SGN – Assignment #2

Davide Iafrate, 962657

Exercise 1: Uncertainty propagation

You are the operator of the sensor network composed by the stations reported in Table 1. You have been assigned the task of tracking the spacecraft GIOVE-A (NORAD ID: 28922, INT.DES.: 2005-051A) and you have been provided with an estimate of the spacecraft state at time t_0 in terms of its mean and covariance, as reported in Table 2. Assume Keplerian motion can be used to model the spacecraft dynamics.

1. By using the mean state to predict the reference trajectory over a uniform time grid (one point per minute), compute all the visibility time windows from the available stations in the time interval from t_0 to t_f = `2021-11-23T14:00:00` (UTC).

2. Propagate the initial mean and covariance to the last epoch of each visibility window (as computed with the uniform time grid in the previous point) using both a linearized approach and the unscented transform. Elaborate on the results of the uncertainty propagation, including possible differences between the two approaches.

3. Perform a Monte Carlo simulation (using at least 100 samples drawn from the initial covariance) to predict the spacecraft state distribution at the very last visibility epoch (the closest to t_f, as resulting from the uniform time grid used at point 1). Compute the sample mean and sample covariance, and compare them with the results obtained at the previous point. Then, for each sample, compute the angle between the topocentric direction of the satellite and the topocentric direction of the center of the field-of-view (FoV) of the sensor. Provide the percentage of the samples that lie inside the sensor FoV.

Table 1: Sensor network for GIOVE-A observation: list of stations, including their features.

Station name	Milano	Wellington	La Silla
Coordinates	LAT = 45.50122° LON = 9.15461° ALT = 20 m	LAT = -41.28437° LON = 174.76697° ALT = 117 m	LAT = -29.26117° LON = -70.73133° ALT = 2400 m
Type	Radar (monostatic)	Optical	Optical
Provided measurements	Az, El [deg] Range (one-way) [km]	Ra, Dec [deg]	Ra, Dec [deg]
Measurements noise (diagonal noise matrix R)	$\sigma_{Az,El}$ = 100 mdeg σ_{range} = 0.01 km	$\sigma_{Ra,Dec}$ = 0.5 mdeg	$\sigma_{Ra,Dec}$ = 1 mdeg
FoV (circular)	6 deg	2 deg	1 deg
Minimum elevation	30 deg	20 deg	10 deg

Table 2: Estimate of the GIOVE-A state at t_0 provided in ECI J2000.

Parameter	Value
Ref. epoch t_0 [UTC]	2021-11-19T14:25:39.652
Mean state $\hat{\boldsymbol{x}}_0$ [km, km/s]	$\hat{\boldsymbol{r}}_0 =$ [23721.610, 17903.673, -49.918] $\hat{\boldsymbol{v}}_0 =$ [-1.150987, 1.529718, 3.122389]
Covariance P_0 [km^2, km^2/s, km^2/s^2]	$\begin{bmatrix} +2.6e-2 & +1.4e-2 & -1.8e-3 & 0 & 0 & 0 \\ +1.4e-2 & +1.8e-2 & +2.3e-3 & 0 & 0 & 0 \\ -1.8e-3 & +2.3e-3 & +1.0e-2 & 0 & 0 & 0 \\ 0 & 0 & 0 & +1.6e-7 & 0 & 0 \\ 0 & 0 & 0 & 0 & +1.6e-7 & 0 \\ 0 & 0 & 0 & 0 & 0 & +1.6e-7 \end{bmatrix}$

Typically in a navigation problem there is a reference state $\hat{\boldsymbol{x}}_0^*$ at the epoch t_0. The reference is used to plan additional observations and the uncertainty is propagated to make sure that the observations are feasible (i.e. they fall within the sensor FoV).

Point 1 - Visibility windows prediction

For each station the position of the spacecraft is converted from the propagated ECI coordinates to the topocentric coordinates. The Azimuth and Elevation angles are computed and the visibility is checked by comparing the Elevation angle to the minimum elevation angle of each station, reported in Tab.[1]. The elevation of the spacecraft as seen in each visibility window is shown in Fig.[1].

