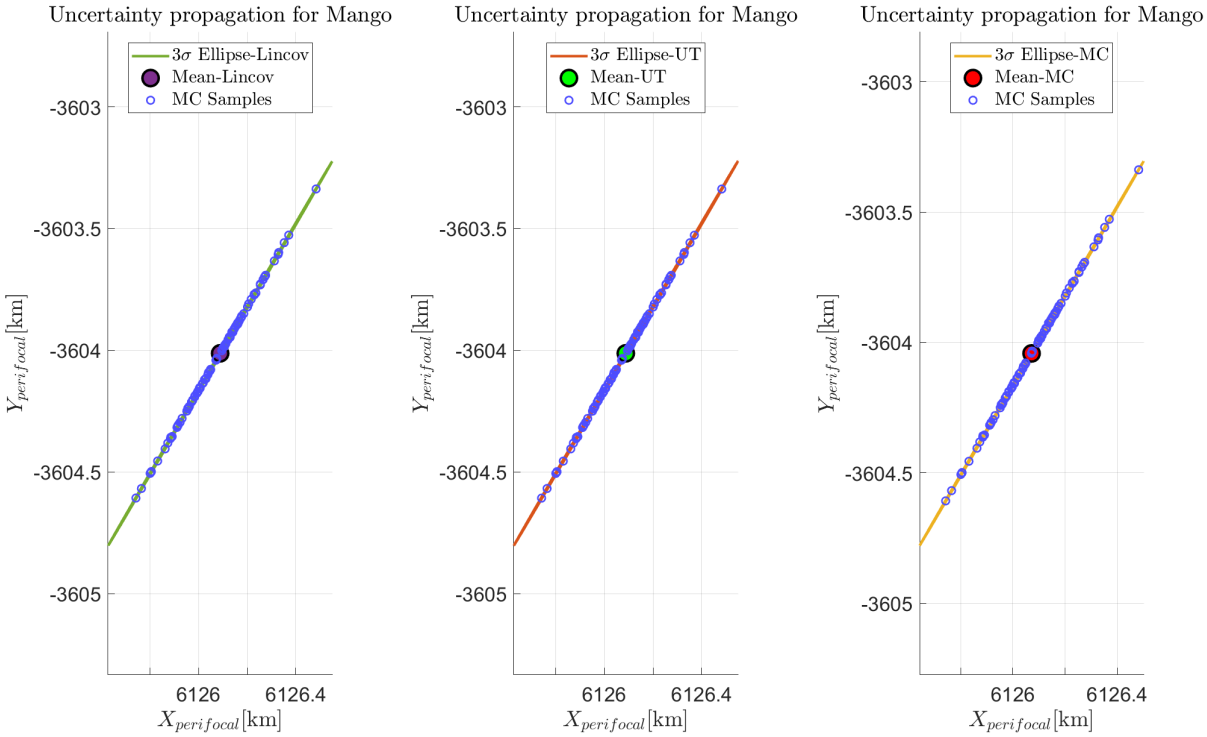

Figure 5: Position mean values, covariance and Montecarlo samples at the fifth revolution

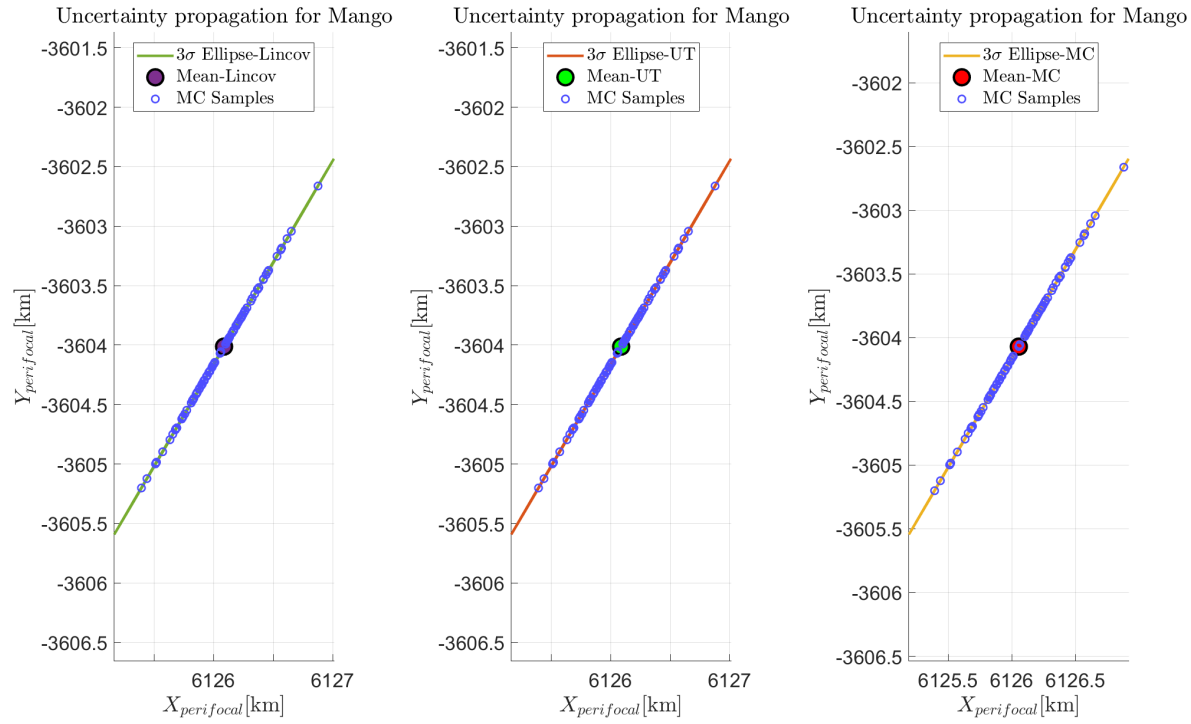

Figure 6: Position mean values, covariance and Montecarlo samples at the last revolution

Figure 7 shows mean value, covariance and Montecarlo samples for the velocity of Mango at the last revolution. In Figure 8 all the quantities for the position of Tango are reported. Since Tango's simulation shows the same trend as Mango's one, only the image corresponding to the position and to the last revolution is inserted.

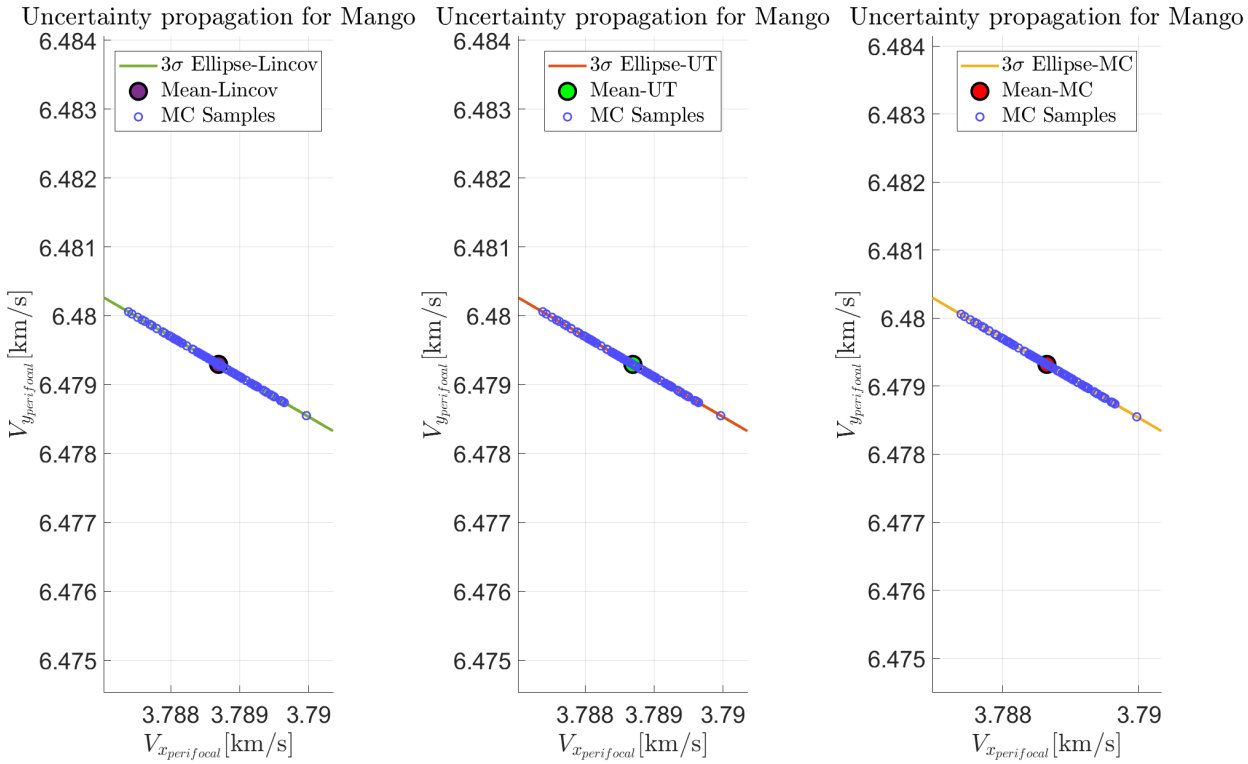

Figure 7: Velocity mean values, covariance and Montecarlo samples at the last revolution

