POLITECNICO DI TORINO
Repository ISTITUZIONALE

Joseph covariance formula adaptation to Square-Root Sigma-Point Kalman filters

Original
Joseph covariance formula adaptation to Square-Root Sigma-Point Kalman filters / DE VIVO, F., Brandl, A., Battipede, M., Gili, P.. - In: NONLINEAR DYNAMICS. - ISSN 0924-090X. - ELETTRONICO. - (2017). [10.1007/s11071-017-3356-x]

Availability:
This version is available at: 11583/2657935 since: 2017-05-03T17:56:48Z

Publisher:
Springer Journals

Published
DOI:10.1007/s11071-017-3356-x

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

11 September 2026

Nonlinear Dynamics manuscript No.
(will be inserted by the editor)

Joseph covariance formula adaptation to Square-Root Sigma-Point Kalman filters

Francesco De Vivo · **Alberto Brandl** · **Manuela Battipede** · **Piero Gili**

Received: date / Accepted: date

Abstract The development of a reliable Sense-And-Avoid (SAA) system is one of the limiting aspects for the integration into civil airspace of Unmanned Aerial Vehicles (UAVs), for which the market demand is becoming viral in many fields. To overcome these limitations, it is required that the Sense and Avoid (SAA) system perform equally or even better than the human eye. This can be achieved integrating information from different sensors using data-fusion algorithms, like Bayesian estimators or neural network techniques. SAA system degradation could arise from both single sensor shortcomings and bad numerical behaviours, injected by the specific fusion algorithm, such as real machine round-off errors or divergences introduced by approximating strongly non-linear functions. An alternative formulation of the Square-Root Unscented Kalman Filter (SRUKF) based on the Joseph form of the state covariance update step, is used in order to avoid numerical instabilities induced by ill-conditioned matrix problems. The novelty of this technique lies in the exploitation of the Sigma-Point Kalman Filters, which ensure a higher order accuracy in the non-linear inference problem solving, and in the application of the Joseph update equation, which improves numerical robustness. Moreover, this approach prevents the algorithm from failing, avoiding the downdating process of the Cholesky factor when the SRUKF is used and to take advantage by the higher numerical stability assured by a lower matrix Condition Number.

Keywords Kalman Filter · Sense And Avoid · UAV · Tracking · Square-Root · Cholesky · Radar · GNSS

Francesco De Vivo
Polytechnic of Turin, C.so Duca degli Abruzzi, 24, 10129 Turin, Italy
Tel.: +39 011090 6868
E-mail: francesco.devivo@polito.it

Alberto Brandl
Polytechnic of Turin
E-mail: alberto.brandl@studenti.polito.it

Manuela Battipede
Polytechnic of Turin
E-mail: manuela.battipede@polito.it

Piero Gili
Polytechnic of Turin
E-mail: piero.gili@polito.it

Nomenclature

$\mathbf{x}$	State space vector
$\mathbf{f}$	Non-linear state transition function
$\mathbf{u}$	Control vector
$\mathbf{v}$	Unmodeled dynamics
$\mathbf{z}$	Measurement vector
$\mathbf{h}$	Non-linear measurement function
$\mathbf{w}$	Measurement noise
$p(\mathbf{x}\|\mathbf{z})$	Conditional probability
$\mathbf{P^{xx}}$	State covariance matrix
$\mathbf{S^{xx}}$	Cholesky factor of $\mathbf{P^{xx}}$
$\mathbf{Q}$	Noise covariance matrix
$\mathbf{S_Q}$	Cholesky factor of $\mathbf{Q}$
$\hat{\mathbf{z}}$	Estimated measurement vector
$\mathbf{P^{zz}}$	Innovation matrix
$\mathbf{S^{zz}}$	Cholesky factor of $\mathbf{P^{zz}}$
$\mathbf{R}$	Measurement covariance matrix
$\mathbf{S_R}$	Cholesky factor of $\mathbf{R}$
$\hat{\mathbf{x}}$	Estimated state space vector
$\mathbf{P^{xz}}$	State-Measurement covariance matrix
$\mathbf{K}$	Kalman gain
$\mathbf{F}$	Linear state transition matrix
$\mathbf{H}$	Linear measurement matrix
$\mathbf{c}^{(j)}$	Normalised sigma-points
$\mathbf{X}^{(j)}$	Sigma-points
ω_j	Sigma-point weights
x, y, z	Cartesian coordinates
Δt	Time Step
q_x, q_y, q_z, q_c	Power Spectral Density
R, θ, ϕ	Range, Azimuth and Elevation
$\rho, \dot{\rho}$	Pseudorange and Pseudorange rate
$\sigma_R, \sigma_\theta, \sigma_\phi, \sigma_\rho, \sigma_{\dot{\rho}}$	Sensor error standard deviation
$\kappa(\mathbf{P})$	Condition Number of a matrix $\mathbf{P}$

1 Introduction

One of the main difficulty for the integration of Unmanned Aerial Vehicles (UAV) into civil airspaces is related to the development of a reliable Sense And Avoid (SAA) system [1–3]. This can be accomplished by requiring performance equal or better than the see-and-avoid ability of the pilot in manned aircraft. The challenge for the future Air Traffic Management (ATM) [4] system will be to dynamically manage UAVs, including structures to support 2D, 3D and 4D operations, precise navigation technologies and enhancing surveillance capabilities, fusing airborne and radar information [1].

Currently, many research fields are focused on developing state-of-the-art sensors working properly for Simultaneous Localization And Mapping (SLAM) [5], obstacle detection and surveillance, particularly when platforms with a very fast dynamics are involved, like military aircraft, rockets and controlled bombs. Passive and active MMW radar, Forward Looking Infra-Red (FLIR), LIDAR, Electronic Surveillance Module (ESM), Electro-Optical (EO) sensors [6–8], sonars are suitable systems for sensing and tracking intruders, estimating location, velocity and size of both ground and flying obstacles [9–11].

