STATISTICAL ORBIT DETERMINATION USING THE ENSEMBLE KALMAN FILTER

Eduard Gamper[(1)], Christopher Kebschull[(2)], and Enrico Stoll[(3)]

[(1)]*Institute of Space Systems, 38114 Braunschweig, Germany, Email: e.gamper@tu-braunschweig.de*
[(2)]*OKAPI:Orbits @ Institute of Space Systems, 38114 Braunschweig, Germany, Email: christopher.kebschull@okapiorbits.space*
[(3)]*Institute of Space Systems, 38114 Braunschweig, Germany, Email: e.stoll@tu-braunschweig.de*

ABSTRACT

A large number of resident space objects (RSO) is located in Earth's vicinity. Today, 19206 RSOs, thereof 2027 are active satellites, are trackable. Nevertheless, the number will increase by thousands of additional satellites as the so-called mega constellations are established. This will lead to an increased risk of collisions with other RSOs. The RSOs can be detected using telescope and radar sensors. Detections of RSOs always come with inaccuracies due to measurement noise. Extrapolation methods for orbit determination in turn suffer from model inaccuracies, which lead to a decay of the orbit information over time. Frequent updates of the orbital states are needed to compensate for the degradation. Methods within the field of statistical orbit determination (SOD) are used to decrease the error of the measurement and improve the latter.

One particular method is the usage of Kalman filters, where a dynamical model of the specific problem is compared against sensor measurements. In the following, a specific type of Kalman filter is investigated, the ensemble Kalman filter. The EnKF uses a set of randomly chosen states based on a probability density function (ensemble) to approximate the uncertainty of the state vector. The ensemble is propagated and updated using the measurement at the respective epoch. It is tested within a simulation environment and compared to the Unscented Kalman filter (UKF) to evaluate, whether it is possible to use the EnKF for orbit determination. Thereby, the number of ensembles and the number of measurements is varied.

Keywords: SSA; Propagator; Unscented Kalman Filter; Ensemble Kalman Filter; Uncertainty; Covariance Matrix.

1. INTRODUCTION

Due to the launch of mega constellations, the population in Earth's vicinity will further increase by more resident space objects (RSO) [29]. Currently, the number of trackable RSOs is $19\,206$, whereas 2027 are active satellites as of 28. Sep. 2018 [1]. The remaining RSOs consist of space debris as inoperative satellites, rocket upper stages, mission related objects, or other space debris bigger than $5\,\mathrm{cm}$ to $10\,\mathrm{cm}$ in diameter [21]. This threshold is defined by the capability of radars and telescopes to track RSOs. In fact, the detection threshold may be lower. A far higher number of smaller RSOs exist that were generated by collisions (e.g. Cosmos and Iridium in 2009 [27]), explosions, or slag and dust from solid rocket motors.

The orbital data of catalogued RSOs are currently made available to the public as two line elements (TLE) datasets (averaged Keplerian elements according to the SGP4 theory) by the Combined Space Operations Center (CSpOC). However, the precision of TLEs is low (about several hundred meters [21, 19]) due to the doubly averaged Keplerian elements. On the other hand, the CSpOC (formerly JSpOC) use precise orbit data that is not accessible to public, which can be used to provide collision warnings (Conjunction Data Messages) to satellite operators.

If precise orbit data is not available or the orbit determination (OD) suffers from a low data quality, OD algorithms have to be applied to improve orbital data by incorporation of further measurements. Propagation of orbital states are of further interest for long term predictions [28] and collision avoidance maneuvers [34]. One approach is the application of Kalman filters. The Kalman filters consist of two parts, the time update and the measurement update. Within the time update, the next state vector[1] is propagated using a dynamical model. Subsequently, a measurement at the specific epoch is used to improve the propagated state vector. The originally postulated *Kalman filter* (KF) [17] is not suitable to orbit determination problems because it is only applicable to linear cases. A non-linear extension of the KF is the *Extended Kalman Filter* (EKF), where the dynamical model is linearized. By the resulting derivation, or Jacobi matrix, the state vector and its uncertainty is propagated. Nevertheless, the calculation of the Jacobi matrix can be complicated due to the nature of the derivatives [16]. The problems of

[1]The state vector in orbit determination consists of the position and velocity information.

the EKF motivated the development of a new variant of the KF, the *Unscented Kalman Filter* (UKF) [16]. Compared to the EKF, the UKF does not use linearized time updates. Instead, a stochastic propagation of the uncertainty is performed. The state and uncertainty is propagated by splitting the state in the so called (deterministic) sigma points that are propagated and merged again.

The filter tested within this paper is the *Ensemble Kalman Filter* (EnKF). The EnKF was introduced to forecast model statistics in an ocean model [4] and was used in other fields like meteorology and climatology [24, 37, 22], oil reservoir models [9], and geo sciences [31]. An extensive overview of application areas and modifications of the EnKF is given in [5, 2, 33]. The EnKF was developed to conquer the problems of the KF and EKF with the propagation of the state. The KF is not applicable with non-linear dynamical models and the usage of the EKF can lead to instability due to the linearized propagation of the covariance matrix. Thus, Evensen [4] proposed a stochastic method using random sampling points (similar to the sigma points of the UKF) to approximate the propagation of the uncertainty.

The general possibility of applying the EnKF in OD is studied in this paper. A part of the tests is accomplished in comparison to the UKF. In section 2, the basics of OD, UKF, and EnKF are shown. Tests are performed and discussed in section 3, where different parameters are varied. Lastly, a conclusion is given in section 4.

2. FILTER

2.1. Basics

The input of the filter has to be the state vector and its covariance matrix because the Kalman filters are not able to calculate the covariance matrix out of given state vectors. The state vector describes the actual state of a RSO. In orbit determination problems, the state vector

$$\boldsymbol{X} \in \mathbb{R}^6,$$

consists of six elements containing position and velocity for each Cartesian coordinate direction. The covariance matrix

$$\boldsymbol{P} \in \mathbb{R}^{6\times 6},$$

describes the uncertainty of the state vector and is a symmetric, positive definite matrix. On its main diagonal, the variances (the squared standard deviations) for each position and velocity direction are assigned. Furthermore, the minor diagonals contain the correlations between the elements of the state vector [36]. Figuratively, the covariance matrix describes an error ellipsoid. In its center, the RSO is located. This error ellipsoid is depicted in Figure 1 using the UVW-reference system[2]. An error ellipsoid is available for the position and the velocity, respectively.

[2]The UVW-reference system describes a system which origin is the orbiting RSO. Its axis U is the radial vector that coincides with the vector pointing from Earths center to the RSOs center, the axis V (along-track) points tangentially into the moving direction, and W (cross-track) is the orthogonal axis of U and V [38].

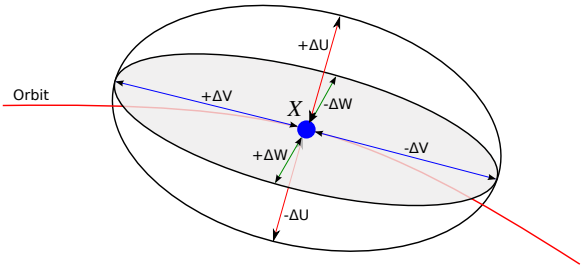

Figure 1. Depiction of uncertainties of the state vector $\boldsymbol{X}$ using the UVW reference system by an error ellipsoid [6].

The process described by the **time update** is used to propagate the state vector. A given state vector at epoch t_0 is propagated using the mathematical model up to epoch t_1. Therefore, (semi-) analytical or numerical propagators can be used. The advantage of analytical propagators is their computational speed but the drawback is the low accuracy. However, numerical propagators are usually more accurate but the calculation time increases [25]. Due to lack of knowledge of the exact physical effects, or simplifications, the propagators do not propagate the state vector exactly. That is why the missing information has to be incorporated additionally. This is achieved by using the **process noise** $\boldsymbol{Q} \in \mathbb{R}^{6\times 6}$. The process noise is described by a matrix which is an equivalent to the covariance matrix. Having Gaussian white noise, the process noise is added to the covariance matrix.

In the next step, the propagated state vector and covariance matrix are processed by the **measurement update**. Here, the observation vector at the corresponding epoch is used to update the propagated state vector and covariance matrix. This is done by the calculation of the **Kalman Gain** $\boldsymbol{K} \in \mathbb{R}^{6\times 6}$. The Kalman Gain is a factor (in the multivariate case it's a matrix) that weighs at what ratio the propagated state vector or the measurement will be used to achieve the new state. The calculation of the Kalman Gain requires the previously mentioned propagated covariance matrix, a measurement covariance matrix (the calculation is specific for the different Kalman filters), and the **measurement noise** $\boldsymbol{R} \in \mathbb{R}^{6\times 6}$. The measurement noise describes the uncertainty (position and velocity) of the measurement. Nevertheless, the measurement noise is not constant and varies for every measurement. Propagated state vector and its covariance matrix are updated using the Kalman Gain and the measurement at the respective epoch. The measurement updates (as the time update) contain specific equations that are different for UKF and EnKF. Thus, the detailed formulation is shown in the following sections. The KF and the EKF will not be described in this paper. These details can be found in the literature [17, 38].