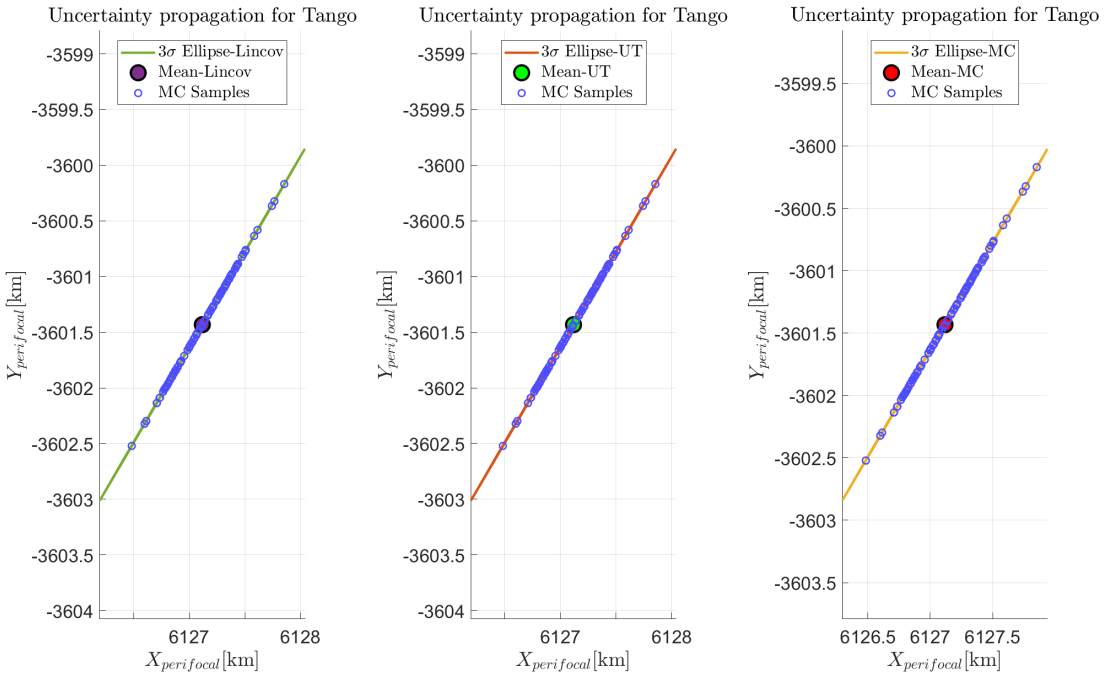

Figure 8: Position mean values, covariance and Montecarlo samples at the last revolution

Conclusions

As explained, the linearization strategy of the LinCov approach behaves well for the uncertainty propagation of the problem considered. Moreover, looking at the distribution of Montecarlo samples, they follow the shape of the ellipse that describes the normal distribution, being confined inside the 3σ boundaries. Therefore, describing the uncertainties with a Gaussian distribution seems reasonable.

Exercise 2: Batch filters

You have been asked to track Mango to improve the accuracy of its state estimate. To this aim, you shall schedule the observations from the two ground stations reported in Table 5.

1. *Compute visibility windows.* By using the mean state reported in Table 1 and by assuming Keplerian motion, predict the trajectory of the satellite over a uniform time grid (with a time step of 60 seconds) and compute all the visibility time windows from the available stations in the time interval from t_0 = `2010-08-12T05:30:00.000` (UTC) to t_f = `2010-08-12T11:00:00.000` (UTC). Plot the resulting predicted Azimuth and Elevation profiles in the visibility windows.

2. *Simulate measurements.* The Two-Line Elements (TLE) set of Mango are reported in Table 6 (and in WeBeep as 36599.3le). Use SGP4 and the provided TLEs to simulate the measurements acquired by the sensor network in Table 5 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 5). 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 6 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 both ground stations.

   (c) Repeat step 3b by using J2-perturbed motion to model the spacecraft dynamics.

4. Provide the obtained navigation solutions and elaborate on the results, comparing the different solutions.

5. Select the best combination of dynamical model and ground stations and perform the orbit determination for the other satellite.

Table 5: Sensor network to track Mango and Tango: list of stations, including their features.

Station name	KOUROU	SVALBARD
Coordinates	LAT = 5.25144° LON = -52.80466° ALT = -14.67 m	LAT = 78.229772° LON = 15.407786° ALT = 458 m
Type	Radar (monostatic)	Radar (monostatic)
Provided measurements	Az, El [deg] Range (one-way) [km]	Az, El [deg] Range (one-way) [km]
Measurements noise (diagonal noise matrix R)	$\sigma_{Az,El}$ = 100 mdeg σ_{range} = 0.01 km	$\sigma_{Az,El}$ = 125 mdeg σ_{range} = 0.01 km
Minimum elevation	10 deg	5 deg

Table 6: TLE of Mango.

```
1 36599U 10028B   10224.22752732 -.00000576  00000-0 -16475-3 0  9998
2 36599 098.2803 049.5758 0043871 021.7908 338.5082 14.40871350  8293
```

Table 7: TLE of Tango.

```
1 36827U 10028F   10224.22753605  .00278492  00000-0  82287-1 0  9996
2 36827 098.2797 049.5751 0044602 022.4408 337.8871 14.40890217    55
```

2.1 Point 1: Visibility windows

Starting from $\hat{\mathbf{x}}_{0,sat1}$, given at the separation time t_{sep}, the mean state is propagated along the time grid using Keplerian dynamics. To compute the azimuth, elevation, and range of Mango, the topocentric position of the ground station and the satellite are requested. Given the station name, the *spice* toolbox allows to calculate its position in an ECI frame and the rotation matrix from an ECI to a topocentric frame. The spacecraft and the station positions are then rotated to a topocentric frame and, at each node of the time grid, the relative position of the satellite with respect to the station is computed. Finally, using the function *cspice_reclat*, azimuth, elevation and range are obtained. Figure 9 represents the elevation time evolution, highlighting the constraint that determines if the satellite is visible:

$$\begin{cases} \text{Elevation} > 10° & \text{for Kourou} \\ \text{Elevation} > 5° & \text{for Svalbard} \end{cases} \tag{8}$$

Two and four visibility windows are respectively noticeable for Kourou and Svalbard.