In order to compensate for individual sensors shortcomings and to provide a much reliable tracking solution, multi sensor data fusion techniques have been developed [12,2,11, 13–15]. Tirri in [33] implements a Particle Filter (PF) with dynamic motion based on Singer model integrating EO and radar sensors. In [34] a track-to-track algorithm using a linear Kalman Filter (KF) is the adopted solution for aircraft Sensing and Avoidance, with respect to Noth that proposes a ground based approach using radar [35]. Ground-Based SAA (GBSAA) systems that use electronic sensors have also been developed. These ground based systems provide information for manoeuvre decisions for terminal-area operations [36]. In [22], the tracking performance has been investigated, comparing the Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) solutions with different update covariance formulations.

As it is well known, the optimal solution to the nonlinear filtering problem is infinite dimensional. For this reason, suboptimal approaches like the EKF, UKF [16], PF [12,17], Statistically linearized filter (SLF), Gauss Hermite Kalman filter (GHKF), Cubature Kalman filter (CKF) and Spherical Simplex Kalman Filter (SSKF) [18–21] are normally considered. However, the EKF has shown several limitations and easily exhibits divergent characteristics when the system model is highly non-linear. An improvement in performance can be obtained using the Sigma-Point Kalman Filters (SPKF) like the Unscented one, which determines mean value and covariance approximating a Gaussian distribution instead of linearising a non-linear transformation. This technique is accurate to the second order, whereas the EKF is only able to obtain a first order accuracy [23]. Another cause of performance degradation and divergence of the filter is the round-off errors affecting the covariance matrix. Increasing the computation precision can be effective, however this solution is costly in computer hardware and time. To limit this problem, the Square Root version of the Sigma Point Kalman Filters (SRSPKF) was proposed [24]. Using this technique the Square Root covariance is predicted and updated, achieving better numerical accuracy. A possible drawback related to this filter is that the negative update, necessary in the classical covariance matrix formulation, might destroy the positive definiteness property of the Cholesky factor. To try to overcome this problem, in [24] the information form of the filter (SRUIF) was proposed. An alternative solution was suggested in [37], where a manipulation of the covariance matrix was carried out in order to solve problems related to the existence of negative matrix eigenvalues.

In this work an novel method is proposed to avoid numerical instabilities and to limit the effect of the round-off errors. The adopted solution makes use of the Joseph formulation for the state covariance matrix updating, applied to the SRSPKFs. After the calculation is performed using the EKF and UKF, a Square-Root version of the Joseph covariance matrix formula has been derived. In this way it was possible to improve the numerical accuracy and to avoid problems related to the negative updating of the Cholesky factor, not required by this algorithm. In the following sections firstly the Bayesian inference is described from a theoretical point of view; the proposed methodology is then presented and finally the simulation results are shown.

2 Bayesian inference

In a discrete dynamic process, the current state of the system is dependent on one or more prior states. When observations are provided at discrete times, estimation conditioned on those observations can only occur at those times [18]. Considering a first order Markov process, it is possible to describe a random Markov dynamic process as

$$\mathbf{x}_n = \mathbf{f}_{n-1}(\mathbf{x}_{n-1}) + \mathbf{u}_n + \mathbf{v}_{n-1} \tag{1}$$

where $\mathbf{x}_n$ is the state of the system at time t_n, $\mathbf{f}_{n-1}$ is a deterministic transition function that moves the state $\mathbf{x}$ from time t_{n-1} to time t_n, $\mathbf{u}_n$ is a known control vector and $\mathbf{v}_{n-1}$ is a white noise vector describing uncertainties about unmodeled dynamics. The goal is to estimate the unobservable state vector $\mathbf{x}_n$, based on the set of all experimental observation vectors $\mathbf{z}_{1:n} = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n\}$. It is assumed that an analytical relationship is known between the observation vector at time t_n and the state vector at time t_n, represented by

$$\mathbf{z}_n = \mathbf{h}_n(\mathbf{x}_n) + \mathbf{w}_n. \tag{2}$$

Here, $\mathbf{z}_n$ is designated as the observation vector and $\mathbf{h}_n$ is a deterministic observation function linking the state vector with the observation and $\boldsymbol{w}_n$ is a white noise vector (not necessarily Gaussian) representative of the sensor accuracy. Equations 1 and 2 represent a complete model of the system and the inference can be turned into an estimation of the conditional posterior density $p(\mathbf{x}_n|\mathbf{z}_{1:n})$ using the Bayes' law

$$p(\mathbf{x}_n|\mathbf{z}_{1:n}) = \frac{p(\mathbf{z}_n|\mathbf{x}_n)p(\mathbf{x}_n|\mathbf{z}_{1:n-1})}{p(\mathbf{z}_n|\mathbf{z}_{1:n-1})} \tag{3}$$

where

$$p(\mathbf{x}_n|\mathbf{z}_{1:n-1}) = \int p(\mathbf{x}_n|\mathbf{x}_{n-1})p(\mathbf{x}_{n-1}|\mathbf{z}_{1:n-1})d\mathbf{x}_{n-1}. \tag{4}$$

From equation 3 and 4, a recursive link has been established between the previous posterior $p(\mathbf{x}_{n-1}|\mathbf{z}_{1:n-1})$ and the current posterior $p(\mathbf{x}_n|\mathbf{z}_{1:n})$ [18] as desired.