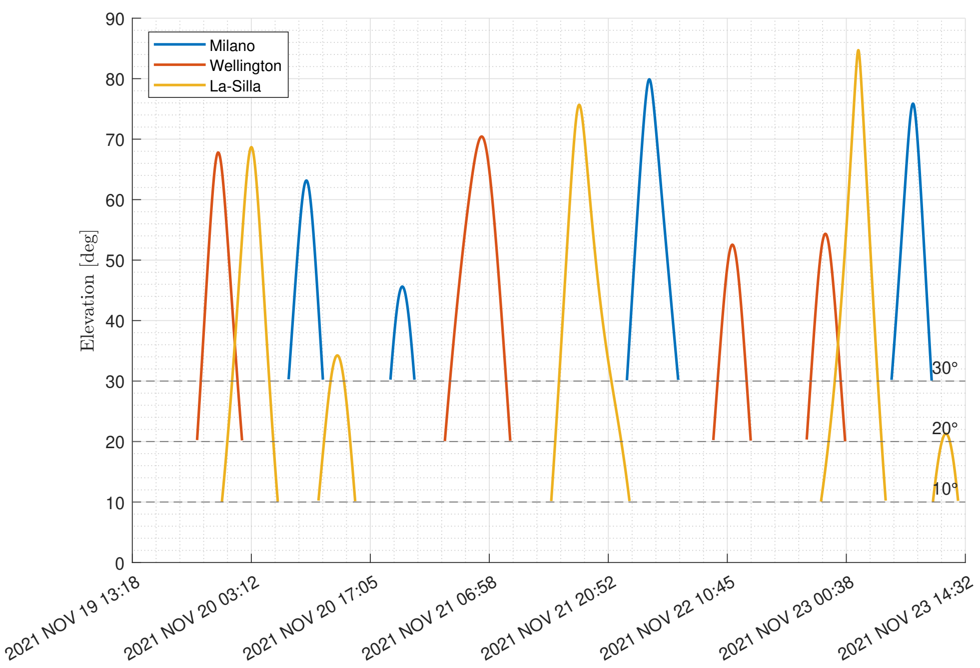


Figure 1: Ground station network visibility windows

Point 2 - Mean state and covariance propagation

The initial mean $\hat{\boldsymbol{x}}_0$ and covariance P_0 are propagated both using the linear method and the unscented transform.
In the linear method a linearization of the dynamics $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ is performed. The resulting orbital mean state and covariance are then shown to be:

$$\hat{\mathbf{x}}(t) = \mathbf{x}^*(t) \tag{1}$$
$$P(t) = \Phi(t_0, t) P_0 \Phi(t_0, t)^T \tag{2}$$

where $\Phi(t_0, t)$ is the State Transition Matrix (STM). At the final instant of time of the last visibility window the mean state and covariance are:

$$\hat{\mathbf{x}}(t_f) = \mathbf{x}^*(t_f) \tag{3}$$
$$P(t_f) = \Phi(t_0, t_f)\, P_0\, \Phi(t_0, t_f)^T \tag{4}$$

In the unscented transform instead the initial distribution is modeled through so-called Sigma points:

$$\begin{cases} \boldsymbol{\chi_0} = \hat{\mathbf{x}} \\ \boldsymbol{\chi_i} = \hat{\mathbf{x}} + (\sqrt{(n+\lambda)P_0})_i \quad for\ i = 1:n \\ \boldsymbol{\chi_i} = \hat{\mathbf{x}} - (\sqrt{(n+\lambda)P_0})_i \quad for\ i = n+1:2n \end{cases} \tag{5}$$

with:

- $\lambda = \alpha^2(n+k) - n$ the scaling parameter
- $\alpha = 10^{-3}$ describing the spread of the Sigma points
- k=0 usually

taken to be representative of the initial statistical distribution. These points are then propagated through the two body problem flow $\phi(t_0, \mathbf{x}_0; t)$ and then from the propagated sigma points the final weighted sample mean and covariance are computed. The weights are:

$$W_0^{(m)} = \lambda/(n+\lambda) \tag{6}$$
$$W_0^{(c)} = \lambda/(n+\lambda) + (1 - \alpha^2 + \beta) \tag{7}$$
$$W_i^{(m)} = W_i^{(c)} = 1/[2(n+\lambda)] \;\; i = 1, ..., 2n \tag{8}$$

The propagated Sigma points (using the flow) are:

$$\boldsymbol{\mathcal{Y}}_i = \mathbf{g}(\boldsymbol{\chi_i}) \tag{9}$$

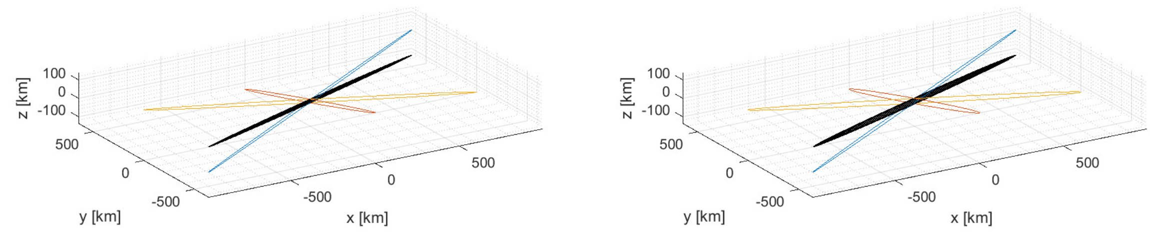
and finally the weighted sample mean and covariance are:

$$\hat{\mathbf{y}} = \sum_{i=01}^{2n} W_i^{(m)} \boldsymbol{\mathcal{Y}_i} \tag{10}$$

$$P_y = \sum_{i=01}^{2n} W_i^{(c)} [\boldsymbol{\mathcal{Y}_i} - \hat{\mathbf{y}}][\boldsymbol{\mathcal{Y}_i} - \hat{\mathbf{y}}]^T \tag{11}$$