2.2. Unscented Kalman Filter

The principle of the UKF is described in Figure 2. The state vector is assumed to be a random variable featuring an expected value (first moment). The uncertainty of this expected value is stated by the covariance matrix (second moment) in the multivariate case.

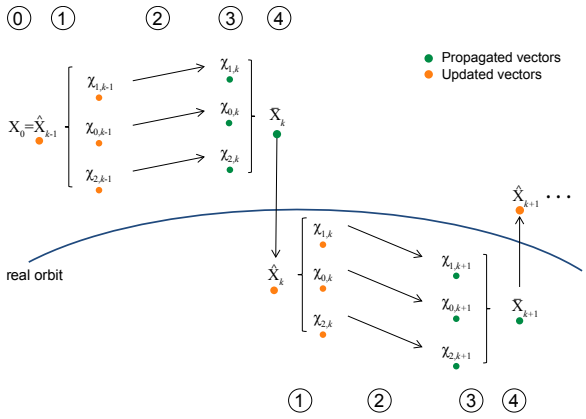

Figure 2. Procedure of the UKF [6].

The steps within the time update are shown by the points 1-3. Firstly, the UKF is initialized by the state vector $\hat{\boldsymbol{X}}_0$ that is equal to the filter input $\hat{\boldsymbol{X}}_{k-1}$ at time step $k-1$. The hat $\hat{(\,)}$ depicts a state vector after the measurement and the bar $\bar{(\,)}$ depicts a state vector after the time update. In step 1, further state vectors are generated using the state vector $\hat{\boldsymbol{X}}_{k-1}$, which is assumed to be a randomly distributed variable. This process is called unscented transformation (UT) that generates 13 so called sigma points. These sigma points map the uncertainty of the random variable and form an ellipsoid. In Figure 2, the sigma points are exemplarily depicted by two sigma points $\boldsymbol{\chi}_{1,k-1}$ and $\boldsymbol{\chi}_{2,k-1}$. The central sigma point $\boldsymbol{\chi}_{0,k-1}$ is equal to the state vector $\hat{\boldsymbol{X}}_{k-1}$. The usage of 12 or 13 sigma points that are with or without the central state vector, depends on the literature. In step 2, the sigma points are propagated by the dynamical model. Subsequently, the sigma points are merged using weighting factors to get the propagated state vector $\bar{\boldsymbol{X}}_k$ (step 3) [16, 11]. Lastly, the measurement update is performed where the measurement $\boldsymbol{y}_k$ is incorporated calculating the updated state vector $\hat{\boldsymbol{X}}_k$. This process (steps 1-4) is repeated for every available measurement as shown in the block diagram in Figure 3. The updated state vectors are moving around the real orbit because, figuratively, they are following the real orbit. Due to the uncertainties, the real orbit cannot ideally be calculated. In the following sections, the filtering process is described in detail. The block diagram in Figure 3 shows the input and output of every process.

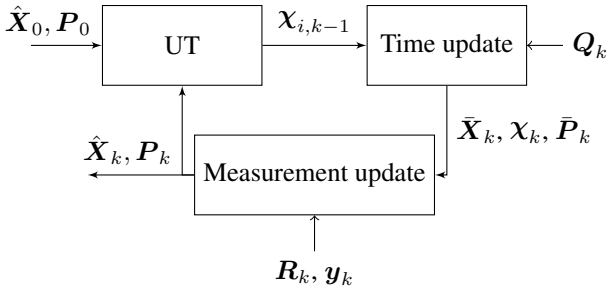

Figure 3. Block diagram of the UKF [6].

2.2.1. Unscented Transformation

The UT splits the state vector into several sigma points. In Equation 1, the calculation of 13 sigma points[3] is shown [16, 11, 30].

$$\left.\begin{aligned} \boldsymbol{\chi}_{0,k} &= \bar{\boldsymbol{X}}_{k-1} \\ \boldsymbol{\chi}_{i,k-1} &= \bar{\boldsymbol{X}}_{k-1} + \sqrt{(n+\lambda)\,\boldsymbol{P}_{k-1}|_i} \\ \boldsymbol{\chi}_{n+i,k-1} &= \bar{\boldsymbol{X}}_{k-1} - \sqrt{(n+\lambda)\,\boldsymbol{P}_{k-1}|_i} \end{aligned}\right\} i = 1, \ldots, n \tag{1}$$

Thereby, 12 of 13 sigma points are calculated using each column of the covariance matrix $\boldsymbol{P}$ at step k indicated by i, whereas the dimension if the state vector is

$$n = 6.$$

A scaling parameter λ is additionally used to vary the distance of the outer sigma points to the central sigma point $\boldsymbol{\chi}_{0,k}$. The square root of $\boldsymbol{P}$ is defined as

$$\sqrt{\boldsymbol{P}} = \boldsymbol{S}, \tag{2}$$

so that

$$\boldsymbol{P} = \boldsymbol{S}\boldsymbol{S}^T. \tag{3}$$

$\boldsymbol{S}$ defines a lower triangular matrix and its transpose an upper triangular matrix. $\boldsymbol{S}$ is usually calculated by the Cholesky decomposition [8]. Examplary, the second expression in Equation 1 is calculated as

$$\boldsymbol{\chi}_{i,k-1} = \bar{\boldsymbol{X}}_{k-1} + \sqrt{n+\lambda}\boldsymbol{S}_{k-1}|_i.$$

The calculation of the scaling parameter λ is defined in Equation 4 where α adjusts the distance between the central and the other sigma points that is set to small, non-negative values [11].

$$\lambda = \alpha^2\,(n+\kappa) - n \quad \text{where} \quad \begin{aligned} 10^{-4} \leq \alpha \leq 1 \\ \kappa = 3 - n. \end{aligned} \tag{4}$$

κ is also a scaling parameter that is dependent on the dimension of the sate vector [11, 16]. The matrix $\boldsymbol{\chi} \in \mathbb{R}^{6\times 13}$ contains the resulting sigma points.

[3] For the case of orbit determination, where $n = 6$ is valid for a three dimensional state space

2.2.2. Time Update

The previously calculated sigma points are propagated using the non-linear dynamical model $\boldsymbol{F}$ (Equation 5).

$$\boldsymbol{\chi}_k^* = \boldsymbol{F}\left(\boldsymbol{\chi}_{k-1}, \boldsymbol{u}_{k-1}\right) \tag{5}$$

The model input $\boldsymbol{u}$ in the general formulation in Equation 5, which describe accelerations that are not induced by the natural environment (e.g. due to usage of rocket engines), is omitted in the following considerations. In the following step, the propagated sigma points are combined to achieve the propagated state vector $\bar{\boldsymbol{X}}_k$ (Equation 6).

$$\bar{\boldsymbol{X}}_k = \sum_{i=0}^{2n} W_i^{(m)} \boldsymbol{\chi}_{i,k}^* \tag{6}$$

The sum is weighted using the weighting factors stated by equations 7 and 8. The index (m) illustrates the affiliation to the state vector.

$$W_0^{(m)} = \frac{\lambda}{n+\lambda} \tag{7}$$

$$W_i^{(m)} = \frac{1}{2\left(n+\lambda\right)} \quad \text{for } i = 1, \ldots, 2n \tag{8}$$

Having calculated the propagated state vector, its propagated covariance matrix is

$$\bar{P}_k = \sum_{i=0}^{2n} W_i^{(c)} \left(\chi_{i,k}^* - \bar{X}_k\right)\left(\chi_{i,k}^* - \bar{X}_k\right)^T + \boldsymbol{Q}_k. \tag{9}$$

As mentioned before, the covariance matrix is not calculated using the state transition matrix used by the KF and EKF [38]. Instead, the UKF deviates the covariance matrix by calculating the residues of sigma points and propagated state vector. This sum is also weighted using equations 10 and 11. The index (c) illustrates the affiliation to the covariance matrix.

$$W_0^{(c)} = \frac{\lambda}{n+\lambda} + 1 - \alpha^2 + \beta \tag{10}$$

$$W_i^{(c)} = \frac{1}{2\left(n+\lambda\right)} \qquad \text{for } i = 1, \ldots, 2n \tag{11}$$

Different to Equation 7, [11] uses further tuning parameters in Equation 10 to weight the covariance matrix. If a Gaussian distribution is assumed, the parameter β is set to

$$\beta = 2$$

as applied by [11]. Additionally, the process noise matrix $\boldsymbol{Q}_k$ is used in Equation 9, which is calculated for the respective time step. This addition is allowed if the process noise is a Gaussian white noise [16, 11], which is assumed in the following. A further calculation method if white noise cannot be assumed, can be found in [11]. Process noise was not incorporated in the actual propagated state vector. Thus, [16, 11] propose to amplify the sigma points to consider the process noise (Equation 12).

$$\boldsymbol{\chi}_k = \left[\bar{\boldsymbol{X}}_k \;\; \bar{\boldsymbol{X}}_k + \sqrt{\left(n+\lambda\right)\bar{\boldsymbol{P}}_k} \;\; \bar{\boldsymbol{X}}_k - \sqrt{\left(n+\lambda\right)\bar{\boldsymbol{P}}_k}\right] \tag{12}$$

The UT is applied by using the propagated covariance matrix $\bar{\boldsymbol{P}}_k$.