2.1 Recursive estimation of mean and covariance

Using equations 3 and 4 it is possible to obtain $\mathbf{x}_{n|n}$ and $\mathbf{P^{xx}}_{n|n}$ conditioned on all observations up to the time step t_n, as shown hereinafter. The state prediction is obtained as

$$\hat{\mathbf{x}}_{n|n-1} = \int_{\mathbb{R}^{n_x}} \mathbf{f}_{n-1}(\mathbf{x}_{n-1})p(\mathbf{x}_{n-1}|\mathbf{z}_{1:n-1})d\mathbf{x}_{n-1} + \mathbf{u}_n + \int_{\mathbb{R}^{n_x}} \mathbf{v}_{n-1}p(\mathbf{x}_{n-1}|\mathbf{z}_{1:n-1})d\mathbf{x}_{n-1}. \tag{5}$$

Defining

$$\tilde{\mathbf{x}}_{n-1|n-1} = \left[\mathbf{f}_{n-1}(\mathbf{x}_{n-1}) + \mathbf{u}_n - \hat{\mathbf{x}}_{n|n-1}\right]$$

the state covariance matrix is

$$\mathbf{P^{xx}}_{n|n-1} = \int_{\mathbb{R}^{n_x}} [\tilde{\mathbf{x}}_{n-1|n-1}][\tilde{\mathbf{x}}_{n-1|n-1}]^T \times p(\mathbf{x}_n|\mathbf{z}_{1:n-1})d\mathbf{x}_{n-1} + \mathbf{Q} \tag{6}$$

where the noise covariance matrix $\mathbf{Q}$ is

$$\mathbf{Q} = \int_{\mathbb{R}^{n_x}} \mathbf{v}_{n-1}\mathbf{v}_{n-1}^T p(\mathbf{x}_n|\mathbf{z}_{1:n-1})d\mathbf{x}_{n-1}. \tag{7}$$

The estimation of the measurement vector is obtained as

$$\hat{\mathbf{z}}_{n|n-1} = \int_{\mathbb{R}^{n_x}} \mathbf{h}_n(\mathbf{x}_{n|n-1})p(\mathbf{x}_n|\mathbf{z}_{1:n-1})d\mathbf{x}_n + \int_{\mathbb{R}^{n_x}} \mathbf{w}_n p(\mathbf{x}_n|\mathbf{z}_{1:n-1})d\mathbf{x}_n. \tag{8}$$

Defining

$$\tilde{\mathbf{z}}_{n|n-1} = \left[\mathbf{h}_n(\mathbf{x}_{n-1}) - \hat{\mathbf{z}}_{n|n-1}\right]$$

the innovation covariance matrix is provided by the following relation

$$\mathbf{P^{zz}}_{n|n-1} = \int_{\mathbb{R}^{n_x}} [\tilde{\mathbf{z}}_{n|n-1}][\tilde{\mathbf{z}}_{n|n-1}]^T \times p(\mathbf{x}_n|\mathbf{z}_{1:n-1})d\mathbf{x}_n + \mathbf{R} \tag{9}$$

where $\mathbf{R}$ is

$$\mathbf{R} = \int_{\mathbb{R}^{n_x}} \mathbf{w}_n\mathbf{w}_n^T p(\mathbf{x}_n|\mathbf{z}_{1:n-1})d\mathbf{x}_n. \tag{10}$$

The covariance matrix between states and measurements is

$$\mathbf{P^{xz}}_{n|n-1} = \int_{\mathbb{R}^{n_x}} [\tilde{\mathbf{x}}_{n-1|n-1}][\tilde{\mathbf{z}}_{n|n-1}]^T \times p(\mathbf{x}_n|\mathbf{z}_{1:n-1})d\mathbf{x}_n \tag{11}$$

and the Kalman gain $\mathbf{K_n}$ is given by

$$\mathbf{K_n} = \mathbf{P^{xz}}_{n|n-1}\left[\mathbf{P^{zz}}_{n|n-1}\right]^{-1}. \tag{12}$$

At this point, it is possible to update the state vector and the covariance matrix as

$$\hat{\mathbf{x}}_{n|n} = \hat{\mathbf{x}}_{n|n-1} + \mathbf{K_n}\left(\mathbf{z_n} - \hat{\mathbf{z}}_{n|n-1}\right) \tag{13}$$

$$\mathbf{P^{xx}}_{n|n} = \mathbf{P^{xx}}_{n|n-1} - \mathbf{K_n}\mathbf{P^{zz}}_{n|n-1}\mathbf{K_n}^T. \tag{14}$$

3 Methodology

Particularizing the previous relations in case of linear state and measurement equations we get the LKF algorithm shown in Table 1. When the state or measurement equations are non-linear, this algorithm can be used by linearising them by means of a Taylor series expansion, getting the EKF. In this case the state transition matrix $\mathbf{F}$ and the measurement matrix $\mathbf{H}$ are substituted by the respective Jacobian matrices $\mathbf{F_J}$ and $\mathbf{H_J}$.

More accurate filtering solutions are based on the numerical estimation of the integrals shown in equations 5-11. Filters

Table 1 LKF algorithm

$\hat{\mathbf{x}}_{n\|n-1}$	$= \mathbf{F}\hat{\mathbf{x}}_{n-1\|n-1} + \mathbf{u}_n$
$\mathbf{P^{xx}}_{n\|n-1}$	$= \mathbf{F}\mathbf{P^{xx}}_{n-1\|n-1}\mathbf{F}^T + \mathbf{Q}$
$\hat{\mathbf{z}}_{n\|n-1}$	$= \mathbf{H}\hat{\mathbf{x}}_{n\|n-1}$
$\mathbf{P^{zz}}_{n\|n-1}$	$= \mathbf{H}\mathbf{P^{xx}}_{n\|n-1}\mathbf{H}^T + \mathbf{R}$
$\mathbf{P^{xz}}_{n\|n-1}$	$= \mathbf{P^{xx}}_{n\|n-1}\mathbf{H}^T$
$\mathbf{K_n}$	$= \mathbf{P^{xz}}_{n\|n-1}(\mathbf{P^{zz}}_{n\|n-1})^{-1}$
$\hat{\mathbf{x}}_{n\|n}$	$= \hat{\mathbf{x}}_{n\|n-1} + \mathbf{K_n}(\mathbf{z} - \hat{\mathbf{z}}_{n\|n-1})$
$\mathbf{P^{xx}}_{n\|n}$	$= \mathbf{P^{xx}}_{n\|n-1} - \mathbf{K_n}\mathbf{P^{zz}}_{n\|n-1}\mathbf{K_n}^T$