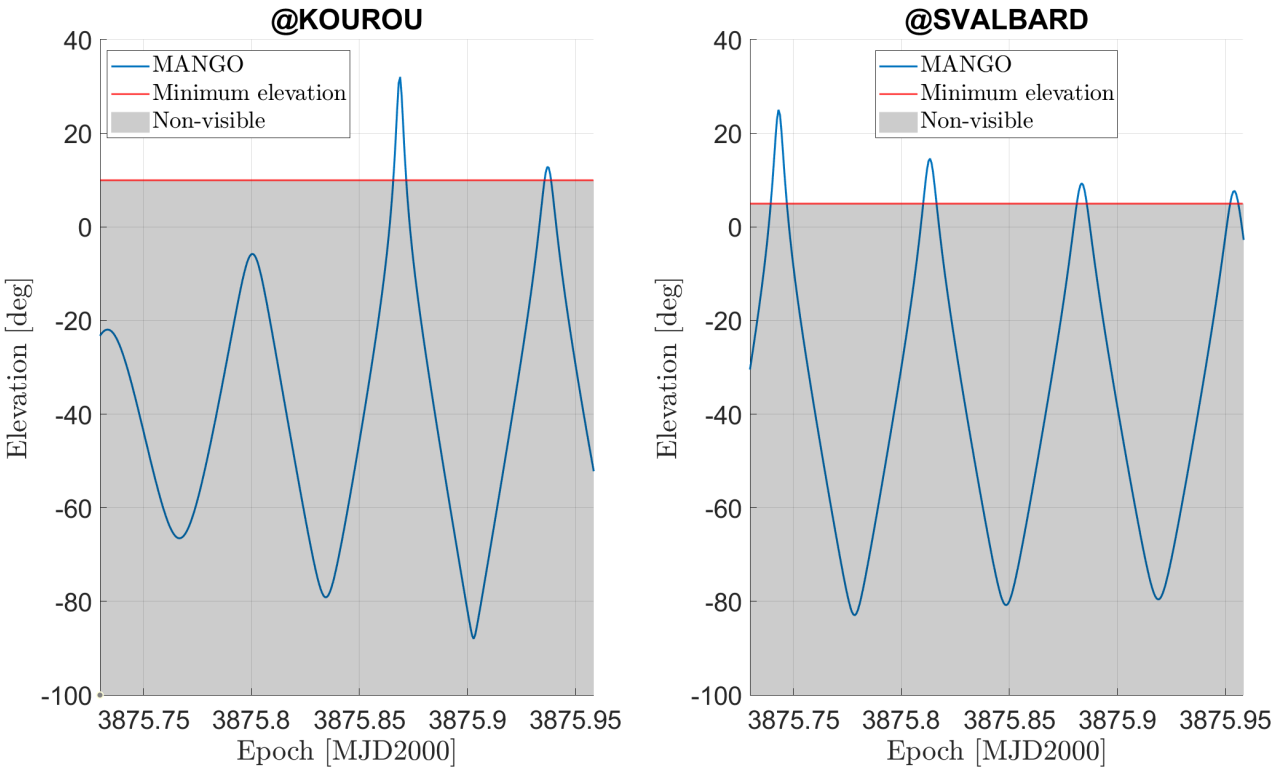

Figure 9: Time evolution of the elevation

By imposing the constraint on the elevation (Equation 8), each element of the time grid is analyzed, selecting only the feasible solutions. Therefore, the visibility window dates are calculated and reported in Table 8 and Table 9.

Visibility window	Initial date	Final date
First	2010 AUG 12 08:46:00.0000 UTC	2010 AUG 12 08:54:00.0000 UTC
Second	2010 AUG 12 10:27:00.0000 UTC	2010 AUG 12 10:30:00.0000 UTC

Table 8: Kourou visibility windows

Visibility window	Initial date	Final date
First	2010 AUG 12 05:44:00.0000 UTC	2010 AUG 12 05:54:00.0000 UTC
Second	2010 AUG 12 07:26:00.0000 UTC	2010 AUG 12 07:34:00.0000 UTC
Third	2010 AUG 12 09:08:00.0000 UTC	2010 AUG 12 09:14:00.0000 UTC
Fourth	2010 AUG 12 10:50:00.0000 UTC	2010 AUG 12 10:55:00.0000 UTC

Table 9: Svalbard visibility windows

Figure 10 represents the azimuth time evolution, highlighting the visibility windows.

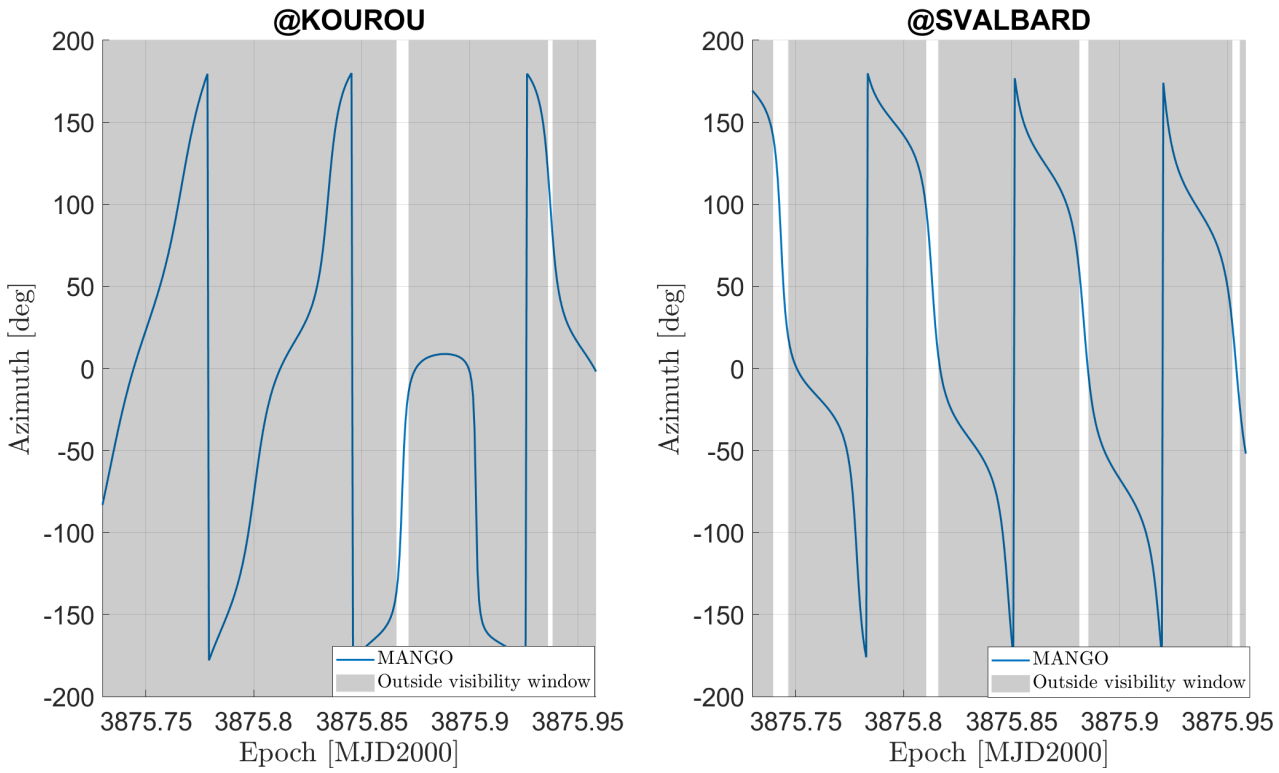

Figure 10: Time evolution of the azimuth

Figure 11 shows the predicted Azimuth-Elevation profile for each ground station and visibility window.

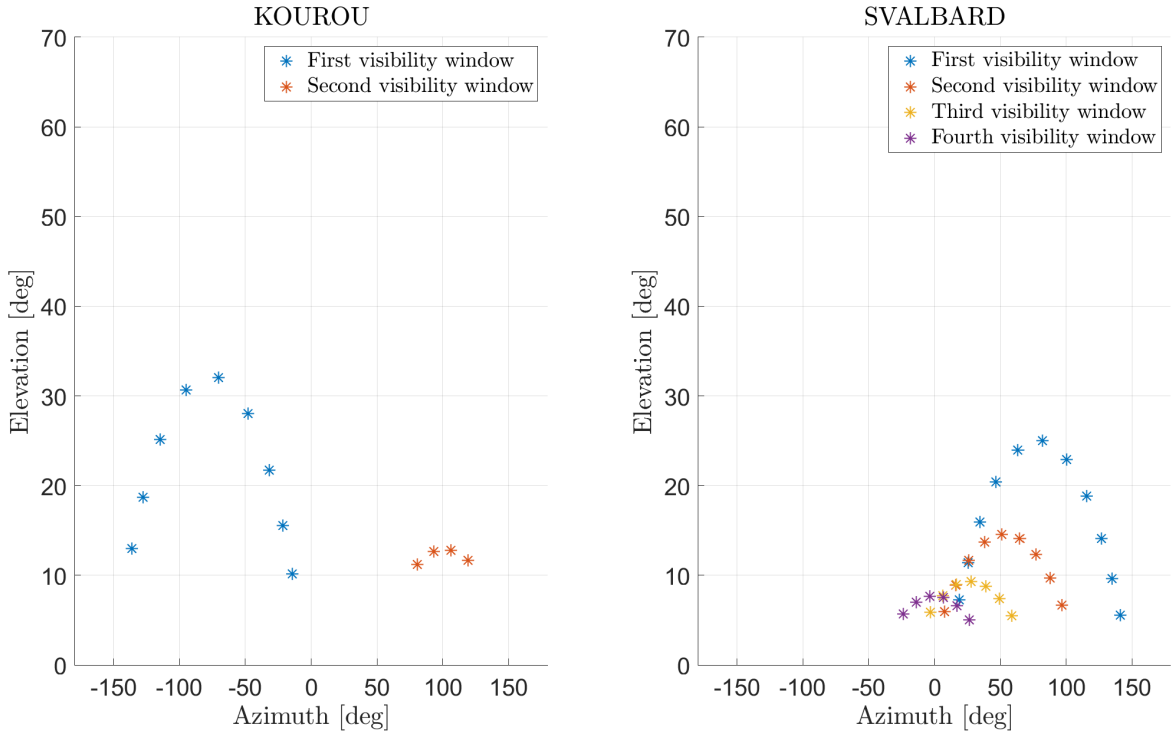

Figure 11: Predicted azimuth and elevation profile

2.2 Point 2: Measurements simulation

Starting from the TLE file and applying the *sgp4* propagation, the spacecraft state in a TEME frame is computed along each visibility window and then rotated to an ECI reference frame. Using the previously explained procedure the azimuth, elevation and range are computed, deriving the expected measurements. Real measurements are then simulated by adding a noise, described by the covariance matrix :

$$R = \begin{bmatrix} \sigma^2_{Az,El} & 0 & 0 \\ 0 & \sigma^2_{Az,El} & 0 \\ 0 & 0 & \sigma^2_{range} \end{bmatrix} \tag{9}$$

The MATLAB function *mvnrnd* is employed, using the expected measurements obtained from the *sgp4* propagation as mean values. By applying the constraint of Equation 8, only the feasible measurements are selected. Figure 12 and Figure 13 show the actual measurements comparing them with the predicted Azimuth-Elevation profile developed in subsection 2.1. The different values are related to the type of propagator employed (Keplerian vs sgp4) and to the noise applied to replicate the instrument uncertainties. For this particular simulation, only one more measurement with respect to the Azimuth-elevation profile of subsection 2.1 is discarded.