In the measurement update described in the next section, one part of the update is the comparison of the true masurement $\boldsymbol{y}_k$ with the propagated state vector $\bar{\boldsymbol{X}}_k$. Therefore, $\bar{\boldsymbol{X}}_k$ has to be transformed into the same coordinate system as the measurement [16]. This non-linear transformation is performed by the function $\boldsymbol{h}$ for every amplified sigma point $\boldsymbol{\chi}_{i,k}$ (Equation 13):

$$\boldsymbol{Y}_{i,k} = \boldsymbol{h}\left(\boldsymbol{\chi}_{i,k}\right) \tag{13}$$

In orbit determination, the function $\boldsymbol{h}$ is used to transform the Cartesian, inertial geocentric coordinate system ECI (state vector) into the topocentric coordinate system. Here, the true measurement consists of the coordinates range, azimuth, elevation, and their derivatives. As for $\bar{\boldsymbol{X}}_k$, the transformed state vector $\bar{\boldsymbol{y}}_k$ is calculated by Equation 14 using the weighting factors obtained from equations 7 and 8.

$$\bar{\boldsymbol{y}}_k = \sum_{i=0}^{2n} W_i^{(m)} \boldsymbol{Y}_{i,k} \tag{14}$$

2.2.3. Measurement Update

The measurement update serves to refine the propagated state vector by the data of the observed RSO. As already mentioned, the Kalman Gain $\boldsymbol{K}$ has to be calculated for this purpose. Therefore, the covariance matrix of the transformed state vector $\boldsymbol{P}_{yy,k}$ (Equation 15), and the cross covariance matrix $\boldsymbol{P}_{xy,k}$ (Equation 16) have to be calculated using the sigma points and its transformed derivate. The indices (yy) and (xy) clarify the considered residues of state vector or transformed state vector.

$$\boldsymbol{P}_{yy,k} = \sum_{i=0}^{2n} W_i^{(c)} \left(\boldsymbol{Y}_{i,k} - \bar{\boldsymbol{y}}_k\right)\left(\boldsymbol{Y}_{i,k} - \bar{\boldsymbol{y}}_k\right)^T + \boldsymbol{R}_k \tag{15}$$

$$\boldsymbol{P}_{xy,k} = \sum_{i=0}^{2n} W_i^{(c)} \left(\boldsymbol{\chi}_{i,k} - \bar{\boldsymbol{X}}_k\right)\left(\boldsymbol{Y}_{i,k} - \bar{\boldsymbol{y}}_k\right)^T \tag{16}$$

For every filtering iteration at step k, the measurement noise $\boldsymbol{R}_k$ have to be calculated (see section 3.2.5) because the measurement noise depends on the actual properties of the measuring station. The Kalman Gain results as a multiplication of the matrix $\boldsymbol{P}_{xy,k}$ and the inverse of matrix $\boldsymbol{P}_{yy,k}$ (Equation 17).

$$\boldsymbol{K}_k = \boldsymbol{P}_{xy,k}\boldsymbol{P}_{yy,k}^{-1} \tag{17}$$

Finally, the propagated state vector $\bar{\boldsymbol{X}}_k$ is updated by Equation 18 to $\hat{\boldsymbol{X}}_k$.

$$\hat{\boldsymbol{X}}_k = \bar{\boldsymbol{X}}_k + \boldsymbol{K}_k\left(\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k\right) \tag{18}$$

The Kalman Gain is used to update the covariance matrix by Equation 19.

$$\hat{\boldsymbol{P}}_k = \bar{\boldsymbol{P}}_k - \boldsymbol{K}_k\boldsymbol{P}_{y_k y_k}\boldsymbol{K}_k^T \tag{19}$$

If further measurements are available, the filtering process is repeated starting with the UT.

2.3. Ensemble Kalman Filter

The procedure of the EnKF is shown in Figure 4.

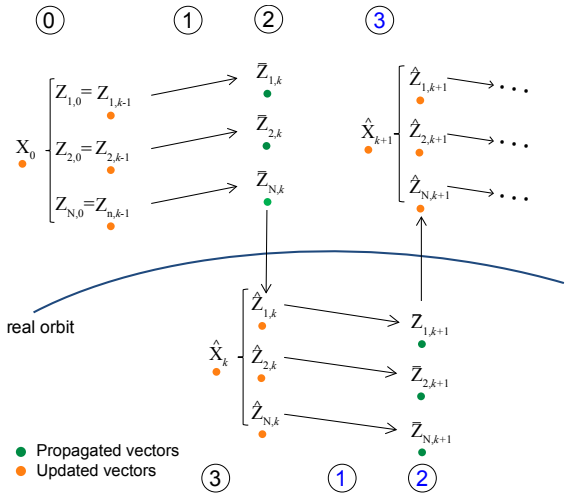

Figure 4. Procedure of the EnKF.

The EnKF is initialized by the state vector $\boldsymbol{X}_0$ (random variable) from which an arbitrary number of sampling points, the so called ensembles [5, 18], are generated (step 0). Within the time update in step 1, the ensembles are propagated to the epoch of the measurements. The update of the ensemble is accomplished within the measurement update in step 2. To achieve the updated state vector $\hat{\boldsymbol{X}}_k$ (and also the updated covariance matrix $\hat{\boldsymbol{P}}_k$), the ensemble is merged (step 3). The ensemble $\hat{\boldsymbol{Z}}_k$ is the input for the next iteration step in contrary to the UKF, where the updated state vector and covariance matrix are used. As shown in the block diagram in Figure 5, the ensemble is not regenerated.

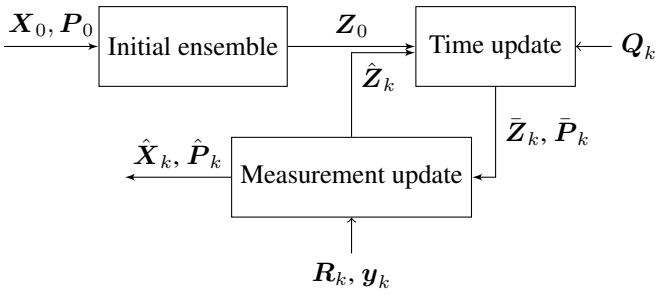

Figure 5. Block diagram of the EnKF.

2.3.1. Initial Ensemble Matrix

The filtering process begins with the generation of sampled state vectors based on the initial state vector. The set of sampling points (ensembles) is stored by the matrix $\boldsymbol{Z}_k \in \mathbb{R}^{6\times N}$, whereas N states the number of sampling points. These ensembles represent the uncertainty of the initial state $\boldsymbol{X}_0$ [5]. Therefore, the initial ensemble matrix is generated using the initial covariance matrix $\boldsymbol{P}_0$. The generation of the initial ensemble $\boldsymbol{Z}_0$ is shown in Equation 20.

$$\begin{aligned} \boldsymbol{Z}_{i,0} &= \boldsymbol{\tau}_i \quad \text{with} \quad \boldsymbol{\tau}_i \sim \mathcal{N}(\boldsymbol{x}_0, \boldsymbol{P}_0) \\ \boldsymbol{Z}_0 &= [\boldsymbol{Z}_{1,0}, \boldsymbol{Z}_{2,0}, \ldots, \boldsymbol{Z}_{N,0}] \end{aligned} \tag{20}$$

$\boldsymbol{\tau}_i$ is a normally distributed random variable, whereas the initial state vector is its mean and the initial covariance matrix its uncertainty.