Table 2 SPKF algorithm

$\mathbf{X}^{(j)}_{n-1\|n-1}$	$= \hat{\mathbf{x}}_{n-1\|n-1} + \sqrt{\mathbf{P^{xx}}_{n-1\|n-1}}\mathbf{c}^{(j)}$
$\hat{\mathbf{x}}_{n\|n-1}$	$= \sum_{j=0}^{ns} \omega_j \mathbf{f}(\mathbf{X}^{(j)}_{n-1\|n-1}) + \mathbf{u}_n$
$\mathbf{P^{xx}}_{n\|n-1}$	$= \sum_{j=0}^{ns} \omega_j \left[\mathbf{f}(\mathbf{X}^{(j)}_{n-1\|n-1}) - \hat{\mathbf{x}}_{n\|n-1}\right]$
	$\times \left[\mathbf{f}(\mathbf{X}^{(j)}_{n-1\|n-1}) - \hat{\mathbf{x}}_{n\|n-1}\right]^T + \mathbf{Q}$
$\mathbf{X}^{(j)}_{n\|n-1}$	$= \hat{\mathbf{x}}_{n\|n-1} + \sqrt{\mathbf{P^{xx}}_{n\|n-1}}\mathbf{c}^{(j)}$
$\hat{\mathbf{z}}_{n\|n-1}$	$= \sum_{j=0}^{ns} \omega_j \mathbf{h}(\mathbf{X}^{(j)}_{n\|n-1})$
$\mathbf{P^{zz}}_{n\|n-1}$	$= \sum_{j=0}^{ns} \omega_j \left[\mathbf{h}(\mathbf{X}^{(j)}_{n\|n-1}) - \hat{\mathbf{z}}_{n\|n-1}\right]$
	$\times \left[\mathbf{h}(\mathbf{X}^{(j)}_{n\|n-1}) - \hat{\mathbf{z}}_{n\|n-1}\right]^T + \mathbf{R}$
$\mathbf{P^{xz}}_{n\|n-1}$	$= \sum_{j=0}^{ns} \omega_j \left[\mathbf{f}(\mathbf{X}^{(j)}_{n-1\|n-1}) - \hat{\mathbf{x}}_{n\|n-1}\right]$
	$\times \left[\mathbf{h}(\mathbf{X}^{(j)}_{n\|n-1}) - \hat{\mathbf{z}}_{n\|n-1}\right]^T$
$\mathbf{K_n}$	$= \mathbf{P^{xz}}_{n\|n-1}(\mathbf{P^{zz}}_{n\|n-1})^{-1}$
$\hat{\mathbf{x}}_{n\|n}$	$= \hat{\mathbf{x}}_{n\|n-1} + \mathbf{K_n}(\mathbf{z} - \hat{\mathbf{z}}_{n\|n-1})$
$\mathbf{P^{xx}}_{n\|n}$	$= \mathbf{P^{xx}}_{n\|n-1} - \mathbf{K_n}\mathbf{P^{zz}}_{n\|n-1}\mathbf{K_n}^T$

using these techniques are called SPKFs. The SPKF algorithm is shown in Table 2 where *ns* is the number of normalised sigma-points $\mathbf{c}^{(j)}$, and ω_j are the weights. In order to find a compete procedure to calculate them see [18].
Using the formulation of the equation 14 for the state covariance matrix update $\mathbf{P^{xx}}_{n|n}$, a matrix subtraction is performed. This operation could generate numerical errors, which can cause even the loss of its positive definiteness. The alternative form (equation 15), is known as the Joseph form covariance update, which is less sensitive to round-off errors [19]

$$\mathbf{P^{xx}}_{n|n} = (\mathbf{I} - \mathbf{K_n}\mathbf{H})\mathbf{P^{xx}}_{n|n-1}(\mathbf{I} - \mathbf{K_n}\mathbf{H})^T + \mathbf{K_n}\mathbf{R}\mathbf{K_n}^T \quad (15)$$

where *I* is the identity matrix and $\mathbf{H}$ is the measurement matrix or equivalently the Jacobian of the measurement function $\mathbf{h}(\mathbf{x})$, indicated above as $\mathbf{H_J}$. With the proper implementation of the products of three matrices, the symmetry is preserved. Furthermore, since the only operation of matrix subtraction is in the term $(\mathbf{I} - \mathbf{K_n}\mathbf{H})$, which appears squared, this form of the covariance updating has the property of preserving the positive definiteness [19]. This formulation can be applied in the case of LKF or EKF once $\mathbf{H}$ or $\mathbf{H_J}$ have been defined, but it can not be used for the SPKFs. As mentioned above, the common technique adopted in this case to increase the filter numerical stability is to use a Square-Root formulation (SRKF). This filter requires to perform the square root of the covariance matrix which needs symmetry and positive definiteness. These two properties might be lost due to errors introduced by arithmetic operations, performed on finite word-length digital computers, or ill-conditioned non-linear filtering problems [24]. A feasible first approach to estimate $\mathbf{H}$ in order to make appropriate the use of the Joseph formula is to start from the definition of $\mathbf{P^{xz}}_{n|n-1}$ for the LKF

$$\mathbf{P^{xz}}_{n|n-1} = \mathbf{P^{xx}}_{n|n-1}\mathbf{H}^T.$$

Inverting this relation in order to explicit $\mathbf{H}$ we obtain

$$\mathbf{H} = \left[\left(\mathbf{P^{xx}}_{n|n-1}\right)^{-1}\mathbf{P^{xz}}_{n|n-1}\right]^T \quad (16)$$

that can be seen as a Jacobian matrix, where $\mathbf{P^{xx}}_{n|n-1}$ and $\mathbf{P^{xz}}_{n|n-1}$ are estimated using the SPKF. Beside the computational cost required for the matrix inversion, a problem might arise in inverting the matrix $\mathbf{P^{xx}}_{n|n-1}$ when this is ill-conditioned. The proposed method, instead, makes use of the Jacobian matrix $\mathbf{H_J}$ of the non-linear measurement function $\mathbf{h}(\mathbf{x})$ estimated at the point $\hat{\mathbf{x}}_{n|n}$

$$\mathbf{H_J} = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}\Big|_{\mathbf{x}=\hat{\mathbf{x}}_{n|n}}. \quad (17)$$