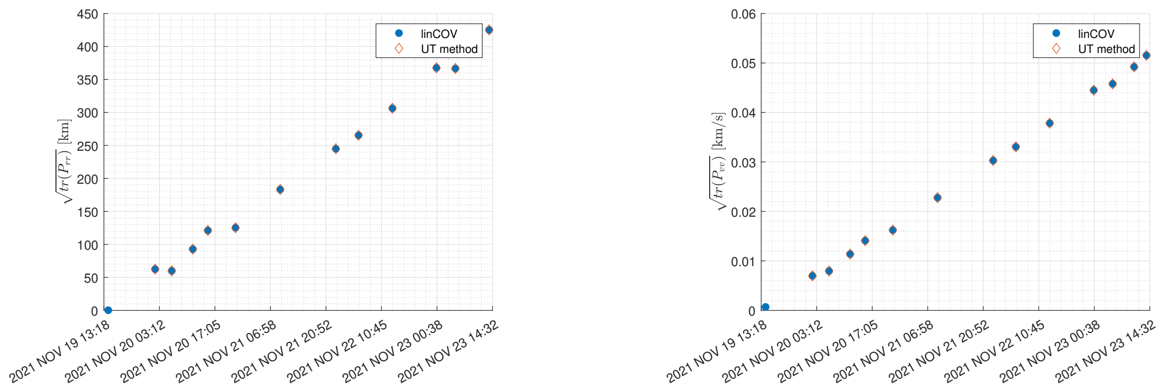
The linear method and unscented transform position uncertainty ellipsoids are shown in Fig.[2]. It can be seen that they are similar in size, however the UT method results in a higher covariance along the z-axis.

In Fig.[3] is shown the evolution in time in the uncertainty both in position (trace of the P_{rr} submatrix) and velocity (trace of the P_{vv} submatrix) using both the linCOV method and the UT.



(a) LinCOV linear method (b) Unscented transform

Figure 2: positional uncertainty ellipsoids 95% confidence



(a) Position (b) Velocity

Figure 3: Time evolution of the uncertainty in the spacecraft state at the end of each visibility window

Point 3 - Montecarlo and FoV

The spacecraft state at the final instant of the visibility windows ($t_f^{visibility}$ ='2021 NOV 23 13:41:30.469') is estimated through a montecarlo simulation, by propagating $n = 300$ points through the nonlinear flow and computing the resulting mean and covariance. The uncertainty ellipsoid is shown in Fig.[4].

The final visibility window corresponds to measurements taken from the La Silla ground station. To estimate the probability of being able to track the spacecraft at this final instant the statistical distribution of the topocentric angle between the mean reference state resulting from the montecarlo simulation and each realization is computed. This is simply done as:

$$\alpha_i = acos(\hat{\mathbf{r}}_{mean}^{TOPO} \cdot \hat{\mathbf{r}}_i^{TOPO}) \tag{12}$$

If this angle is less than half the FoV the point is inside the FoV. The resulting distribution is shown in Fig.[5] revealing a probability of around 50% of being able to succesfully take measurements of the spacecraft.

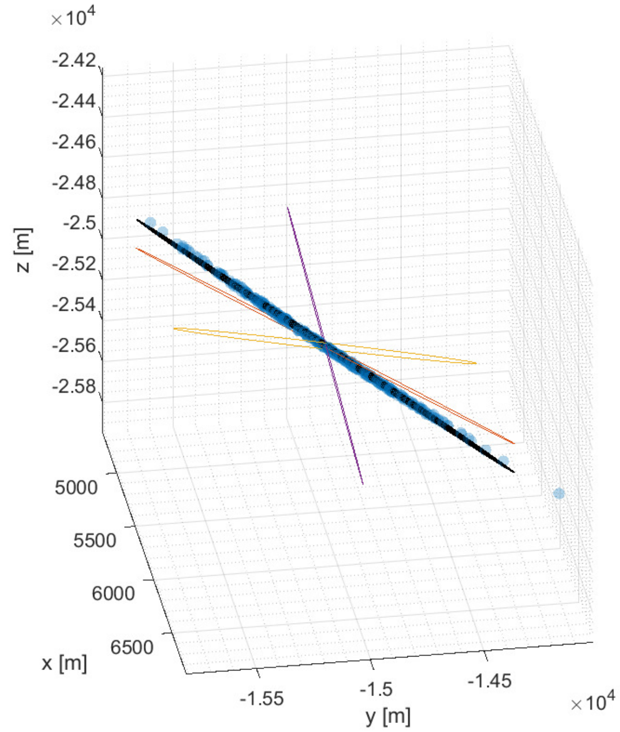


Figure 4: Montecarlo simulation final points and uncertainty ellipsoid 95% confidence region