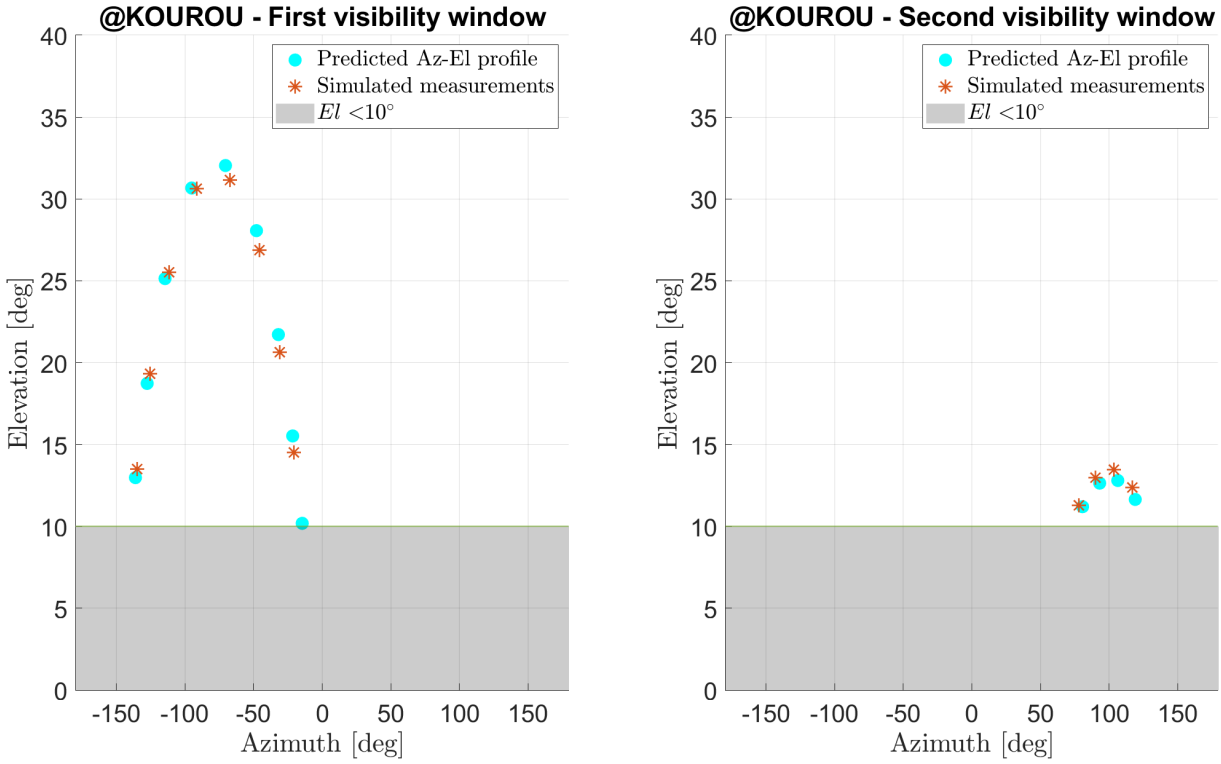

Figure 12: Simulated measurements at Kourou

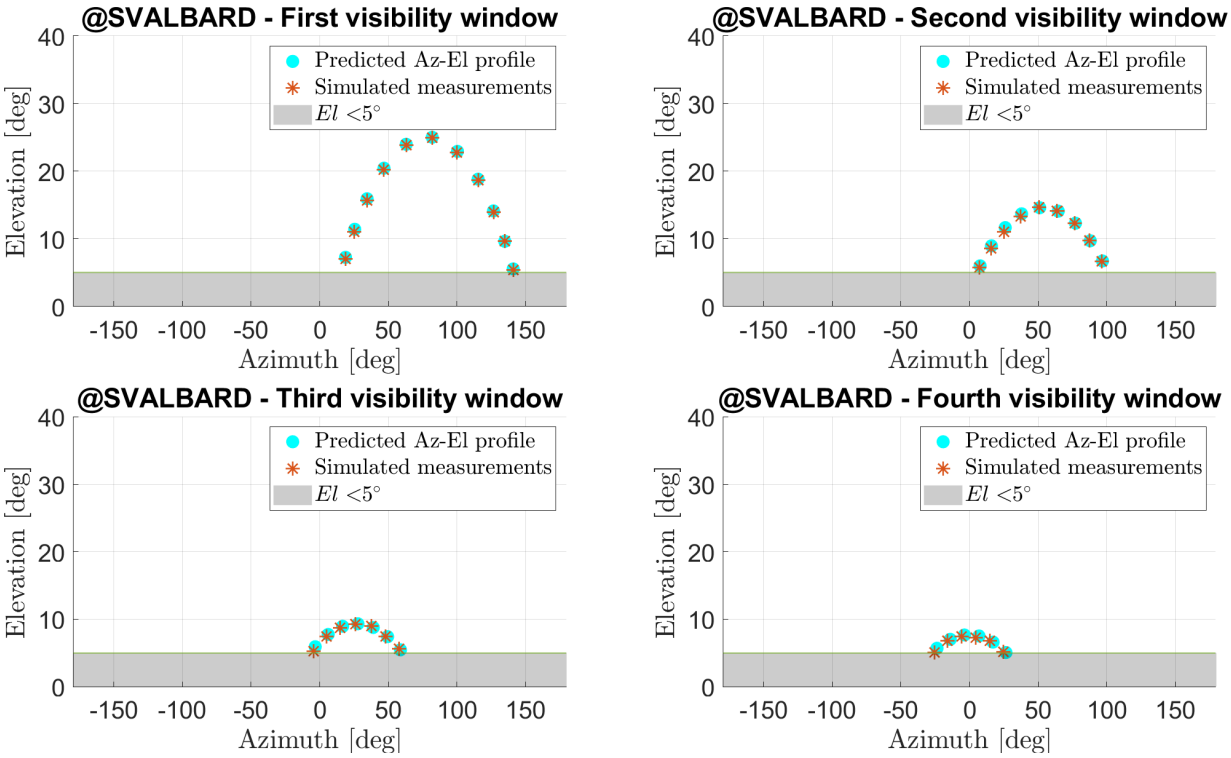

Figure 13: Simulated measurements at Svalbard

2.3 Point 3: Description of the navigation problem

The weighted least squares (minimum covariance) solution is developed by building the cost function that behaves as the input to the MATLAB command *lsqnonlin*. The *sgp*4 propagation of the TLE until the epoch t_0 is the initial guess. The cost function propagates this guess, computes the azimuth, elevation and range associated and finally subtracts the actual measurements from them. The output is a weighted difference, using the following weight matrix:

$$W = \begin{bmatrix} \sigma_{Az,El} & 0 & 0 \\ 0 & \sigma_{Az,El} & 0 \\ 0 & 0 & \sigma_{range} \end{bmatrix}^{-1} \tag{10}$$

Where σ_{range}, $\sigma_{Az,El}$, are the standard deviations corresponding to the range, azimuth, and elevation measurements respectively. Three different strategies for the least squares (minimum covariance) solution are analyzed:

1. Pure Keplerian motion to model the spacecraft dynamics + simulated measurements obtained for the Kourou ground station only.
2. Pure Keplerian motion to model the spacecraft dynamics + simulated measurements obtained for both the Kourou and Svalbard ground stations.
3. Keplerian motion improved with J2 contribution + simulated measurements obtained for both the Kourou and Svalbard ground stations.

2.4 Point 4: Solution of the navigation problem

According to the procedure described in subsection 2.3, the navigation problem is solved three times taking into account the measurements of different ground stations and different propagation methods.