Calculating $\mathbf{H_J}$ after the state updating step, enables a much more accurate matrix to be calculated, compared to the one used for the EKF, estimated at $\hat{\mathbf{x}}_{n|n-1}$. This improvement derives from the lower $\hat{\mathbf{x}}_{n|n}$ variance, ensured by the measurement information available at this step, with respect to $\hat{\mathbf{x}}_{n|n-1}$ which prediction is based only on the accuracy of the model chosen to represent the target dynamics. Once $\mathbf{H_J}$ is obtained, the Joseph form can be applied

$$\mathbf{P^{xx}_J}_{n|n} = (\mathbf{I} - \mathbf{K_n}\mathbf{H_J})\mathbf{P^{xx}}_{n|n-1}(\mathbf{I} - \mathbf{K_n}\mathbf{H_J})^T + \mathbf{K_n}\mathbf{R}\mathbf{K_n}^T. \quad (18)$$

Using this approach, higher filtering performance is provided and, at the same time, numerical stability is improved by the Joseph formulation, at the expense of a slightly computational higher cost, as can be seen in Table 3.

Table 3 Filters Computational Complexity. n is the state space vector dimension and m the number of measurements [25]

Algorithm	Math Operations			
	$+$	$\times$	$\div$	$\sqrt{\cdot}$
Conventional $P^x - KP^zK^T$	$(1.5n^2+3.5n)m$	$(1.5n^2+4.5n)m$	m	0
U-D factorization $P^x = UDU^T$	$0.5n^2-0.5n$ $+(1.5n^2+1.5n)m$	n^2-n $+(1.5n^2+5.5n)m$	nm	0
Triangular Cov. Square Root	$0.5n^2+0.5n$ $+(1.5n^2+3.5n)m$	$0.5n^2+0.5n$ $+(2n^2+5n)m$	$2nm$	nm
Kalman Stable Joseph	$(4.5n^2+5.5n)m$	$(4n^2+7.5n)m$	m	0

3.1 Square-Root SPKF

In order to improve filter performance and numerical stability, a triangular Covariance Square-Root form of the Equation 18 has been derived, in order to apply the Square-Root filter. As extensively demonstrated in literature, using the Square-Root formulation, the matrix Condition Number is strongly reduced, improving the filter numerical behaviour. As introduced in the previous paragraph, the Joseph covariance matrix updating does not require a subtraction as for the classical one, avoiding the risks of losing the matrix positiveness associated with the negative Cholesky updating step. Defining

$$\mathbf{S} = chol\{\mathbf{P}\} \mid \mathbf{S}\mathbf{S}^T = \mathbf{P}$$

and:

$$\begin{aligned} \mathbf{S^{xx}}_{n-1|n-1} &= chol\left\{\mathbf{P^{xx}}_{n-1|n-1}\right\} \\ \mathbf{S_Q} &= chol\{\mathbf{Q}\} \\ \mathbf{S_R} &= chol\{\mathbf{R}\} \end{aligned}$$

it is possible to estimate the state vector

$$\mathbf{X}^{(j)}_{n-1|n-1} = \hat{\mathbf{x}}_{n-1|n-1} + \sqrt{\mathbf{P^{xx}}_{n-1|n-1}}\mathbf{c}^{(j)} \tag{19}$$

$$\hat{\mathbf{x}}_{n|n-1} = \sum_{j=0}^{ns} \omega_j \mathbf{f}(\mathbf{X}^{(j)}_{n-1|n-1}) + \mathbf{u}_n \tag{20}$$

where $\mathbf{c}^{(j)}$ are the normalised sigma-points and ω_j are the sigma-point weights. In order to predict the state covariance matrix, the upper triangular matrix obtained from the QR-Orthogonal-triangular decomposition is used;

$$\mathbf{S^{xx}}^T_{n|n-1} = qr\left\{\begin{bmatrix} \sqrt{\omega_1}(\mathbf{X}^{(1)}_{n-1|n-1} - \hat{\mathbf{x}}_{n|n-1})^T \\ \sqrt{\omega_2}(\mathbf{X}^{(2)}_{n-1|n-1} - \hat{\mathbf{x}}_{n|n-1})^T \\ \vdots \\ \sqrt{\omega_j}(\mathbf{X}^{(j)}_{n-1|n-1} - \hat{\mathbf{x}}_{n|n-1})^T \\ \vdots \\ \sqrt{\omega_{ns}}(\mathbf{X}^{(ns)}_{n-1|n-1} - \hat{\mathbf{x}}_{n|n-1})^T \\ \mathbf{S_Q^T} \end{bmatrix}\right\} \tag{21}$$

$$\mathbf{S^{xx}}_{n|n-1} = cholupdate\left\{\mathbf{S^{xx}}_{n|n-1}, (\mathbf{X}^{(0)}_{n-1|n-1} - \hat{\mathbf{x}}_{n|n-1}), \omega_0\right\}. \tag{22}$$

Now it is possible to update the sigma-points and predict the measurement vector as

$$\mathbf{X}^{(j)}_{n|n-1} = \hat{\mathbf{x}}_{n|n-1} + \sqrt{\mathbf{P^{xx}}_{n|n-1}}\mathbf{c}^{(j)} \tag{23}$$

$$\hat{\mathbf{z}}_{n|n-1} = \sum_{j=0}^{ns} \omega_j \mathbf{h}(\mathbf{X}^{(j)}_{n|n-1}). \tag{24}$$