2.3.2. Time Update

Within the time update, each ensemble member $\hat{\boldsymbol{Z}}_{i,k-1}$ is propagated by the dynamic model $\boldsymbol{F}$ to the next time step (Equation 21). $\hat{\boldsymbol{Z}}_{i,k-1} = \boldsymbol{Z}_{i,0}$ is applied for the first pass of the filter.

$$\begin{aligned} \bar{\boldsymbol{Z}}_{i,k} &= \boldsymbol{F}\left(\hat{\boldsymbol{Z}}_{i,k-1}\right) + \boldsymbol{w}_{i,k} \quad \text{with} \quad \boldsymbol{w}_{i,k} \sim \mathcal{N}(0, \boldsymbol{Q}_k) \\ \bar{\boldsymbol{Z}}_k &= \left[\bar{\boldsymbol{Z}}_{1,k}, \bar{\boldsymbol{Z}}_{2,k}, \ldots, \bar{\boldsymbol{Z}}_{N,k}\right] \end{aligned} \tag{21}$$

Additionally, a random error $\boldsymbol{w}_{i,k}$ is added [18]. This error represents the uncertainty of the model itself. The covariance matrix of the model error is the process noise matrix $\boldsymbol{Q}_k$. Following, the propagated mean $\bar{\boldsymbol{Z}}_{m,k}$ of the ensemble matrix is calculated using Equation 22.

$$\bar{\boldsymbol{Z}}_{m,k} = \frac{1}{N}\sum_{i=1}^{N} \bar{\boldsymbol{Z}}_{i,k} \tag{22}$$

To avoid the divergence of the filter due to an underestimated state covariance matrix, a scaling parameter γ was applied to "inflate" the covariance matrix [10]. This inflation is achieved by an artificial increase of the ensembles residues as shown in Equation 23.

$$\bar{\boldsymbol{Z}}_{i,k} \leftarrow \gamma\left(\bar{\boldsymbol{Z}}_{i,k} - \bar{\boldsymbol{Z}}_{m,k}\right) + \bar{\boldsymbol{Z}}_{m,k} \tag{23}$$

The propagated covariance matrix is obtained by Equation 24 [35].

$$\bar{\boldsymbol{P}}_k = \frac{1}{N-1}\sum_{i=1}^{N}\left(\bar{\boldsymbol{Z}}_{i,k} - \bar{\boldsymbol{Z}}_{m,k}\right)\left(\bar{\boldsymbol{Z}}_{i,k} - \bar{\boldsymbol{Z}}_{m,k}\right)^T \tag{24}$$

The weighting in Equation 22 and Equation 24 is the same for every state vector because the ensemble members are threated equally. The process noise matrix $\boldsymbol{Q}_k$ is not added as in Equation 9 anymore because the model error was already incorporated randomly in the ensembles.

2.3.3. Measurement Update

The measurement update of the original EnKF ([4, 5]) has a drawback. Although the time update contains no linearizations, the measurement update still contains the measurement operator $\boldsymbol{H}$ that transforms the state vector. Nevertheless, $\boldsymbol{H}$ is linear, or linearized. [15] proposed an approach (and revisited among others of [35]) according to [16] that got rid of this measurement operator. This approach is similar to the procedure of the UKF.

Within this paper, a scheme is used that is proposed by Tang [35]. At first, ensembles as well as its mean have to be transformed with the measurement function h (Equation 25). The upper index (e) states that the matrix or vector belongs to an ensemble.

$$\begin{aligned} \boldsymbol{Y}_{i,k}^{(e)} &= h\left(\bar{\boldsymbol{Z}}_{i,k}\right) \\ \boldsymbol{Y}_{k}^{(e)} &= \left[\boldsymbol{Y}_{1,k}^{(e)}, \boldsymbol{Y}_{2,k}^{(e)}, \ldots, \boldsymbol{Y}_{N,k}^{(e)}\right] \\ \boldsymbol{Y}_{m,k} &= h\left(\bar{\boldsymbol{Z}}_{m,k}\right) \end{aligned} \tag{25}$$

The transformed ensembles are stated column-wise in the matrix $Y_k^{(e)} \in \mathbb{R}^{6\times N}$. The measurement covariance matrix $\boldsymbol{P}_{yy,k}^{(e)}$ (Equation 26) and the cross covariance matrix $\boldsymbol{P}_{xy,k}^{(e)}$ (Equation 27) have to be calculated including the transformed ensemble. Two possibilities to calculate these two matrices are proposed in [35]. One method consist of the usage of the transformed measurements and the transformed mean to calculate the residues (as done by the UKF). The second method uses randomly perturbed (real) measurements instead of the transformed mean as also shown by [5]. By this method, the measurement noise is directly incorporated into the measurements. Nevertheless, the first method mentioned led to a stable filtering process. The calculation of $\boldsymbol{P}_{yy,k}^{(e)}$ and $\boldsymbol{P}_{xy,k}^{(e)}$ is shown in Equation 26 and Equation 27, respectively.

$$\bar{\boldsymbol{P}}_{yy,k}^{(e)} = \frac{1}{N-1}\sum_{i=1}^{N}\left[\boldsymbol{Y}_{i,k}^{(e)} - \bar{\boldsymbol{y}}_{k}^{(e)}\right]\left[\boldsymbol{Y}_{i,k}^{(e)} - \bar{\boldsymbol{y}}_{k}^{(e)}\right]^{T} + \boldsymbol{R}_k \tag{26}$$

$$\bar{\boldsymbol{P}}_{xy,k}^{(e)} = \frac{1}{N-1}\sum_{i=1}^{N}\left[\bar{\boldsymbol{Z}}_{i,k} - \bar{\boldsymbol{Z}}_{m,k}\right]\left[\boldsymbol{Y}_{i,k}^{(e)} - \bar{\boldsymbol{y}}_{k}^{(e)}\right]^{T} \tag{27}$$

To incorporate the uncertainty of the measurements, the measurement noise R_k is added to the measurement covariance matrix.

The Kalman Gain $\boldsymbol{K}_k^{(e)}$ is calculated (Equation 28) by the quotient of equations 27 and 26.

$$\boldsymbol{K}_{k}^{(e)} = \bar{\boldsymbol{P}}_{xy,k}^{(e)}\left(\bar{\boldsymbol{P}}_{yy,k}^{(e)}\right)^{-1} \in \mathbb{R}^{6\times 6} \tag{28}$$

Finally, the propagated ensemble $\bar{\boldsymbol{Z}}_k$ is updated by Equation 29 incorporating the Kalman Gain, the perturbed measurements, and the transformed ensemble matrix.

$$\hat{\boldsymbol{Z}}_k = \bar{\boldsymbol{Z}}_k + \boldsymbol{K}_k^{(e)}\left(\boldsymbol{y}_k^{(e)} - \boldsymbol{Y}_k^{(e)}\right) \tag{29}$$

A possibilitiy to update the covariance matrix that was proposed by [35] is

$$\hat{\boldsymbol{P}}_k = \bar{\boldsymbol{P}}_k - \boldsymbol{K}_k^{(e)}\bar{\boldsymbol{P}}_{xy,k}^{(e)}. \tag{30}$$

Unfortunately, this method leads to a erroneous calculation of $\hat{\boldsymbol{P}}_k$ because certain variances could be negative. Another consideration was to use Equation 19 that was applied in the measurement update of the UKF. Thus, the right-hand side term was substituted by the term $\boldsymbol{K}_k\boldsymbol{P}_{y_k y_k}\boldsymbol{K}_k^T$ yielding to Equation 31. The following equation was applied within the test to update the covariance matrix.

$$\hat{\boldsymbol{P}}_k = \bar{\boldsymbol{P}}_k - \boldsymbol{K}_k^{(e)}\boldsymbol{P}_{y_k y_k}^{(e)}\boldsymbol{K}_k^{(e)\,T} \tag{31}$$

The updated covariance matrix $\hat{\boldsymbol{P}}_k$ is not used in the next filtering step as an input. Instead, the updated ensemble is reutilised, and will be propagated according to the described procedure. The updated covariance matrix is calculated from scratch using only the deviations induced through the propagation and process noise.

3. TEST OF FILTERS

In this section, the testing scenario is firstly outlined to which the filters are applied. Then, the settings of the filters are shown that contain the initialization of the filters, and the selection of the process and measurement noise. Lastly, the results of the variation of the parameters are outlined.

3.1. Testing scenario

The filters are tested using a software that provides a simulation environment for Space Surveillance and Tracking (SST) purposes, the Radar System Simulator (RSS) [20]. It is being developed with the goal to be able to study and evaluate different SST setups, from the sensor to the catalogue generation. The simulation software is made of five tools [13]:

- MWG: Measurement generation
- SMART: Orbit determination algorithms
- PROCOR: Processes coordination
- CAT: Catalogue statistics
- CAMP: Catalog Maintenance and Pass prediction tool

The measurement generator simulates different kinds of radar systems. It uses a radar performance model supplied by the Fraunhofer FHR in order to be able to create detections with three different operational modes of radars in the SST context:

1. Mechanical Tracking,
2. Electronic Tracking, and
3. Surveillance.

The model regards the location of the simulated sensor as well as several performance influencing parameters, such as the transmit energy, wavelength, gain, pulse repetition frequency, pulse duration and the $3\,\mathrm{db}$ opening angle of the beam, loss rate, false alarm probability, assumed measurement noise, and pulse integration settings. Within this paper the first mode (mechanical tracking) is used without any integration technique. The MWG is used to simulate the detections with applied uncertainties to the state based on the definition of the range and angular resolution capability of the simulated tracking radar. The measurement generator simulates the ground based sensors and space based objects on a millisecond basis as needed for the pulses of the radar. The numerical propagator NEPTUNE is used to extrapolate the state vector of the RSO over a given timeframe and create chebyshev polynomials [3]. Based on chebyshev polynomials the states are interpolated for each pulse. The output from the MWG is information in the form of a noisy observation state: range (R), range-rate (RR), azimuth (Az) and elevation (El), runtime corrected time of the detection and the signal-to-noise-ratio (SNR). The values as well as information about the radar sensor are transferred to a database, where multiple measurements of the same RSO during a single pass over the sensor are grouped into tracklets.