CASE 1: Keplerian motion, Kourou measurements

From the *lsqnonlin* function, the following results are obtained:

$$\hat{\mathbf{x}} = \begin{bmatrix} 4661.551465 & [km] \\ 5261.783030 & [km] \\ 982.993740 & [km] \\ 0.105754 & [km/s] \\ -1.439591 & [km/s] \\ 7.372574 & [km/s] \end{bmatrix}, \qquad \overline{\text{residuals}} = \begin{bmatrix} 5.515116 & [-] \\ 3.964478 & [-] \\ 23.636532 & [-] \end{bmatrix} \tag{11}$$

Where $\hat{\mathbf{x}}$ is the solution of the minimum variance navigation problem and $\overline{\text{residuals}}$ is a vector containing the average of the weighted residuals of azimuth, elevation and range. Since the most precise evaluation of the spacecraft state at t_0 comes from the *sgp*4 propagation of the TLE, the latter is used as a comparison for the analysis of the navigation solution error. Moreover, the size of the 3σ covariance ellipsoid is evaluated from the maximum eigenvalue of the covariance matrix.

$$err = abs(\mathbf{x}_{sgp4} - \hat{\mathbf{x}}) = \begin{bmatrix} 23.777158 & [km] \\ 23.199971 & [km] \\ 59.441871 & [km] \\ 0.024049 & [km/s] \\ 0.116850 & [km/s] \\ 0.027963 & [km/s] \end{bmatrix}, \qquad \begin{aligned} 3\sqrt{\max(\lambda_i(P_r))} &= 27.373377\,\text{km} \\ 3\sqrt{\max(\lambda_i(P_v))} &= 0.194176\,\text{km/s} \end{aligned} \tag{12}$$

The reduced set of measurements obtained only from Kourou and the simplified Keplerian dynamics affect the outcome: the result is far from the *sgp*4 propagation at t_0 and the covariance ellipsoid is large, a signal of a bad minimum covariance solution.

CASE 2: Keplerian motion, Kourou and Svalbard measurements

The following outputs are obtained from *lsqnonlin*:

$$\hat{\mathbf{x}} = \begin{bmatrix} 4666.286591 \\ 5248.918353 \\ 1039.527935 \\ 0.092178 \\ -1.547744 \\ 7.350356 \end{bmatrix}, \qquad \overline{\text{residuals}} = \begin{bmatrix} 1.072696 & [-] \\ 2.361558 & [-] \\ 83.922432 & [-] \end{bmatrix} \tag{13}$$

The evaluation of the estimate error and the covariance leads to the following results:

$$err = abs(\mathbf{x}_{sgp4} - \hat{\mathbf{x}}) = \begin{bmatrix} 19.042032 & [km] \\ 10.335295 & [km] \\ 2.907676 & [km] \\ 0.010474 & [km/s] \\ 0.008697 & [km/s] \\ 0.005745 & [km/s] \end{bmatrix}, \qquad \begin{aligned} 3\sqrt{\max(\lambda_i(P_r))} &= 3.051879\,\text{km} \\ 3\sqrt{\max(\lambda_i(P_v))} &= 0.004412\,\text{km/s} \end{aligned} \tag{14}$$

Adding Svalbard's measurements, the solution improves providing a smaller size of the covariance ellipsoid. However, the Keplerian dynamics affects the result which is still quite far from $\mathbf{x}_{sgp4}$. Moreover, even if the residuals related to azimuth and elevation are lower than in the previous case, the one on the range shows an increased value, contradicting the expectations. In conclusion, even if Svalbard's measurements enhance the solution globally, the dynamics shall be modified adding the lacking most significant contributions.

CASE 3: Keplerian motion with J2 contribution, Kourou and Svalbard measurements

The following outputs are obtained from *lsqnonlin*:

$$\hat{\mathbf{x}} = \begin{bmatrix} 4685.425101 \\ 5238.586991 \\ 1042.560934 \\ 0.081572 \\ -1.556588 \\ 7.344494 \end{bmatrix}, \qquad \overline{\text{residuals}} = \begin{bmatrix} 0.833262 & [-] \\ 0.810598 & [-] \\ 0.792422 & [-] \end{bmatrix} \tag{15}$$

The evaluation of the estimate error and the covariance leads to the following results:

$$err = abs(\mathbf{x}_{sgp4} - \hat{\mathbf{x}}) = \begin{bmatrix} 0.096478 & [km] \\ 0.003932 & [km] \\ 0.125323 & [km] \\ 0.000133 & [km/s] \\ 0.000147 & [km/s] \\ 0.000117 & [km/s] \end{bmatrix}, \qquad \begin{aligned} 3\sqrt{\max(\lambda_i(P_r))} &= 0.057982\,\text{km} \\ 3\sqrt{\max(\lambda_i(P_v))} &= 0.000080\,\text{km/s} \end{aligned} \tag{16}$$

The addition of the J2 contribution to the Keplerian dynamics improves the outcome: the employment of the full set of measurements from both Kourou and Svalbard, in addition to a more refined dynamics, provides a very small covariance ellipsoid and lower residuals compared to the previous cases. As expected, the solution is close to $\mathbf{x}_{sgp4}$, meaning that J2 acceleration was the most important contribution lacking from the dynamical model. In conclusion, the third case provides the best outcome and the combination of both Kourou's and Svalbard's measurements with the modified Keplerian equations is applied to the analysis of Tango.

2.5 Point 5: Navigation problem of Tango

The visibility windows computation and measurements simulation described in subsection 2.1 and subsection 2.2 are repeated for Tango. The least squares (minimum covariance) solution is computed by applying the best combination selected for the Mango analysis: measurements from both ground stations and Keplerian dynamics with J2 contribution. The following outputs are obtained from *lsqnonlin*:

$$\hat{\mathbf{x}} = \begin{bmatrix} 4684.948240 \\ 5238.779845 \\ 1038.770938 \\ 0.083787 \\ -1.554209 \\ 7.345631 \end{bmatrix}, \qquad \overline{\text{residuals}} = \begin{bmatrix} 0.659981 & [-] \\ 0.724261 & [-] \\ 15.691983 & [-] \end{bmatrix} \tag{17}$$

The evaluation of the estimate error and the covariance leads to the following results:

$$err = abs(\mathbf{x}_{sgp4} - \hat{\mathbf{x}}) = \begin{bmatrix} 0.082778 & [km] \\ 0.009898 & [km] \\ 0.686153 & [km] \\ 0.000414 & [km/s] \\ 0.000336 & [km/s] \\ 0.000154 & [km/s] \end{bmatrix}, \qquad \begin{aligned} 3\sqrt{\max(\lambda_i(P_r))} &= 0.597077\,\text{km} \\ 3\sqrt{\max(\lambda_i(P_v))} &= 0.000847\,\text{km/s} \end{aligned} \tag{18}$$

As expected, the resulting state is quite similar to the one of Mango. The mean azimuth and elevation deviations are similar as well, whereas the mean range error and the uncertainties are larger with respect to the ones of Mango.

Exercise 3: Sequential filters

According to the Formation Flying In Orbit Ranging Demonstration experiment (FFIORD), PRISMA's primary objectives include testing and validation of GNC hardware, software, and algorithms for autonomous formation flying, proximity operations, and final approach and recede operations. The cornerstone of FFIORD is a Formation Flying Radio Frequency (FFRF) metrology subsystem designed for future outer space formation flying missions.

FFRF subsystem is in charge of the relative positioning of 2 to 4 satellites flying in formation. Each spacecraft produces relative position, velocity and line-of-sight (LOS) of all its companions.

You have been asked to track Mango to improve the accuracy of the estimate of its absolute state and then, according to the objectives of the PRISMA mission, validate the autonomous formation flying navigation operations by estimating the relative state between Mango and Tango by exploiting the relative measurements acquired by the FFRF subsystem. The Two-Line Elements (TLE) set of Mango and Tango satellites are reported in Tables 6 and 7 (and in WeBeep as 36599.3le, and 36827.3le).

The relative motion between the two satellites can be modelled through the linear, Clohessy-Wiltshire (CW) equations[†]

$$\begin{aligned} \ddot{x} &= 3n^2 x + 2n\dot{y} \\ \ddot{y} &= -2n\dot{x} \\ \ddot{z} &= -n^2 z \end{aligned} \tag{19}$$

where x, y,and z are the relative position components expressed in the LVLH frame, whereas n is the mean motion of Mango, which is assumed to be constant and equal to:

$$n = \sqrt{\frac{GM}{R^3}} \tag{20}$$

where R is the position of Mango at t_0.