The innovation matrix is calculated using the same technique adopted for predicting the covariance matrix

$$\mathbf{S^{zz}}^T_{n|n-1} = qr\left\{\begin{bmatrix} \sqrt{\omega_1}(\mathbf{h}(\mathbf{X}^{(1)}_{n|n-1}) - \hat{\mathbf{z}}_{n|n-1})^T \\ \sqrt{\omega_2}(\mathbf{h}(\mathbf{X}^{(2)}_{n|n-1}) - \hat{\mathbf{z}}_{n|n-1})^T \\ \vdots \\ \sqrt{\omega_j}(\mathbf{h}(\mathbf{X}^{(j)}_{n|n-1}) - \hat{\mathbf{z}}_{n|n-1})^T \\ \vdots \\ \sqrt{\omega_{ns}}(\mathbf{h}(\mathbf{X}^{(ns)}_{n|n-1}) - \hat{\mathbf{z}}_{n|n-1})^T \\ \mathbf{S_R^T} \end{bmatrix}\right\} \tag{25}$$

$$\mathbf{S^{zz}}_{n|n-1} = cholupdate\left\{\mathbf{S^{zz}}_{n|n-1}, (\mathbf{h}(\mathbf{X}^{(0)}_{n|n-1}) - \hat{\mathbf{z}}_{n|n-1}), \omega_0\right\} \tag{26}$$

$$\mathbf{P^{xz}}_{n|n-1} = \sum_{j=0}^{ns} \omega_j \left[\mathbf{f}(\mathbf{X}^{(j)}_{n-1|n-1}) - \hat{\mathbf{x}}_{n|n-1}\right] \times \left[\mathbf{h}(\mathbf{X}^{(j)}_{n|n-1}) - \hat{\mathbf{z}}_{n|n-1}\right]^T. \tag{27}$$

The Kalman gain can be calculated as

$$\mathbf{K_n} = \left(\mathbf{P^{xz}}_{n|n-1} / \mathbf{S^{zzT}}_{n|n-1}\right) / \mathbf{S^{zz}}_{n|n-1}. \quad (28)$$

Once the Kalman gain is available, it is possible to update the state vector

$$\hat{\mathbf{x}}_{n|n} = \hat{\mathbf{x}}_{n|n-1} + \mathbf{K_n}(\mathbf{z} - \hat{\mathbf{z}}_{n|n-1}). \quad (29)$$

In order to use the Joseph formulation, firstly the Jacobian matrix $\mathbf{H_J}$ must be calculated as shown in Equation 17, secondarily the *QR*-decomposition algorithm can be applied to a generic matrix $\mathbf{A}$, defined as

$$\mathbf{S} = \{qr\{\mathbf{A}\}\}^T = chol\left\{\mathbf{A}^T\mathbf{A}\right\} \quad (30)$$

Writing $\mathbf{A}$ as

$$\mathbf{A} = \mathbf{S^{xxT}}_{n|n-1}(\mathbf{I} - \mathbf{K_n H_J})^T,$$

and introducing $\mathbf{G} = (\mathbf{I} - \mathbf{K_n H_J})$ for simplicity, we obtain

$$\begin{aligned}\mathbf{A}^T\mathbf{A} &= \mathbf{G}\underbrace{\mathbf{S^{xx}}_{n|n-1}\mathbf{S^{xxT}}_{n|n-1}}_{\mathbf{P^{xx}}_{n|n-1}}\mathbf{G}^T \\ &= (\mathbf{I} - \mathbf{K_n H_J})\mathbf{P^{xx}}_{n|n-1}(\mathbf{I} - \mathbf{K_n H_J})^T \quad (31)\end{aligned}$$

that is the first term of the Joseph covariance matrix update. Applying the *QR*-decomposition to the matrix $\mathbf{A}$, as defined in the Equation 30, it is possible to obtain the Cholesky matrix of this term. To take into account also the second term, it is necessary to apply the *QR*-decomposition as it follows

$$\mathbf{S^{xx}}_{n|n} = qr\left\{\begin{bmatrix}\mathbf{S^{xxT}}_{n|n-1}(\mathbf{I} - \mathbf{K_n H_J})^T \\ \mathbf{S_R^T K_n^T}\end{bmatrix}\right\}. \quad (32)$$

In order to obtain the correct solution, it is necessary to take the upper triangular matrix given by the the *qr* MATLAB function.

4 Model Validation

4.1 Dynamic model

This simulation concerns the tracking of an aircraft from a fixed sensor in a Cartesian space. Assuming a constant accelerated dynamics, the discrete time state space equation is described by the following linear relation

$$\mathbf{x}_n = \mathbf{F}_{n-1}\mathbf{x}_{n-1} + \mathbf{v}_{n-1} \quad (33)$$

where the state space vector is the following

$$\mathbf{x} = [x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}, z, \dot{z}, \ddot{z}]^T$$

with $\dot{x}$ and $\ddot{x}$ indicating respectively the first and second time derivatives and the state transition matrix $\mathbf{F}_{n-1}$ is defined as

$$\mathbf{F} = \begin{bmatrix}\mathbf{F_1} & 0 & 0 \\ 0 & \mathbf{F_1} & 0 \\ 0 & 0 & \mathbf{F_1}\end{bmatrix}$$

with:

$$\mathbf{F_1} = \begin{bmatrix}1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 1 & \Delta t \\ 0 & 0 & 1\end{bmatrix}$$

and Δt is the time step. The noise is assumed to be independent, with zero-mean and Gaussian $\mathbf{v}_{n-1} \backsim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ with

$$\mathbf{Q} = \begin{bmatrix}q_x\mathbf{Q_1} & 0 & 0 \\ 0 & q_y\mathbf{Q_1} & 0 \\ 0 & 0 & q_z\mathbf{Q_1}\end{bmatrix}$$

and

$$\mathbf{Q_1} = \begin{bmatrix}\frac{\Delta t^5}{20} & \frac{\Delta t^4}{8} & \frac{\Delta t^3}{6} \\ \frac{\Delta t^4}{8} & \frac{\Delta t^3}{6} & \frac{\Delta t^2}{2} \\ \frac{\Delta t^3}{6} & \frac{\Delta t^2}{2} & \Delta t\end{bmatrix} \quad (34)$$

where q_x, q_y and q_z, are the noise Power Spectral Densities (PSD) along each Cartesian direction.