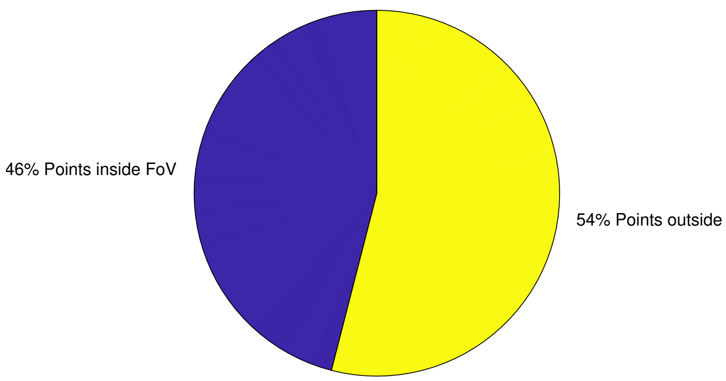


Figure 5: Probability of spacecraft being inside FoV

Exercise 2: Batch filters

The Two-Line Elements (TLE) set of the spacecraft GIOVE-A of Exercise 1 is reported in Table 3 (and in WeBeep as 28922.tle).

1. *Simulate measurements.* Use SGP4 and the provided TLE to simulate the measurements acquired by the sensor network in Table 1 by:
   (a) Computing the spacecraft position over the same visibility windows identified in Exercise 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 1).
2. *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 mean state provided in Table 2 as first guess;
      - the simulated measurements up to epoch $t_{f,1}$ =`2021-11-21T14:00:00`;
      - pure Keplerian motion and J2-perturbed motion to model the spacecraft dynamics.
   (b) Repeat step 2a by solving the navigation problem with the a priori information reported in Table 2.
   (c) Repeat step 2a by adding incrementally the simulated measurements up to $t_{f,2}$ =`2021-11-22T14:00:00` and $t_{f,3}$ =`2021-11-23T14:00:00`.

Table 3: TLE of spacecraft GIOVE-A.

`1␣28922U␣05051A␣␣␣21323.60115338␣-.00000088␣␣00000-0␣␣00000+0␣0␣␣9998`
`2␣28922␣␣58.3917␣␣37.3090␣0004589␣␣75.2965␣284.7978␣␣1.69421077␣98469`

Measurements Simulation

The acquired measurements are simulated from the SGP4 propagator, computing the propagated state of the spacecraft every 60s and then using a measurement function to return the appropriate measurements, i.e. either Azimuth and Elevation angles plus range or Right Ascension and Declination. Then a random error is added to the measurements by using MATLAB's `mvnrnd` function, and as covariance matrix the R matrix of the measurements of each station:

$$R_{Milano} = \begin{bmatrix} \sigma_{Az}^2 & 0 & 0 \\ 0 & \sigma_{El}^2 & 0 \\ 0 & 0 & \sigma_{range}^2 \end{bmatrix} \tag{13}$$

$$R_{Wellington,LaSilla} = \begin{bmatrix} \sigma_{Ra}^2 & 0 \\ 0 & \sigma_{Dec}^2 \end{bmatrix} \tag{14}$$

Each measurement is computed through the measuring function as:

$$\mathbf{y}_i = \mathbf{h}(\mathbf{x}_i) \tag{15}$$

The measuring function outputs two types of measurements, depending on the station acquiring them through the visibility window:

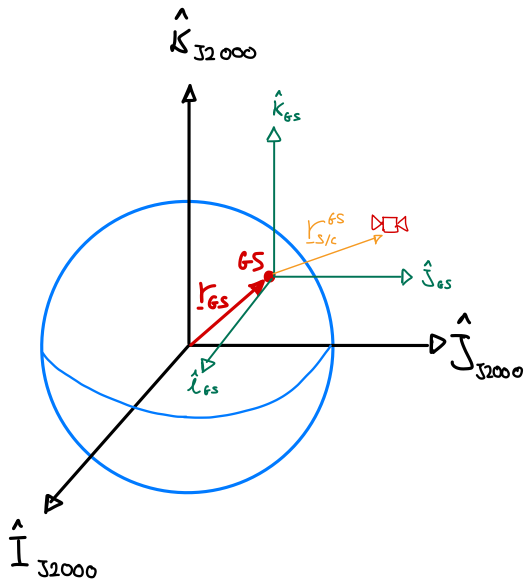


Figure 6: Ground station topocentric J2000 reference frame

- Azimuth, Elevation and Range (Milan ground station): The Azimuth and Elevation angles are computed in the topocentric reference frame of the ground station.
- Right Ascension and Declination (Wellington and La Silla). Due to the proximity of the spacecraft to the measuring station the RA and DEC are the **local** ones, taking as reference frame the ground station topocentric J2000 frame as in Fig.[6].