The unit vectors of the LVLH reference frame are defined as follows:

$$\hat{\boldsymbol{i}} = \frac{\boldsymbol{r}}{r}, \quad \hat{\boldsymbol{j}} = \hat{\boldsymbol{k}} \times \hat{\boldsymbol{i}}, \quad \hat{\boldsymbol{k}} = \frac{\boldsymbol{h}}{h} = \frac{\boldsymbol{r} \times \boldsymbol{v}}{\|\boldsymbol{r} \times \boldsymbol{v}\|} \tag{21}$$

To perform the requested tasks you should:

1. *Estimate Mango absolute state.* You are asked to develop a sequential filter to narrow down the uncertainty on the knowledge of Mango absolute state vector. To this aim, you shall schedule the observations from the SVALBARD ground station[‡] reported in Table 5, and then proceed with the state estimation procedure by following these steps:

   (a) By using the mean state reported in Table 1 and by assuming Keplerian motion, predict the trajectory of the satellite over a uniform time grid (with a time step of 5 seconds) and compute the first visibility time window from the SVALBARD station in the time interval from t_0 = `2010-08-12T05:30:00.000` (UTC) to t_f = `2010-08-12T06:30:00.000` (UTC).

   (b) Use SGP4 and the provided TLE to simulate the measurements acquired by the SVALBARD station for the Mango satellite only. For doing it, compute the spacecraft position over the visibility window using a time-step of 5 seconds, and derive the associated expected measurements. Finally, simulate the measurements by adding a random error (assume a Gaussian model to generate the random error, with noise provided in Table 5).

[†]Notice that the system is linear, therefore it has an analytic solution of the state transition matrix $\boldsymbol{\Phi}$

[‡]Note that these are the same ones computed in Exercise 2

(c) 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. Plot the time evolution of the error estimate together with the 3σ of the estimated covariance for both position and velocity.

2. *Estimate the relative state.* To validate the formation flying operations, you are also asked to develop a sequential filter to narrow down the uncertainty on the knowledge of the relative state vector. To this aim, you can exploit the relative azimuth, elevation, and range measurements obtained by the FFRF subsystem, whose features are reported in Table 10, and then proceed with the state estimation procedure by following these steps:

   (a) Use SGP4 and the provided TLEs to propagate the states of both satellites at epoch t_0 in order to compute the relative state in LVLH frame at that specific epoch.

   (b) Use the relative state as initial condition to integrate the CW equations over the time grid defined in Point 1a. Finally, simulate the relative measurements acquired by the Mango satellite through its FFRF subsystem by adding a random error to the expected measurements. Assume a Gaussian model to generate the random error, with noise provided in Table 10.

   (c) Consider a time interval of 20 minutes starting from the first epoch after the visibility window (always with a time step of 5 seconds). Use an UKF to provide an estimate of the spacecraft relative state in the LVLH reference frame (in terms of mean and covariance) by sequentially processing the measurements acquired during those time instants in chronological order. Plot the time evolution of the error estimate together with the 3σ of the estimated covariance for both relative position and velocity.

3. *Reconstruct Tango absolute covariance.* Starting from the knowledge of the estimated covariance of the absolute state of Mango, computed in Point 1, and the estimated covariance of the relative state in the LVLH frame, you are asked to provide an estimate of the covariance of the absolute state of Tango. You can perform this operation as follows:

   (a) Pick the estimated covariance of the absolute state of Mango at the last epoch of the visibility window, and propagate it within the time grid defined in Point 2c.

   (b) Rotate the estimated covariance of the relative state from the LVLH reference frame to the ECI one within the same time grid.

   (c) Sum the two to obtain an estimate of the covariance of the absolute state of Tango. Plot the time evolution of the 3σ for both position and velocity and elaborate on the results.

Table 10: Parameters of FFRF.

Parameter	Value
Measurements noise $\sigma_{Az,El}$	1 deg
(diagonal noise matrix R) σ_{range}	1 cm

3.1 Point 1: Estimate Mango absolute state

a) Visibility window

Starting from $\hat{\mathbf{x}}_{0,sat1}$ at t_{sep}, the integration is performed with a Keplerian propagator to obtain the value of the mean state along the time grid. Using the procedure explained in subsection 2.1, the mean state is mapped into azimuth, elevation and range. Finally, the constraint on the elevation is evaluated, selecting only the feasible predicted measurements and consequently the predicted visibility window. The predicted measurements are shown in Figure 14, while Table 11 highlights the initial and final date of the visibility window.

Predicted visibility window	Initial date	Final date
First	2010 AUG 12 05:43:55.0000 UTC	2010 AUG 12 05:54:35.0000 UTC

Table 11: Svalbard first visibility window

b) Measurements simulation

The sgp4 propagation of the TLE is employed to simulate the real state along the time grid. The state is then mapped into azimuth, elevation and range; finally, a Gaussian distribution is employed to simulate the measurements noise, using the state propagated through sgp4 as mean value and Equation 9 as covariance. The actual measurements are represented in Figure 14, together with the predicted measurements.

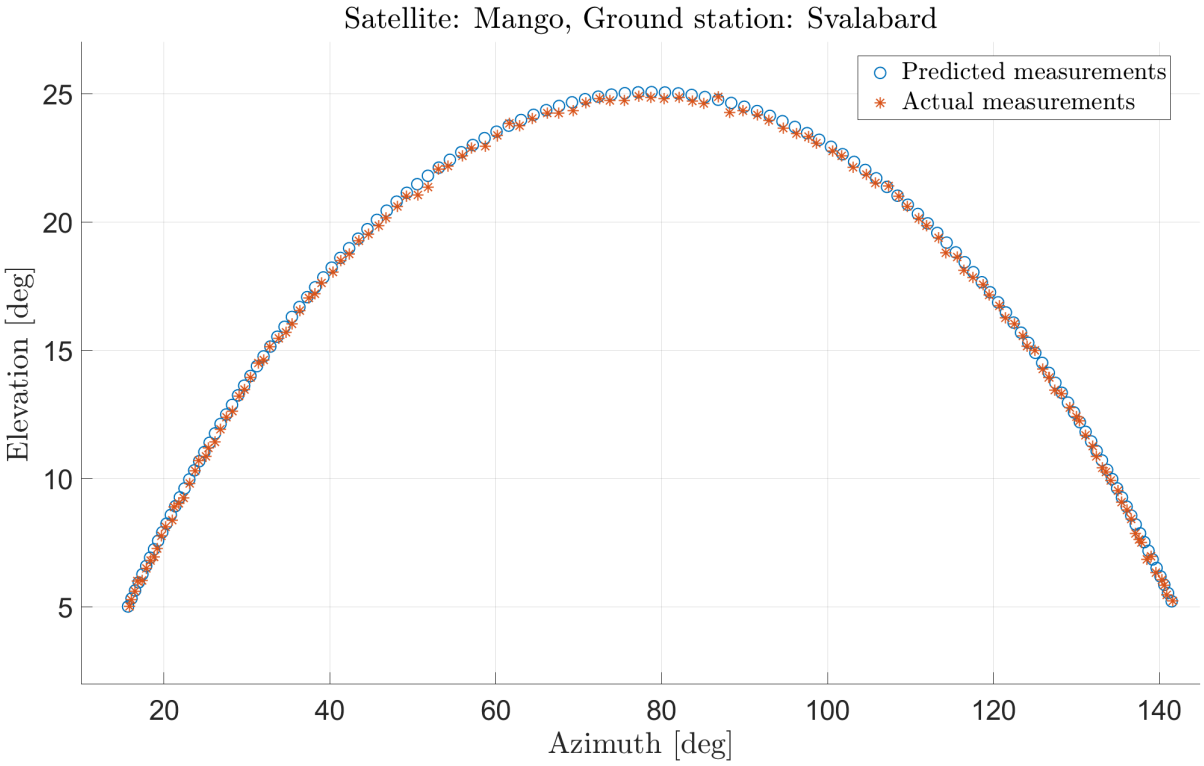

Figure 14: Predicted and actual measurements

In most cases, the actual counterpart of the last predicted measurement does not respect the constraint on the elevation; therefore, the actual visibility window does not coincide with the predicted visibility window, reducing of five seconds the final date.