The second tool SMART (Sophisticated Module for the Analysis of Radar Tracklets) is used to perform orbit determination on the tracklets in the database. SMART retrieves the corresponding measurements from the database and processes them. Different techniques are available to process measurements. For the case of no a priori knowledge of the RSO (no prior state vector in the database) two initial orbit determination algorithms have been implemented:

- Gibbs,
- Herrick-Gibbs,
- Preliminary orbit determination method documented in the Goddard Trajectory Determination System (GTDS).

The methods are based on [38] and [23] and have been implemented and tested in [32] and [14], respectively. When SMART processes a known RSO, usually a state vector is already available in the database which can be used in the statistical orbit determination. Four methods have been implemented [6, 12, 20]:

- Weighted Least Squares (WLS),
- Extended Kalman Filter (EKF),
- Unscented Kalman Filter (UKF) and
- Ensemble Kalman Filter (EnKF).

Given many incoming tracklets from a sensor PROCOR (Process Coordinator) is able to utilize multiple SMART instances in order to process the orbital information in parallel. Furthermore through PROCOR, measurement filters can be enabled and settings can be passed to the SMART instances in order to optimally process the sensor measurements. In a simulation environment these settings can also be determined using the PROCOR tool. The last tool in the chain CAT (Catalogue Analysis Tool) can compare the achieved accuracy for individual RSOs, processed with various settings in the database against the truth that is known from the measurement generator. Thus knowledge can be gained which sensor and which settings are preferable in order to optimize the data in the catalogue.

The test of the filters requires the knowledge of the uncertainty of the available measurements. Otherwise, it is hardly possible to decide, whether the filter provides the correct solution. RSS is able to procure these data. So it is possible to compare the filtered data with the exact orbit.

The tested orbit is a nearly circular, sun synchronous low Earth Orbit (LEO). The orbit parameters are:

$$\begin{aligned} a &= 7150\,\mathrm{km} \\ e &= 0.0012 \\ i &= 98^\circ \end{aligned}$$

The simulated radar is the Tracking and Imaging Radar (TIRA) in Germany. For the tests, 100 tracklets were used, which cover an overall period of approximately two weeks. Before the filtering process is started, the measurements from the first tracklet are used to calculate the initial state vector (within IOD) and to calculate the covariance matrix (by WLS).

The filter is tested varying three types of data. Firstly, the EnKF is tested while varying scaling parameter γ to inflate the covariance matrix. Then, the number of ensemble members (section 3.3.2) is varied. This test should show how many ensembles are needed to ensure a stability of the filter and converging results. In the third test, the number of measurements is varied (section 3.3.3). The latter test comprises the comparison of EnKF with the UKF.

3.2. Filter settings

3.2.1. Initial state vector and covariance matrix

The initial state vector $\boldsymbol{X}_0$ is provided in RSS by the Gibbs or Herrick-Gibbs algorithm within the initial orbit determination (IOD). These IOD algorithms require the usage of three measurements, where the position vectors are used to obtain the full state vector, because the telescope or radar data usually does not provide the rates of the azimuth and elevation. Nevertheless, the IOD does not contain the generation of the initial covariance matrix $\boldsymbol{P}_0$. To obtain the covariance matrix, a least squares algorithm is applied on the next available measurements within the first tracklet. This method minimizes the error among the solution and the measurements. Using the minimized errors, the covariance matrix is calculated. The determination of the initial state vector and initial covariance matrix are already implemented parts of RSS.

3.2.2. Generation of initial Ensemble

The initial ensemble is generated using the initial covariance matrix $\boldsymbol{P}_0$ for the sampling of the ensemble members $\boldsymbol{Z}i_{,0}$. The sampling points are calculated [7] by

$$\boldsymbol{Z}_{i,_0} = \boldsymbol{S}\boldsymbol{n}_i + x_0 \quad \text{with} \quad \boldsymbol{n}_i \sim \mathcal{N}(0,1)\,. \tag{32}$$

$\boldsymbol{n}_i \in \mathbb{R}^6$ denotes a vector that contains randomly chosen values with mean 0 and standard deviation 1. The matrix $\boldsymbol{S}$ is calculated using $\boldsymbol{P}_0$ and Equation 3. The number of ensembles is chosen variably depending on the testing scenarios that are described in the following.

3.2.3. Model for propagation

The model $\boldsymbol{F}$ is used to propagate the state vector. Within the scope of this work, the numerical propagator NPI Ephemeris Propagation Tool with Uncertainty Extrapolation (NEPTUNE) is applied [3]. This propagator is able to integrate the acceleration due to different perturbations (gravitational and atmospheric perturbations, solar radiation pressure, third body perturbations, albedo as well as solid and ocean tides). NEPTUNE has a high accuracy but, similar to other numerical propagators, NEPTUNE is computationally expensive. Due to this drawback, the perturbation sources, except the geopotential, are deactivated. Regarding the ratios of the perturbations at the orbit altitude defined for the test case, the geopotential has the most influence on the calculation for different sources [26]. According to that, the geopotential has an impact that is several orders of magnitude higher. The degree of the geopotential is set to 6. So J_6 perturbations are considered.

3.2.4. Process noise

The modeling of the process noise $\boldsymbol{Q}_k$ is challenging because different sources of perturbations have to be considered. These sources are for example the geopotential, atmospheric density/drag, solar radiation pressure, third body perturbations, and tides. The modeling of these sources was not performed within this paper. Instead, a constant process noise was assumed to be

$$\boldsymbol{Q}_k = \begin{bmatrix} \sigma_x^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_y^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_z^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{dx}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{dy}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{dz}^2 \end{bmatrix}$$

with

$$\sigma_x^2 = \sigma_y^2 = \sigma_z^2 = 5 \cdot 10^{-4}\,\mathrm{km}^2$$
$$\sigma_{dx}^2 = \sigma_{dy}^2 = \sigma_{dz}^2 = 1 \cdot 10^{-11}\,\mathrm{km/s}^2.$$

3.2.5. Measurement noise

The calculation of measurement noise $\boldsymbol{R}_k$ is done by SMART based on the signal-to-noise ratio SNR of the considered radar. A root-means-square error is assumed

$$\sigma(M, SNR) = \frac{M}{\sqrt{2 \cdot SNR}}, \tag{33}$$

where the radars basic resolution M and the SNR per detection is considered to get an estimate of the detection uncertainty. The resulting standard deviation is depicted by σ. For high SNR values a level-off is assumed, where the resolution is not further improved and remains constant even though the SNR is rising. The threshold is SNR_{Ref} and leads to a constant σ_{ref}:

$$\sigma(M, SNR) = \begin{cases} \dfrac{M}{\sqrt{2 \cdot SNR}} & ,\text{if } SNR < SNR_{ref} \\ \sigma_{ref} & ,\text{if } SNR \geq SNR_{ref} \end{cases}. \tag{34}$$

The reference values for M, σ_{ref} and SNR_{ref} are listed in Table 1. A constant value for the SNR was not applied as the SNR varies in dependence of the objects position within the radar cone. Figure 6 shows the SNR for the first used tracklet. The closer the object approaches the cones center, the higher the SNR becomes.

Table 1. Reference Values for M, σ_{ref} and SNR_{ref}.

	Range	Az. & El.	Range Rate
M	758 m	0.0032°	632 m/s
σ_{ref}	12 m	0.2545°	10 m/s
SNR_{ref}	33 dB	35 dB	33 dB

The deviations of the rates of azimuth and elevation cannot be stated because they are not measured by the considered radar. Hence, the value for the uncertainty azimuth and elevation is assumed to be very high ($\sigma^2 =$

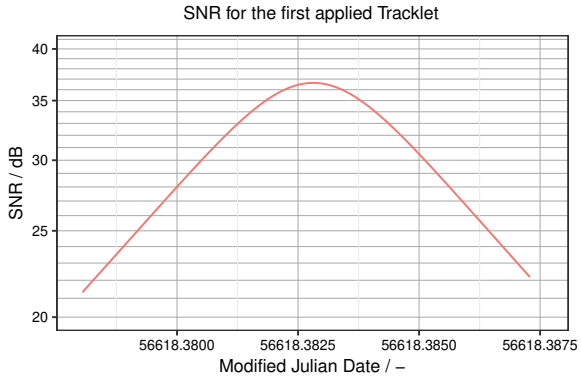

Figure 6. Signal to Noise ratio for the first available tracklet.

10^{100}). The consequence is that these values, calculated by the dynamical model, are accepted instead of using the observed rates. As a result the measurement noise is represented using the matrix:

$$\boldsymbol{R}_k = \begin{bmatrix} \sigma_R^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{Az}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{El}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{RR}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^{100} & 0 \\ 0 & 0 & 0 & 0 & 0 & 10^{100} \end{bmatrix}. \tag{35}$$

Alternatively, a measurement vector with the lower dimension measurement $\boldsymbol{y} \in \mathbb{R}^4$ could be used, instead of setting the azimuth and elevation rate to zero. As a result, the measurement noise matrix and covariance matrix would have the dimension $\mathbb{R}^{4\times4}$, and the Kalman Gain the dimension $\mathbb{R}^{6\times4}$. In the used filter algorithms, this simplification was not applied to keep the possibility using larger measurement vectors in further simulations.