  The position vector used to compute the RADEC measurements is then:

$$\mathbf{r}_{S/C}^{GS} = \mathbf{r}_{S/C} - \mathbf{r}_{G/S} \tag{16}$$

$$\tag{17}$$

The error is computed from the reference SGP4-propagated measurements (simulating the actual acquired measurements) $\tilde{\mathbf{y}}$ and by using the flow of both unperturbed and perturbed 2 body problem ($\mathbf{h}(\mathbf{x}_i)$):

$$\mathbf{x}_i = \phi(t_0, \mathbf{x}_k; t_i) \tag{18}$$

as:

$$\boldsymbol{\epsilon}_i = \tilde{\mathbf{y}} - \mathbf{y}_i \tag{19}$$

Weighted Least Square solution to the navigation problem

The weighted least square solution to the navigation problem tries to find the initial state $\hat{\mathbf{x}}_k$ that, when propagated, minimize the cost function:

$$J = \frac{1}{2}\boldsymbol{\epsilon} W \boldsymbol{\epsilon}^T \tag{20}$$

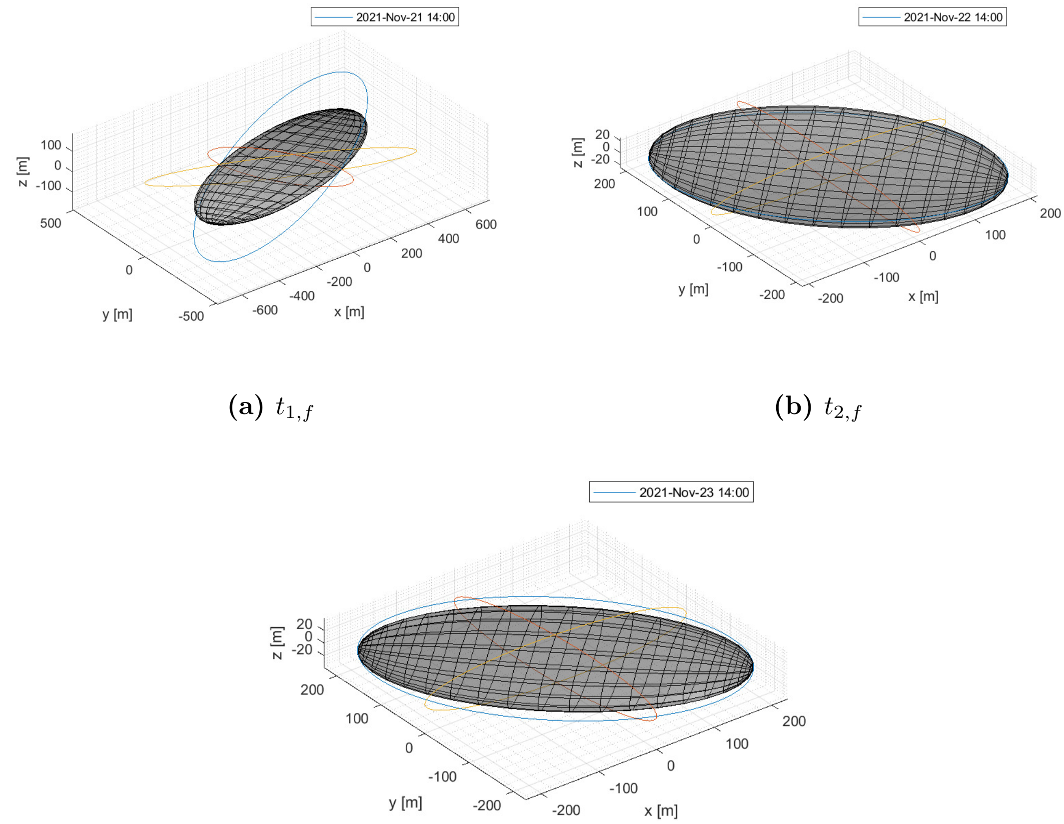
$$W = R_{station}^{-1} \tag{21}$$

where W is a diagonal matrix of weights defined in 21. The problem is then solved through MATLAB's `lsqnonlin`, which takes as input the residual computed as:

$$\mathbf{res} = \sqrt{W}\boldsymbol{\epsilon} \tag{22}$$

The problem is solved using both an unperturbed flow (Fig.[7]) and a J2 perturbed flow (Fig.[8]).