4.2 Sensor Model

The sensor provides measurements of the range, azimuth and elevation angles

$$\mathbf{z} = [R, \theta, \phi]^T.$$

From the nature of measured variables it is clear that the function $\mathbf{h}(\mathbf{x})$ relating these measurements to the state vector is non-linear

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{w}_n \quad (35)$$

where

$$\mathbf{h}(\mathbf{x}) = \begin{bmatrix} \sqrt{x^2+y^2+z^2} \\ \arctan\left(\frac{x}{y}\right) \\ \arctan\left(\frac{z}{\sqrt{x^2+y^2}}\right) \end{bmatrix}. \tag{36}$$

The observation noise $\mathbf{w}_n$ is considered to be independent, with zero-mean and Gaussian $\mathbf{w}_n \backsim \mathcal{N}(\mathbf{0}, \mathbf{R})$ with

$$\mathbf{R} = \begin{bmatrix} \sigma_R^2 & 0 & 0 \\ 0 & \sigma_\theta^2 & 0 \\ 0 & 0 & \sigma_\phi^2 \end{bmatrix}$$

where σ_R, σ_θ and σ_ϕ are the sensor error standard deviations. In this case the Jacobian matrix $\mathbf{H_J}$ is given by

$$\mathbf{H_J} = \begin{bmatrix} \frac{\partial \mathbf{h}(1)}{\partial x} & 0 & 0 & \frac{\partial \mathbf{h}(1)}{\partial y} & 0 & 0 & \frac{\partial \mathbf{h}(1)}{\partial z} & 0 & 0 \\ \frac{\partial \mathbf{h}(2)}{\partial x} & 0 & 0 & \frac{\partial \mathbf{h}(2)}{\partial y} & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial \mathbf{h}(3)}{\partial x} & 0 & 0 & \frac{\partial \mathbf{h}(3)}{\partial y} & 0 & 0 & \frac{\partial \mathbf{h}(3)}{\partial z} & 0 & 0 \end{bmatrix} \tag{37}$$

where

$$\begin{aligned}
\frac{\partial \mathbf{h}(1)}{\partial x} &= \frac{x}{\sqrt{x^2+y^2+z^2}} \\
\frac{\partial \mathbf{h}(1)}{\partial y} &= \frac{y}{\sqrt{x^2+y^2+z^2}} \\
\frac{\partial \mathbf{h}(1)}{\partial z} &= \frac{z}{\sqrt{x^2+y^2+z^2}} \\
\frac{\partial \mathbf{h}(2)}{\partial x} &= \frac{y}{x^2+y^2} \\
\frac{\partial \mathbf{h}(2)}{\partial y} &= -\frac{x}{x^2+y^2} \\
\frac{\partial \mathbf{h}(3)}{\partial x} &= -\frac{xz}{\sqrt{x^2+y^2}(x^2+y^2+z^2)} \\
\frac{\partial \mathbf{h}(3)}{\partial y} &= -\frac{yz}{\sqrt{x^2+y^2}(x^2+y^2+z^2)} \\
\frac{\partial \mathbf{h}(3)}{\partial z} &= \frac{\sqrt{x^2+y^2}}{x^2+y^2+z^2}
\end{aligned}$$

4.3 Numerical results

In this section the results obtained from the MATLAB numerical simulations are presented. For the case under study we have considered a sensor sampling frequency of 2 Hz and the three standard deviations respectively of 30 m in range, 1.45 deg in azimuth and 5 deg in elevation. The theoretical trajectory, the measurements and the filtered solution are shown in Fig. 1. Each figure is generated using LaTeX.

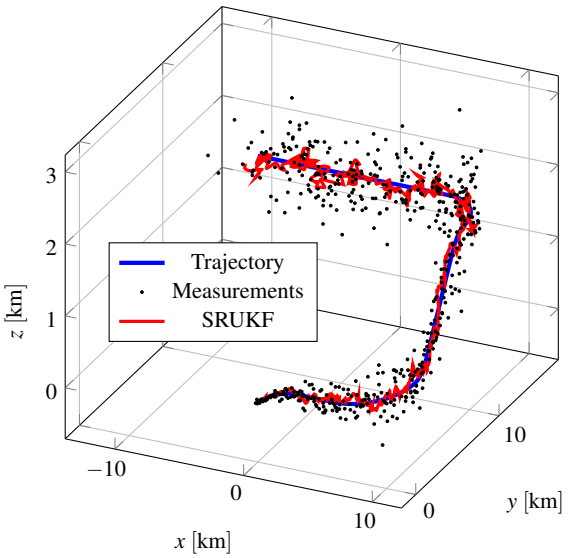

Fig. 1 Aircraft trajectory

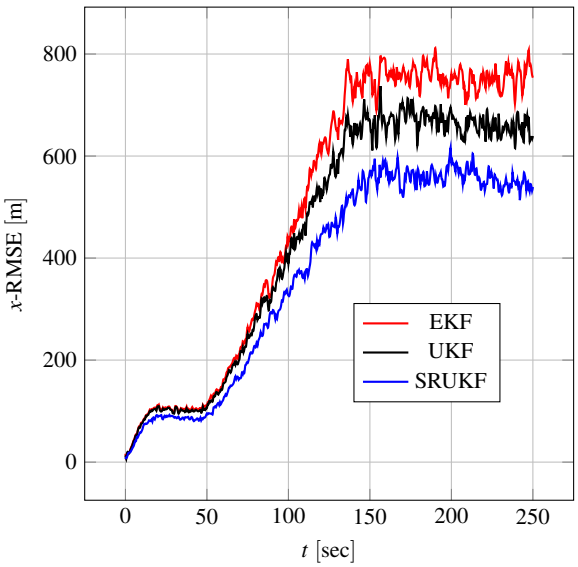

Fig. 2 RMSE of the x coordinate from 500 Monte Carlo runs for the EKF, UKF and SRUKF using Joseph formulation

In order to get statistically consistent results, 500 Monte Carlo (MC) simulations were performed. As it can be seen from the estimated position Root Mean Square Error (RMSE) in Fig. 2, 3 and 4, the UKF exhibits better performance than the EKF version, as expected, although the best accuracy is obtained by applying the Joseph formula to the SRUKF, as confirmed also by outcomes in Fig. 5.