Actual visibility window	Initial date	Final date
First	2010 AUG 12 05:43:55.0000 UTC	2010 AUG 12 05:54:30.0000 UTC

Table 12: Svalbard first visibility window

c) State estimate with UKF

In order to reduce the uncertainties on the state propagation, a UKF is applied. For the sequential application of the UKF, measurements are required. Therefore, the filter is applied inside the actual visibility window (see Table 12). An unscented transform approach is employed to propagate mean state and covariance from t_{sep} to five seconds before the first available measurement. The values obtained are used as the initial guess for the first UKF step. The procedure begins from the a posteriori state $\hat{x}^+_{k-1}$ and covariance P^+_{k-1} at instant $k-1$ and the current measurements y. Sigma points are computed, propagated using Keplerian dynamics with the addition of J2 perturbation and converted into the predicted measurements. Then, the a priori mean $\hat{x}^-_k$, the measurements mean $\hat{y}^-_k$, the a priori covariance P^-_k, the measurements covariance $P_{yy,k}$ and the cross-covariance $P_{xy,k}$ are calculated. With these, the Kalman gain K_k can be constructed and finally, the a posteriori state $\hat{x}^+_k$ and covariance P^+_k are found and serve as the output of the function. At each iteration, the size of the covariance is estimated with the formula $3\sqrt{trace(P)}$, while the error estimate is obtained by comparing the estimated Mango absolute state with the sgp4 propagation of the TLE. The Kalman filter is a sequential procedure

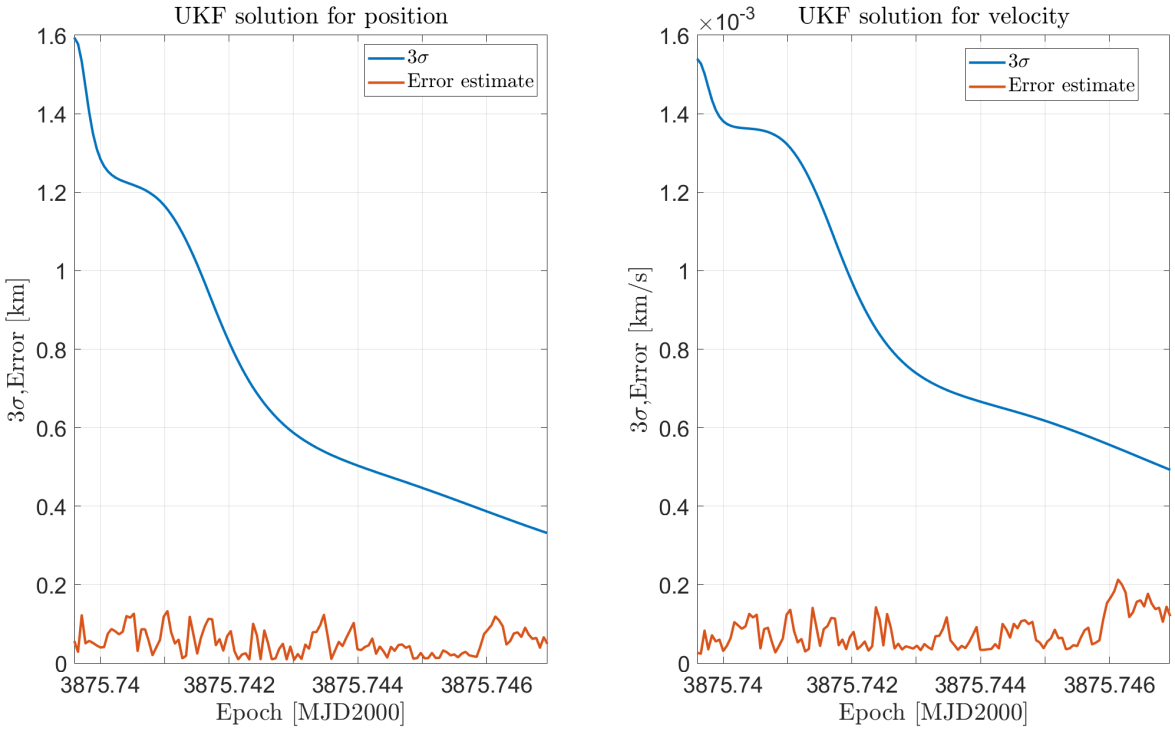

Figure 15: Error estimate and 3σ for Mango absolute state

of covariance minimization with a priori information; therefore, the resulting covariance should reduce in size along the process and, since the locus of the $99,7\%$ of the solutions is represented by the 3σ ellipsoid, the error estimate should stay under its limit value. Indeed, the images show clearly that the time evolution of the 3σ has a decreasing trend for both position and velocity, with an error estimate that is always lower.

3.2 Point 2: Estimate the relative state

a) Relative state in LVLH frame

The initial relative state is computed through the following procedure:

1. **Relative state of Tango in the ECI frame**: The states of Mango and Tango at t_0 in the ECI frame are computed through the sgp4 propagation of the TLEs. The relative state of Tango with respect to Mango is obtained by applying a vectorial difference.

$$\mathbf{x}_{t_0,rel,ECI} = \mathbf{x}_{t_0,sat2,ECI} - \mathbf{x}_{t_0,sat1,ECI} \tag{22}$$

2. **Computation of the rotation matrix**: the rotation matrix from ECI to LVLH has the following appearance:

$$A_{eci\rightarrow lvlh} = \begin{bmatrix} R & 0 \\ \dot{R} & R \end{bmatrix} \qquad R = \left[\hat{\boldsymbol{i}}; \hat{\boldsymbol{j}}; \hat{\boldsymbol{k}}\right] \qquad \dot{R} = \left[\frac{d\hat{\boldsymbol{i}}}{dt}; \frac{d\hat{\boldsymbol{j}}}{dt}; \frac{d\hat{\boldsymbol{k}}}{dt}\right] \tag{23}$$

   Where $\hat{i}, \hat{j}, \hat{k}$ are calculated as explained in Equation 21, while their derivatives are computed as follows:

$$\begin{cases} \dfrac{d\hat{\boldsymbol{i}}}{dt} = \dfrac{1}{||\mathbf{r}||}\left(\mathbf{v} - (\boldsymbol{i}\cdot\mathbf{v})\boldsymbol{i}\right) \\ \dfrac{d\hat{\boldsymbol{j}}}{dt} = \hat{\boldsymbol{k}} \times \dfrac{d\hat{\boldsymbol{i}}}{dt} \\ \dfrac{d\hat{\boldsymbol{k}}}{dt} = [0,0,0] \end{cases} \tag{24}$$

3. **Rotation from ECI to LVLH frame**: the rotation matrix is employed to obtain the relative state of Tango in the LVLH frame.

$$\mathbf{x}_{t_0,rel,LVLH} = A_{eci\rightarrow lvlh} \cdot \mathbf{x}_{t_0,rel,ECI} \tag{25}$$

b) Measurements simulation with Clohessy-Wiltshire (CW) equations

Starting from the initial value described in Equation 25, the relative state is computed by integrating Clohessy-Wiltshire (CW) equations from t_0 to t_f. For each element of the time grid, the ideal measurements are calculated through the function *cspice_reclat*; finally, Equation 9 is employed with the new values supplied by the text, to add noise to the ideal measurements. For this purpose, a Gaussian distribution is applied.

c) Relative state estimate with UKF

The last date of the actual visibility window (Table 12) is the starting point for the application of the UKF. The relative state propagated with CW equations until this date is used as the initial mean value, while the initial covariance is $P_0 = diag([0.01, 0.01, 0.1, 0.0001, 0.0001, 0.001])$. The relative state is processed along the given time grid and the solution is evaluated in terms of 3σ and error estimate; the 3σ is computed as $3\sqrt{trace(P)}$, where P is the covariance matrix associated to position/velocity, while the error estimate is obtained as the norm of the difference between the state estimated with the unscented Kalman filter and the state propagated through CW equations. Figure 16 shows the results of the procedure. In Figure 17 the same results are represented, but with a *semilogy* plot, in order to better highlight the difference between the 3σ and the error estimate. As expected, the 3σ has a decreasing trend, keeping a value higher than the error estimate.