3.3. Variation of parameters

The variation of different parameters should show, whether the EnKF is able to improve noisy measurements in comparison to the UKF. Moreover, it will be investigated if the filters are working stably and if they diverge using different data. It has to be taken into consideration that the total number of measurements per tracklet differs as the crossing trajectories among object and the radars cone are not the same. The number of measurements coincides with the SNR and radial distance, and thus, the frequency by which the measurements are obtained. For the tests, a specific number of measurements were selected from the complete measurement range of each tracklet. The currently used algorithm does not choose the measurements randomly from the tracklet. Instead, the selected measurements are evenly distributed distributed along the respective detection range. Thus, distance between each measurement of the chosen amount is the same but differs for different tracklets. It has to be regarded that the distance of last measurement of a tracklet and the first measurement of the subsequent tracklet is approximately one orbital revolution. The iteration process is not re-initiated for every tracklet in the range of 100 tracklets.

The position and velocity errors are described as the absolute value of the errors in the directions U, V, and W. The errors are the difference between the updated (transformed) state vectors and the real measurement. Thereby, the shown errors are describes using different quantiles (90 %, 95 %, and 99 %). Hence, the quantiles state an upper threshold below which the resulting errors, calculated at every filtering iteration, are regarded in the evaluation. By this step, large errors that result at the beginning of an iteration loop, and thus the converging process, are omitted.

3.3.1. Varying the scaling parameter γ

Within the first test, the scaling parameter γ was varied. This parameter inflates (or deflates) the covariance matrix of the propagated state vector modifying the residues among ensembles and its mean. Here, the residues are multiplied with the scaling parameter within a range of $0.8 \ldots 1.2$ ($80\,\% \ldots 120\,\%$). In contrast to [10], where the covariance matrix is inflated by 1 %, this test varies the scaling parameter within a higher range. The reason is that the propagated ensemble is sophisticated by the process noise, and thus the residues depend also on the latter. Consequently, choosing a smaller process noise could yield to the same result as if choosing a lower scaling parameter. The following settings were chosen for this test:

- Scaling parameter γ: $0.8 \ldots 1.2$
- No. ensembles N: 50
- No. measurements: 200

Figure 7 and Figure 8 show that the position and velocity error increases for a very low and a very high γ. Looking at the quantile of 99 %, the minimum position error (871 m) is obtained using $\gamma = 0.95$. Nevertheless, for the 90 % quantile, the minimum position error moves to smaller scaling parameters (402 m at $\gamma = 0.9$). In contrast, the position error for the 90 % quantile is not at a minimum.

It can carefully be concluded that the convergence rate is lower for smaller scaling parameters, and thus the amplitude of errors is larger. But the amplitude of errors is lower after the filter has converged. If the scaling factor is further decreased below $\gamma = 0.8$, the filter begins to diverge. As the ensemble is not recalculated, the deflating of covariance matrix (by reducing the residues of the ensembles) is having an effect on every subsequent iteration. Apparently, the covariance matrix is becoming too

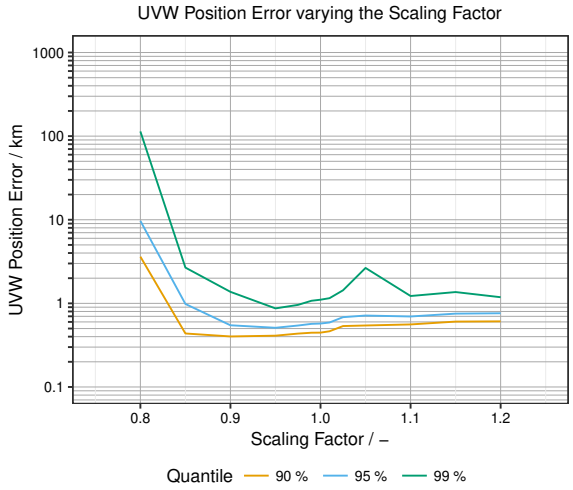

Figure 7. Position error in UVW varying the scaling parameter γ.

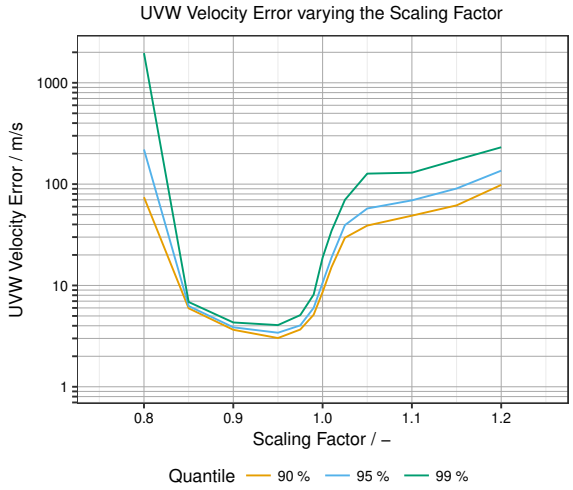

Figure 8. Velocity error in UVW varying the scaling parameter γ.

low by which the state vector is over- and measurements are underestimated.

In the case of an higher scaling parameter, the filter is diverging above an value of $\gamma = 1.2$. Despaired to the higher position error at $\gamma = 0.8$, the position error at is by several orders of a magnitude lower. Viewing at Figure 8, the velocity error is showing a large increase of the velocity error at low but at high scaling factors as well having an error of $231\,\mathrm{m/s}$ ($99\,\%$ quantile). Due to the large velocity error, the filter begins to diverge.

Also here, the increased residues of the ensembles affects the subsequent iterations. Nevertheless, the filter has less information to correct the velocity residues as a large variance in the measurement noise was applied to the azimuth rate and elevation rate. Hence, due to the scaling parameter γ, the residues are steadily increased. It can be assumed that the velocity error can not be corrected as fast as needed because of the applied variances of azimuth rate and elevation rate. A similar behavior is also discussed in section 3.3.3.

3.3.2. Varying number of ensemble members

The EnKF is using a set of sampled ensembles, whereas the number of ensembles is freely selectable. So, the behavior of the EnKF is tested for a various number of ensemble members. The parameters are:

- Scaling parameter γ: 0.95
- No. ensembles N: $5 \ldots 200$
- No. measurements: 200

The resulting position error is shown in Figure 9. In the test runs, where the ensemble size was $N = 5$, the filter was not able to follow the measurements. The filter diverged after several tracklets. Here, the position error increased steadily and reached a value of approximately $45.0\,\mathrm{km}$ (Figure 9) using a $95\,\%$ quantile. The velocity error reached a value of $2.2\,\mathrm{km/s}$ (Figure 10). Nevertheless, these errors can not be interpreted as fixed values as the stopping criterion of the filtering process is freely choosable. It is evident that a low, or improperly distributed number of ensembles is not able to map the uncertainty correctly, hence, the filter is diverging. In other words, if the ensembles are not properly distributed (in contrary to the sigma points), a wrong uncertainty is mapped. In the former application fields of the EnKF mentioned in the introduction, the sample size is several orders higher and the uncertainties can be mapped correctly using just a simple randomizer.

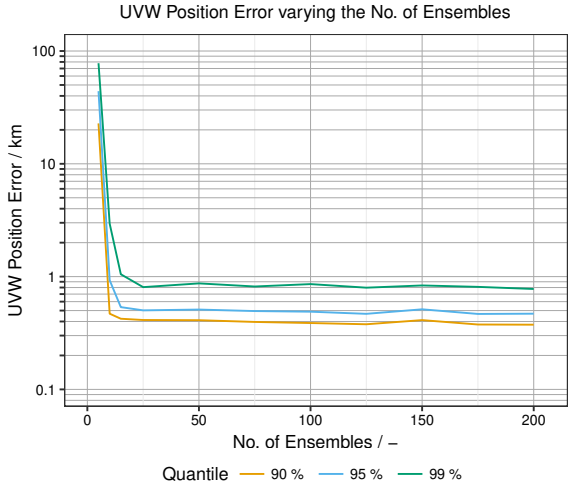

Figure 9. Position error in UVW varying the number of ensemble members.

A stable filtering process was achieved with $N = 10$ or more. The EnKF achieved in the first stable test a position error of $929\,\mathrm{m}$ ($95\,\%$ quantile). Above an ensemble number of 25, an increase of the ensemble number did not result in an steady decrease of the error. The position error varied within a range of $467\,\mathrm{m}$ to $510\,\mathrm{m}$. The minimum was obtained using an ensemble number of $N = 175$.

The error of the velocity vector is shown in Figure 10. Here, the filter showed a stable behavior equivalently for $N \geq 10$ and better convergence above 25 ensembles as well. The velocity errors are located within $2.19\,\mathrm{m/s}$ to $3.41\,\mathrm{m/s}$ for a $95\,\%$ quantile.