It can be seen that in all cases the measurements help to reduce a lot the size of the uncertainty ellipse. In particular the J2 perturbed flow solution achieves a very good precision.



(a) $t_{1,f}$

(b) $t_{2,f}$

(c) $t_{3,f}$

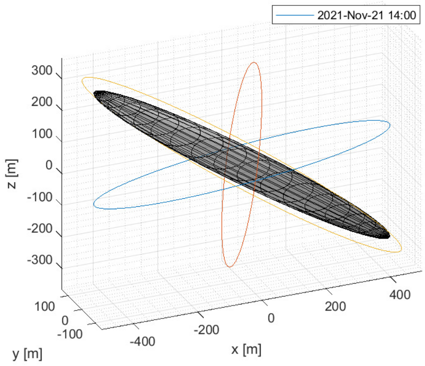
Figure 7: Covariance position ellipsoids
Unperturbed flow solution

In order to improve further the estimate of the state and shrink the confidence ellipsoid it is possible to add a priori information to the least square estimation. This consists simply in adding a further residual term, based on the a priori known covariance matrix P_0. The residual term is thus:

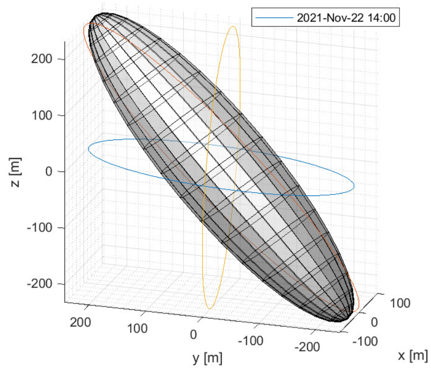
$$\mathbf{res} = \begin{bmatrix} \sqrt{P_0^{-1}}(\hat{\mathbf{x}}_k - \mathbf{x}_0) \\ \sqrt{W}\epsilon \end{bmatrix} \tag{23}$$

where $\hat{\mathbf{x}}_k$ is the initial state to be found and $\mathbf{x}_0$ is the a priori mean state.

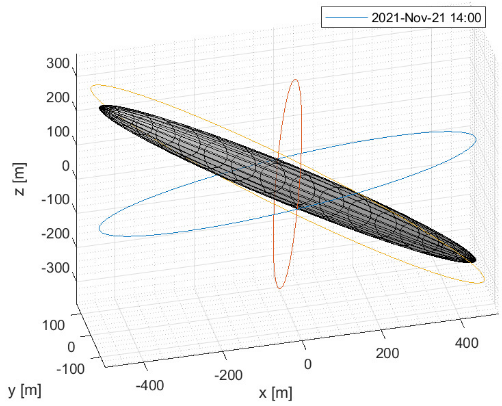
It is possible to see the improvement in Fig.[9].



(a) $t_{f,1}$

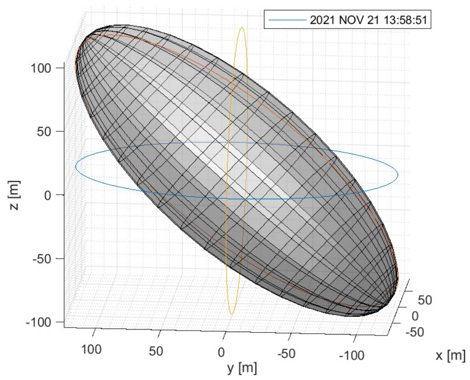


(b) $t_{f,2}$

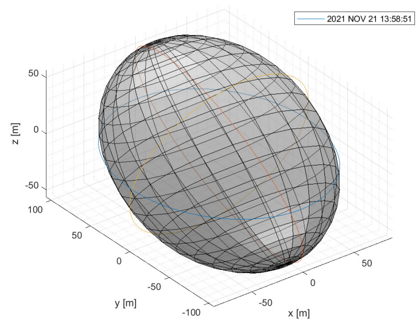


(c) $t_{f,3}$

Figure 8: Covariance position ellipsoids
J2 Perturbed flow solution



(a) J2 perturbed 2BP flow



(b) Unperturbed 2BP flow

Figure 9: Covariance position ellipsoids at $t_{f,1}$
with a priori information

Exercise 3: Sequential filters

The spacecraft of the ExoMars 2016 mission is approaching Mars. A retargeting maneuver is planned on 2016-10-17 to insert the Trace Gas Orbiter module into its target orbit around Mars after releasing the Schiaparelli EDM lander. A one-week tracking campaign is executed to narrow down the uncertainty on the spacecraft state before the retargeting maneuver. You have just received the measurements acquired by the two stations reported in Table 4. These are stored in the form of TDMs and are available on WeBeep, in the subfolder `tdm` contained in `Assignment02.zip`.