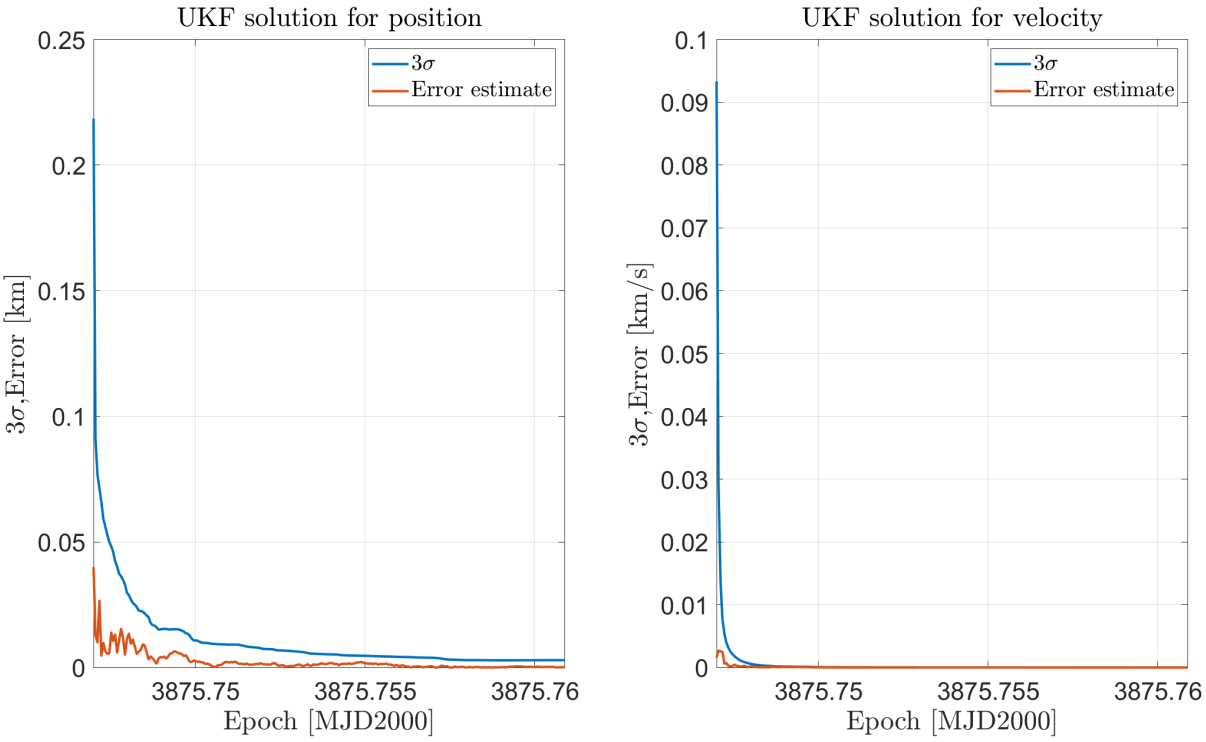

Figure 16: Error estimate and 3σ for Tango relative state

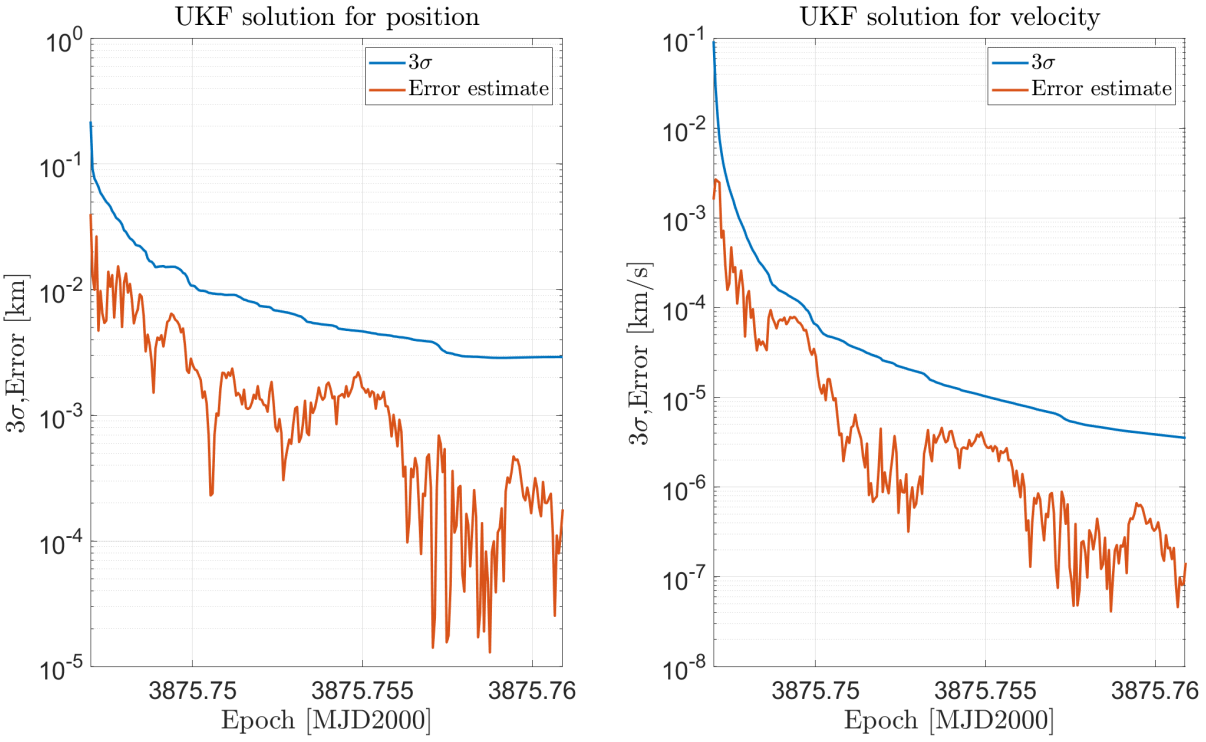

Figure 17: Error estimate and 3σ for Tango relative state (semilogy)

3.3 Point 3: Reconstruct Tango absolute covariance

The procedure is:

a. Starting from the mean state and covariance calculated at the last date of the actual visibility window as described in subsection 3.1, an unscented transform approach is employed

to propagate the absolute covariance of Mango.

b. The covariance of the relative state of Tango is rotated from the LVLH to the ECI frame, pre-multiplying the covariance for the rotation matrix from LVLH to ECI and post-multiplying for its transpose.

$$A_{lvlh \to eci} = \begin{bmatrix} R^T & 0 \\ \dot{R}^T & R^T \end{bmatrix} \tag{26}$$

c. The two previous contributions are summed up to obtain the covariance of the absolute state of Tango. The time evolution of the 3σ is represented in Figure 18, both for position and velocity.

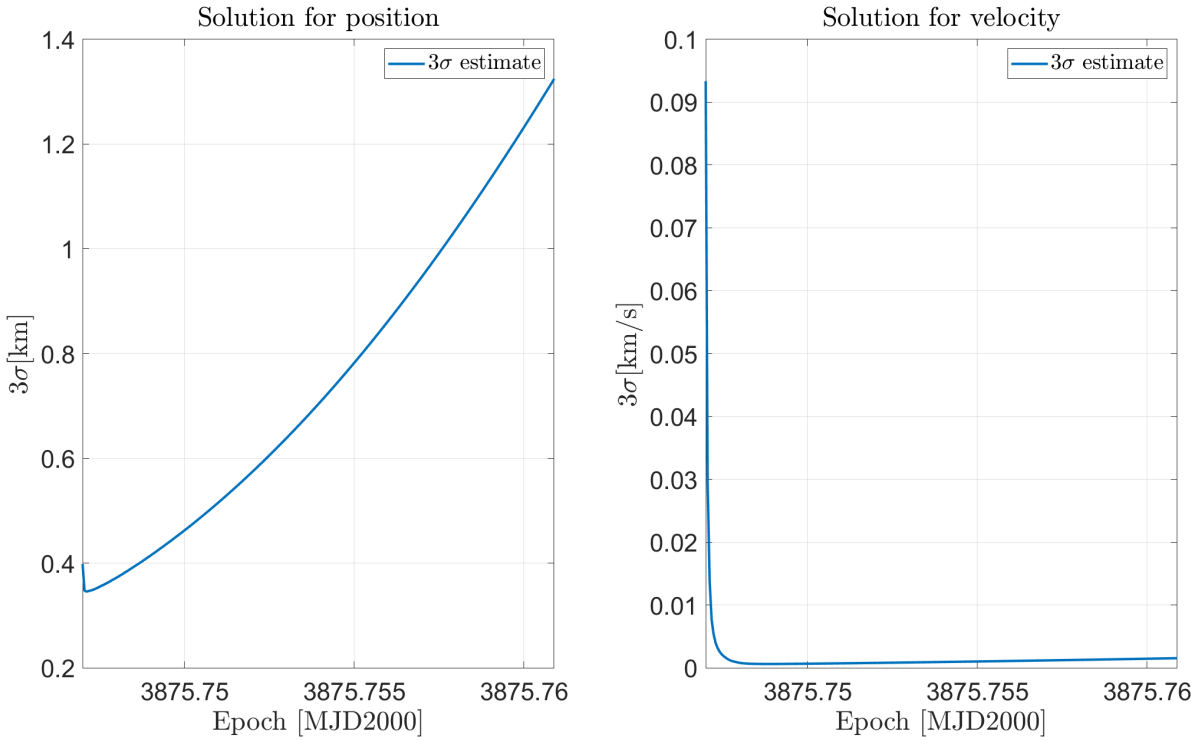

Figure 18: Error estimate and 3σ for Tango absolute state

The time evolution of the covariance related to the absolute position of Tango is characterized by an initial reduction, followed by an increasing behavior. This is due to the fact that Tango covariance is reconstructed by summing two contributions: the covariance related to the absolute state of Mango grows in time since it is computed through uncertainties propagation (with UT approach). Instead, the covariance associated with the relative state of Tango decreases in time due to the employment of the unscented Kalman filter; however, after a fast initial drop, it does not change a lot. Therefore, initially, the second component prevails, but, when the decreasing rate of the covariance associated with the relative state of Tango is low, the first element takes over. The same considerations can be made for the covariance of the velocity, even if it is not so clear from the image: after an initial reduction the curve starts growing.