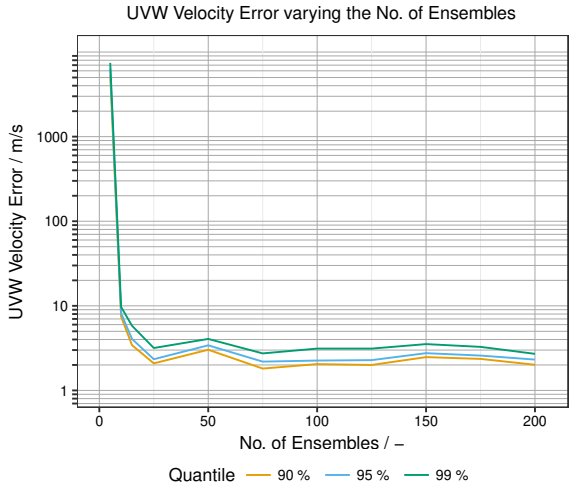

Figure 10. Velocity error in UVW varying the number of ensemble members.

3.3.3. Varying number of measurements

A high number of measurements is often not available or it differs for specific RSOs. In this test, it will be investigated, whether the EnKF is able to handle a different number of measurements per tracklet for a RSO. In contrast, the detection time among each measurement of the respective tracklet decreases with increasing number of measurements. The number of ensembles is held constant. As the previous test has shown, the filter is stable for $N \geq 10$. To ensure that the filter remains stable within this test, the ensemble number is slightly increased to $N = 25$. The parameters are:

- Scaling parameter γ: 0.95
- No. ensembles: $N = 25$
- No. measurements: $15 \ldots 1000$

Within this test, the results of the EnKF are compared with the UKF using the same values. A covariance matrix scaling is not proceeded for the UKF.

The position error using the EnKF decreases with an increasing number of measurements from $600\,\mathrm{m}$ at 15 measurements to $468\,\mathrm{m}$ at 750 measurements for a $95\,\%$ quantile (Figure 11). The filter diverges using 1000 measurements per tracklet (having lower distances among the measurements). Additionally, a further simulation described by EnKF* was performed applying an increased velocity process noise that resulted in an larger position error. But, the errors are firstly approaching a similar value of both EnKF simulations for all quantiles. Subsequently, at approximately 400 measurements, the error becomes larger with decreasing measurement distance. The filter (EnKF*) diverges at 750 measurements.

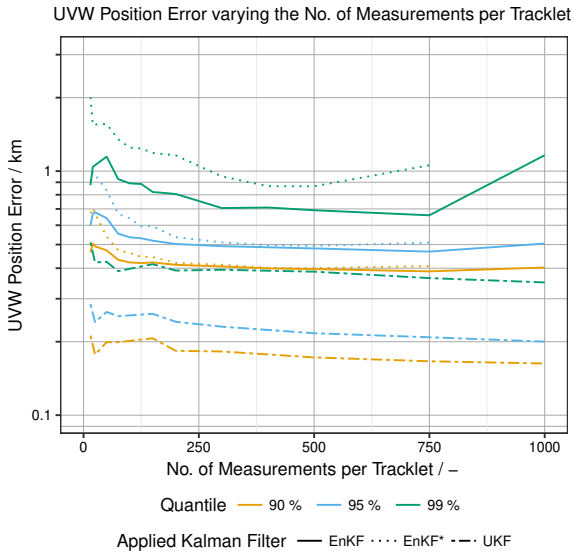

Figure 11. Position error in UVW of EnKF and UKF varying the number of measurements. The EnKF test using an increased velocity process noise is shown by EnKF.*

In contrast, the UKF yields to substantially lower position errors than the EnKF of $285\,\mathrm{m}$ with 15 measurements and $200\,\mathrm{m}$ with 1000 measurements. But the UKF is not diverging for the maximally applied number of measurements. Having a lower number of measurements up to 200, both filter show a non steady character for all quantiles.

In contrary to the position error, the velocity error (Figure 12) becomes larger for an increasing measurements number, and thus lower distances among each measurement. This behavior is visible for both filters. So, in dependence of the applied test parameters, a lower measurement number has to be preferred to achieve a low velocity error.

The velocity error of the UKF is by one order of a magnitude lower than the error of the EnKF. Using 15 measurements the velocity error amounts to $0.39\,\mathrm{m/s}$ in contrast to $1.13\,\mathrm{m/s}$ reached by the EnKF ($95\,\%$ quantile and lower velocity process noise). Increasing the number to 750, the velocity errors increase to $0.59\,\mathrm{m/s}$ and

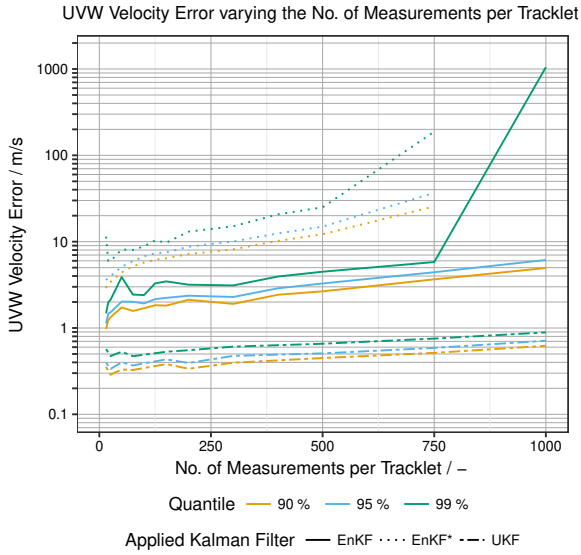

Figure 12. Velocity error in UVW of EnKF and UKF varying the number of measurements. The EnKF test using an increased velocity process noise is shown by EnKF.*

$4.43\,\mathrm{m/s}$ for UKF and EnKF, respectively. Thus, the error of the EnKF is by far larger as determined for the position error. Nevertheless, the velocity error of the EnKF using 1000 measurements increases to a value that is again several orders of magnitude larger. As previously mentioned, the filter is diverging at this point. Looking at the test EnKF*, the filter is diverging at 750 measurements.

The comparison of both process noise simulations shows that the usage of a larger velocity process noise yields to larger errors (in position and velocity). This is unexpected as a larger process noise should yield to lower errors. Repeating, the error is calculated by the difference of updated (transformed) state vector and real measurement. Having a larger process noise, the propagated state vector has to be less trustworthy by which the measurement has to be taken more into account. However, within the assumed measurement noise, the azimuth rate and elevation rate variance was set to a large value to continually trust the model. As already mentioned in section 3.3.1, by this method, less information is available to correct the velocity. Apparently, the velocity components within the state vector can not be corrected properly. The velocity error is larger over the whole measurement width. Moreover, the velocity error is added up for every available measurement due to the continually added constant process noise. Consequently, the velocity error has an impact on an increased position error, which is visible in Figure 11. Further simulations have to be performed to substantiate these statements.

3.3.4. Time effort of calculations

The calculation time of each filtering step for the UKF and EnKF is strongly dependent on the used propagator. Because a numerical propagator, instead of an analytical propagator, was used, the propagation of the state vectors took the most time. In comparison to the propagation time, the calculation time of the remaining processes is significantly lower. Consequently, the calculation time increases approximately linear with every state vector that has to be propagated. In the case of the UKF, 13 sigma points, and in the case of the EnKF, the arbitrary number of ensembles has to be propagated. So the effort of the EnKF can be lower, but in most cases, the calculation time was higher (e.g. the processing of 130 ensembles took the tenfold time).

It should be mentioned, that the filtering process was not parallelized. The substantially higher calculation time of the EnKF can be decreased at least by a parallelization of the propagator. This is possible for the UKF as well.

4. CONCLUSION

Within this paper, the basics of orbit determination using the UKF and EnKF were shown. The EnKF was implemented into the simulation environment RSS and compared against the UKF in the last proceeded test. In the first test, the scaling parameter γ was varied to inflate or to deflate the covariance matrix. For low scaling parameters, the convergence time apparently was higher, but the position and velocity errors were lower when the EnKF has converged. Having a large scaling parameter, the EnKF is diverged.

The second test comprised the variation of ensemble members. When a low number of ensembles (lower than 10) was used, the EnKF has diverged. The reason was mostly a poor distribution of the ensembles, and thus, the uncertainty could not be mapped correctly.

The variation of the number of measurements per tracklet showed, that the position error of the EnKF and UKF decreases with increasing number of measurements. In contrast to the position error, the velocity error has increased for both filters, which is mostly resulting due to the assumed measurement noise. Nevertheless, the position and velocity error of the UKF was significantly lower than the error of the EnKF.

The time effort of both filters can vary tremendously. So, the most time consuming process was the time update of every sigma point or ensemble member. Thus, the overall time effort can approximately be scaled linear depending on the number of states that have to be propagated.

The test showed that it is possible to apply the EnKF in OD having noticeably larger errors in comparison to the UKF. Further tests have to be performed to investigate the

influence of the process and measurement noise Modifications can be applied to improve the EnKF as well. For example, the usage of another sampling method for the generation of the ensembles and the random error, which was added to the propagated state vector and measurement, is conceivable. So, the impact on the behavior of the EnKF and the resulting error can be researched.