Based on previous tracking campaigns, you are provided with the initial orbital estimate of the ExoMars spacecraft reported in Table 5, in terms of its mean and covariance. By assuming that a heliocentric Keplerian motion can be used to model the spacecraft dynamics along the entire tracking window:

1. Use an unscented Kalman filter (UKF) to update sequentially the spacecraft state (in terms of mean and covariance) by processing the acquired measurements in chronological order.
2. Compute the error along the tracking campaign between the estimated mean states and the true trajectory provided in the file `exomars.bsp` available on WeBeep (target ID = -143). Check the consistency between the covariance matrices estimated by the filter during the tracking campaign and the computed error.
3. Compute the square roots of the traces of the position and velocity covariance submatrices at the last measurement epoch available in the TDMs. Check that their values are smaller than 75 km and 2E–2 km/s, respectively.

Table 4: Sensor network for ExoMars observation: list of stations, including their features.

Station name	Malargüe	New Norcia
Coordinates	LAT = -35.77601° LON = -69.39819° ALT = 1550 m	LAT = -31.04823° LON = 116.19147° ALT = 252 m
Type	Radar (monostatic)	Radar (monostatic)
Minimum elevation	20 deg	20 deg
Provided measurements	Az, El [deg] Range (one-way) [km]	Az, El [deg] Range (one-way) [km]
Measurements noise (noise matrix R)	σ_{Az} = 1.5 mdeg σ_{El} = 1.3 mdeg σ_{range} = 75 m $\rho_{Az-El} = 0.1$ $\rho_{Az-rng} = \rho_{El-rng} = 0$	σ_{Az} = 1.5 mdeg σ_{El} = 1.3 mdeg σ_{range} = 75 m $\rho_{Az-El} = 0.1$ $\rho_{Az-rng} = \rho_{El-rng} = 0$

Table 5: Estimate of the ExoMars spacecraft state at the beginning of the tracking campaign, provided in the Sun-centric J2000 reference frame.

Parameter	Value
Ref. epoch t_0 [UTC]	2016-10-10T00:00:00.000
Mean state $\hat{\boldsymbol{x}}_0$ [km, km/s]	$\hat{\boldsymbol{r}}_0 = [+1.68467660241\text{E}{+}08 \quad -1.07050886902\text{E}{+}08 \quad -5.47243873455\text{E}{+}07]$ $\hat{\boldsymbol{v}}_0 = [+1.34362486580\text{E}{+}01 \quad +1.68723391839\text{E}{+}01 \quad +8.66147058235\text{E}{+}00]$
Covariance P_0 [km^2, km^2/s, km^2/s^2]	$\begin{bmatrix} +2.01\text{E}{+}04 & -7.90\text{E}{+}00 & -4.05\text{E}{+}00 & -5.39\text{E}{-}03 & +6.37\text{E}{-}06 & +3.26\text{E}{-}06 \\ -7.90\text{E}{+}00 & +2.01\text{E}{+}04 & +2.64\text{E}{+}00 & +6.37\text{E}{-}06 & -5.38\text{E}{-}03 & -2.07\text{E}{-}06 \\ -4.05\text{E}{+}00 & +2.64\text{E}{+}00 & +2.01\text{E}{+}04 & +3.25\text{E}{-}06 & -2.03\text{E}{-}06 & -5.38\text{E}{-}03 \\ -5.39\text{E}{-}03 & +6.37\text{E}{-}06 & +3.25\text{E}{-}06 & +1.92\text{E}{-}07 & -2.28\text{E}{-}09 & -1.16\text{E}{-}09 \\ +6.37\text{E}{-}06 & -5.38\text{E}{-}03 & -2.03\text{E}{-}06 & -2.28\text{E}{-}09 & +1.91\text{E}{-}07 & +7.31\text{E}{-}10 \\ +3.26\text{E}{-}06 & -2.07\text{E}{-}06 & -5.38\text{E}{-}03 & -1.16\text{E}{-}09 & +7.31\text{E}{-}10 & +1.91\text{E}{-}07 \end{bmatrix}$

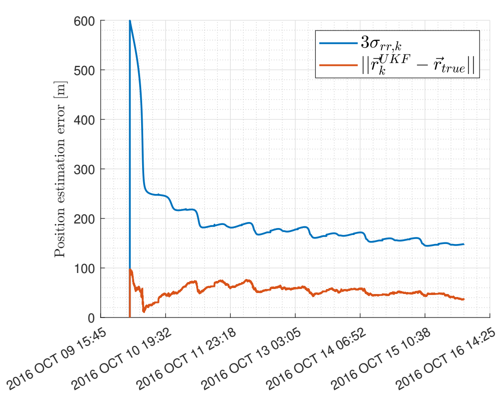
The Unscented Kalmand Filter (UKF) uses the Unscented Transform (UT) to predict mean and covariance of both the state and the measurements at a certain time.