ACKNOWLEDGEMENTS

This work was supported by the Department of Mechanical Engineering at TU Braunschweig and the DLR that support the development of the simulation environment RSS as part of the contract AZA 50 LZ 1404 Entwicklung eines Radar-System-Simulators.

REFERENCES

1. Celestrak SATCAT Boxscore. `https://celestrak.com/satcat/boxscore.asp`. as of 12. Feb. 2018.
2. Kay Bergemann and Sebastian Reich. A mollified ensemble Kalman filter. *Quarterly Journal of the Royal Meteorological Society*, 136(651):1636–1643, Jul 2010. ISSN 0035-9009. doi: 10.1002/qj.672.
3. Vitali Braun. *Providing orbit information with predetermined bounded accuracy*. Logos Berlin, Berlin, 2016. ISBN 978-3-8325-4405-8.
4. Geir Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research*, 99(C5):10143, 1994. doi: 10.1029/94jc00572.
5. Geir Evensen. The Ensemble Kalman Filter: theoretical formulation and practical implementation. *Ocean Dynamics*, 53(4):343–367, Nov 2003. ISSN 1616-7228. doi: 10.1007/s10236-003-0036-9.
6. E. Gamper. Analysis and Implementation of an Unscented Kalman Filter for Orbit Determination. Studienarbeit, Institute of Space Systeme — Technische Universität Braunschweig, 2015. R 1516 S.
7. James E. Gentle. Computational statistics. *Statistics and Computing*, 2009. ISSN 1431-8784. doi: 10.1007/978-0-387-98144-4.
8. Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 3rd edition, 1996. ISBN 978-0801854149.
9. Yaqing Gu and Dean S. Oliver. The Ensemble Kalman Filter for Continuous Updating of Reservoir Simulation Models. *Journal of Energy Resources Technology*, 128(1):79, 2006. ISSN 0195-0738. doi: 10.1115/1.2134735.
10. Thomas M. Hamill, Jeffrey S. Whitaker, and Chris Snyder. Distance-dependent filtering of background error covariance estimates in an ensemble kalman filter. *Monthly Weather Review*, 129(11):2776–2790, 2001. doi: 10.1175/1520-0493(2001)129⟨2776:ddfobe⟩2.0.co;2.
11. Simon S. Haykin. *Kalman filtering and neural networks*. John Wiley & Sons, Inc., New York, 2001. ISBN 0-471-22154-6. doi: 10.1002/0471221546.
12. S. Hesselbach. Analysis and Implementation of Different Methods for Orbit Determination. Studienarbeit, Institute of Space Systeme — Technische Universität Braunschweig, 2015. R 1520 S.
13. André Horstmann, Christopher Kebschull, Sven Müller, Eduard Gamper, Sebastian Hesselbach, Kerstin Soggeberg, Mohamed Khalil Ben Larbi, Marcel Becker, Jürgen Lorenz, Carsten Wiedemann, and Enrico Stoll. Survey of the current activities in the field of modeling the space debris environment at tu braunschweig. *Aerospace*, 5(2), 2018. ISSN 2226-4310. doi: 10.3390/aerospace5020037. URL `http://www.mdpi.com/2226-4310/5/2/37`.
14. S. Horstmann. Untersuchung des GTDS Bahnbestimmungsalgorithmus. Bachelorarbeit, Institute of Space Systeme — Technische Universität Braunschweig, 2018. R 1828 B.
15. P. L. Houtekamer and Herschel L. Mitchell. A Sequential Ensemble Kalman Filter for Atmospheric Data Assimilation. *Monthly Weather Review*, 129(1):123–137, jan 2001. doi: 10.1175/1520-0493(2001)129.
16. Simon J. Julier and Jeffrey K. Uhlmann. New extension of the Kalman filter to nonlinear systems. *Signal Processing, Sensor Fusion, and Target Recognition VI*, Jul 1997. doi: 10.1117/12.280797.
17. R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35, 1960. doi: 10.1115/1.3662552.
18. Matthias Katzfuss, Jonathan R. Stroud, and Christopher K. Wikle. Understanding the Ensemble Kalman Filter. *The American Statistician*, 70(4):350–357, Oct 2016. ISSN 1537-2731. doi: 10.1080/00031305.2016.1141709.
19. Christopher Kebschull, Sven Kevin Flegel, Vitali Braun, Johannes Gelhaus, Marek Möckel, Carsten Wiedemann, and Peter Vörsmann. Reducing variability in short term orbital lifetime prediction. *Advances in Space Research*, 51(7):1110–1115, Apr 2013. ISSN 0273-1177. doi: 10.1016/j.asr.2012.11.003.
20. Christopher Kebschull, Lorenz Reichstein, and Enrico Stoll. A Simulation Environment to Determine the Performance of SSA Systems. In *Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*, 2017.
21. Heiner Klinkrad. *Space Debris*. Springer, 2006. ISBN 3-540-25448-X.
22. Lili Lei, Jeffrey L. Anderson, and Jeffrey S. Whitaker. Localizing the impact of satellite radiance observations using a global group ensemble filter. *Journal of Advances in Modeling Earth Systems*, 8(2):719–734, May 2016. ISSN 1942-2466. doi: 10.1002/2016ms000627.

23. A. C. Long, J. O. Capellari, and C. E. Velez A. J. Fuchs. Goddard Trajectory Determination System (GTDS), Mathematical Theory, Revision 1. Technical report, NASA, July 1989.
24. Vivien Mallet, Gilles Stoltz, and Boris Mauricette. Ozone ensemble forecast with machine learning algorithms. *Journal of Geophysical Research*, 114 (D5), Mar 2009. ISSN 0148-0227. doi: 10.1029/2008jd009978.
25. Linda M. McNeil and T.S. Kelso. *Spatial Temporal Information Systems: An Ontological Approach using STK®*. CRC Press, 2013. ISBN 9781466500457.
26. Oliver Montenbruck and Eberhard Gill. *Satellite Orbits: Models, Methods and Applications*. Springer, 2011. ISBN 3-540-67280-X.
27. Smriti Nandan Paul and Carolin Frueh. Space-Object Charging and Its Effect on Orbit Evolution. *Journal of Guidance, Control, and Dynamics*, 40(12):3180–3198, Dec 2017. ISSN 1533-3884. doi: 10.2514/1.g002733.
28. Jonas Radtke and Enrico Stoll. Comparing long-term projections of the space debris environment to real world data – looking back to 1990. *Acta Astronautica*, 127:482–490, 2016. doi: 10.1016/j.actaastro.2016.06.034.
29. Jonas Radtke, Christopher Kebschull, and Enrico Stoll. Interactions of the space debris environment with mega constellations—using the example of the OneWeb constellation. *Acta Astronautica*, 131:55–68, 2017. doi: 10.1016/j.actaastro.2016.11.021.
30. Afshin Rahimi, Krishna Dev Kumar, and Hekmat Alighanbari. Enhanced adaptive unscented Kalman filter for reaction wheels. *IEEE Transactions on Aerospace and Electronic Systems*, 51 (2):1568–1575, Apr 2015. ISSN 0018-9251. doi: 10.1109/taes.2014.130766.
31. Rolf H. Reichle, Jeffrey P. Walker, Randal D. Koster, and Paul R. Houser. Extended versus Ensemble Kalman Filtering for Land Data Assimilation. *Journal of Hydrometeorology*, 3(6):728–740, dec 2002. doi: 10.1175/1525-7541(2002)003.
32. B. Reihs. Analysis of LEO Observation Campagnes. Studienarbeit, Institute of Space Systeme — Technische Universität Braunschweig, 2015. R 1401 S.
33. Jon Saetrom and Henning Omre. Uncertainty Quantification in the Ensemble Kalman Filter. *Scandinavian Journal of Statistics*, 40(4):868–885, Oct 2013. ISSN 0303-6898. doi: 10.1111/sjos.12039.
34. Enrico Stoll, Klaus Merz, Holger Krag, Brian D Souza, and Benjamin Virgili. Collision Probability Assessment for the Rapideye Satellite Constellation. In *6th European Conference on Space Debris, Darmstadt*, 2013.
35. Youmin Tang, Jaison Ambandan, and Dake Chen. Nonlinear measurement function in the ensemble Kalman filter. *Advances in Atmospheric Sciences*, 31(3):551–558, Apr 2014. ISSN 1861-9533. doi: 10.1007/s00376-013-3117-9.
36. Byron D. Tapley, Bob E. Schutz, and George H. Born. *Statistical Orbit Determination*. Elsevier Academic Press, Amsterdam and Boston, 1. edition, 2004. ISBN 0-12-683630-2.
37. J. Thorey, V. Mallet, C. Chaussin, L. Descamps, and P. Blanc. Ensemble forecast of solar radiation using TIGGE weather forecasts and HelioClim database. *Solar Energy*, 120:232–243, Oct 2015. ISSN 0038-092X. doi: 10.1016/j.solener.2015.06.049.
38. David A. Vallado and Wayne D. McClain. *Fundamentals of Astrodynamics and Applications*. Space Technology Library. Microcosm Press, New York, 4th edition, 2013. ISBN 978-1881883180.