The algorithm is, at each time step t_k, the following:

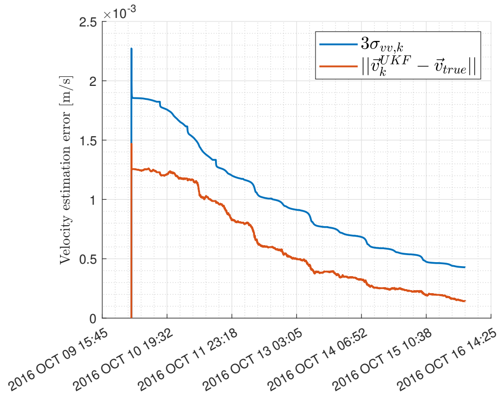
1. Compute the sigma points $\boldsymbol{\chi_{i,k-1}}$ and the weights $W_i^{(m)}$, $W_i^{(c)}$ for mean and covariance respectively.
2. Perform the prediction step:
   - Propagate the sigma points to time t_k: $\boldsymbol{\chi_{i,k}} = \phi(t_0, \boldsymbol{\chi_{i,k-1}}; t_k)$
   - Compute the a priori weighted mean state: $\hat{\mathbf{x}}_k^-$
   - Compute the a priori weighted covariance: P_k^-
   - Compute the measurements of the Sigma points $\mathcal{Y} = \mathbf{h}(\boldsymbol{\chi_{i,k}})$
   - Compute the a priori measurements mean $\hat{\mathbf{y}}_\mathbf{k}^-$
3. Perform the update step:
   - Compute the weighted measurements covariance $P_{yy,k}$ (adding also the measurement noise matrix R_k) and cross-covariance $P_{xy,k}$
   - Compute the gain: $K_k = P_{xy,k} P_{yy,k}^{-1}$
   - Compute the a posteriori mean and covariance:
     $$\begin{cases} \hat{\mathbf{x}}_\mathbf{k}^+ = \hat{\mathbf{x}}_\mathbf{k}^- + K_k(\mathbf{y}_k - \hat{\mathbf{y}}_k^-) \\ P_k^+ = P_k^- - K_k P_{yy,k} K_k^T \end{cases}$$

   The error is compared between the real error, computed as the difference between the mean estimated state $\mathbf{x}_k$ and the true trajectory provided in the file, and the error estimated by the filter, computed from the covariance matrix as the square root of the trace of the position and velocity covariance submatrices. The two errors, for both position and velocity are plotted respectively in Fig.[10a] and Fig.[10b] and they show that the filter is quite good at estimating the actual error that is present.

   The final uncertainty ellipsoid is shown then in Fig.[11] with the quantities to be computed reported in Tab.[6]. Both are below the required precision level, thus the UKF works correctly and improves the state accuracy.



(a) Position error comparison



(b) Velocity error comparison

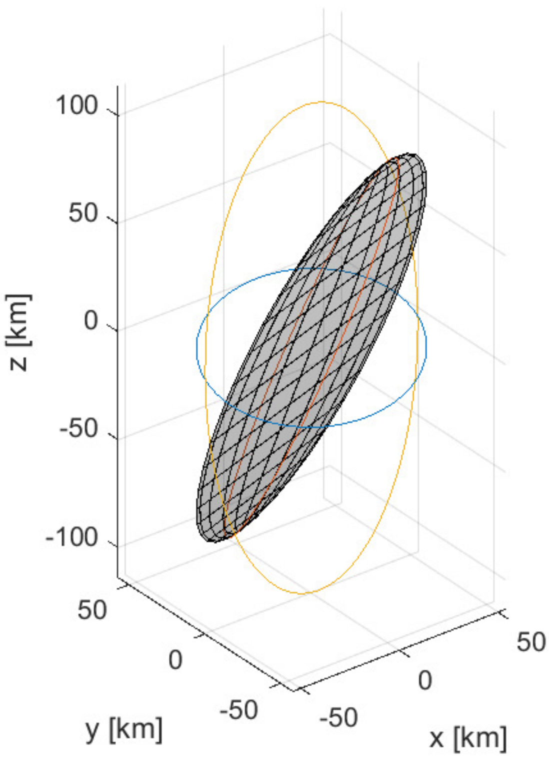


Figure 11: Final estimate uncertainty ellipsoid (95% confidence level)

$\sqrt{tr(P_{f,r})}$	49.5 [km]
$\sqrt{tr(P_{f,v})}$	$1.43e-4$ [km/s]

Table 6: Final precision