A key result that validates the potentiality of the present approach is shown in Fig. 6 and 7. With the EKF method in the classical formulation, the ill-conditioned covariance matrix makes the Condition Number diverge after 75 sec, with

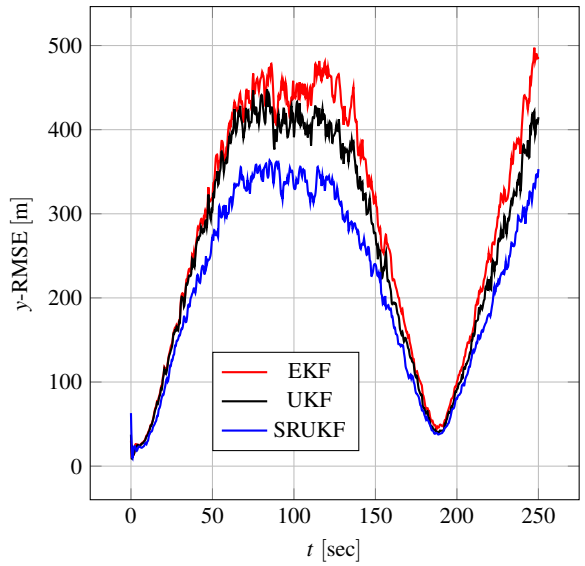

Fig. 3 RMSE of the y coordinate from 500 Monte Carlo runs for the EKF, UKF and SRUKF using Joseph formulation

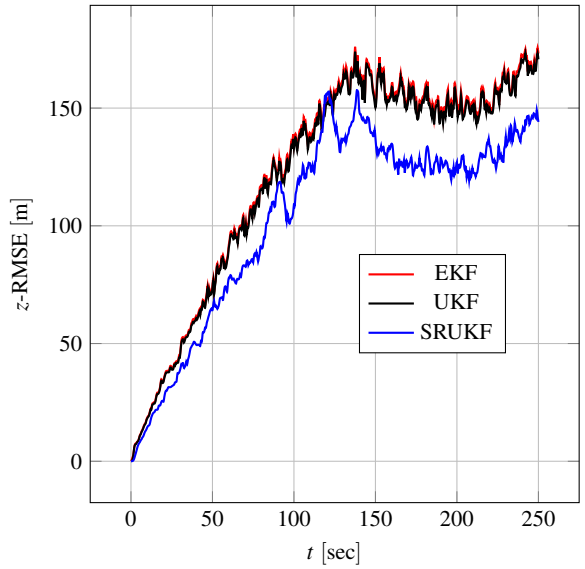

Fig. 4 RMSE of the z coordinate from 500 Monte Carlo runs for the EKF, UKF and SRUKF using Joseph formulation

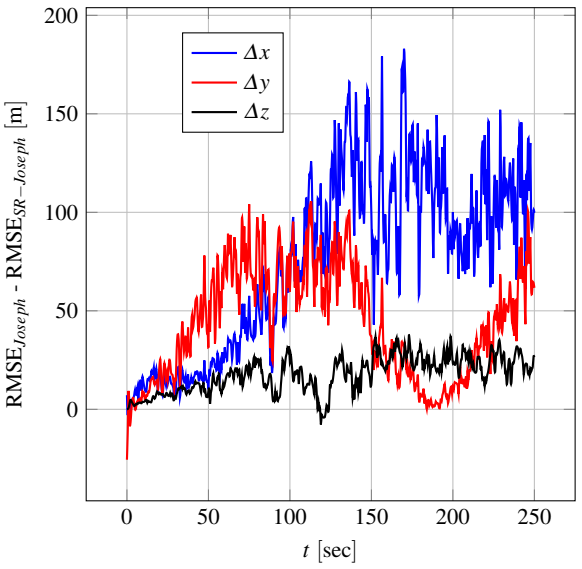

Fig. 5 Difference between the position RMSEs from 500 Monte Carlo runs obtained using the UKF and SRUKF with the Joseph covariance formulation

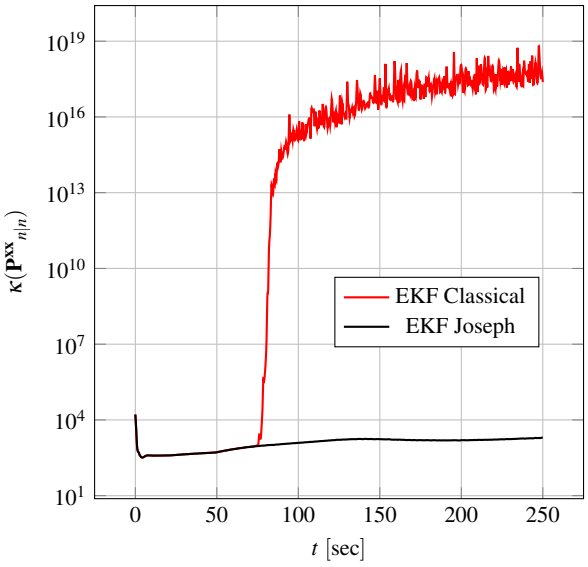

Fig. 6 Condition Number of the state covariance matrix $\mathbf{P}^{\mathbf{xx}}{}_{n|n}$ for the EKF using classical and Joseph formulation from 500 Monte Carlo runs

direct effects on the accuracy of the filtered solution. This can be confirmed by a rule of thumb stating that if the condition number is expressed as $\kappa(\mathbf{P}) = 10^k$, a precision loss of at least k digits can be expected [26]. Better outcomes are guaranteed by the UKF algorithm. Using the Joseph formulation, moreover, it is possible to obtain much better performance with both the EKF and UKF. Using the SRUKF, the Condition Number is reduced by two orders of magnitude with respect to the UKF, ensuring a higher numerical accuracy. The real advantage in applying the Joseph covariance formula to SRUKF is not strictly related to the κ-reduction: it lies in the capability of preserving the positive definiteness, when the downdating process is applied to the classical formulation. This encouraging results open the possibility of the application of this technique to manage situations where strong function non-linearities make the EKF inapplicable and where ill-conditioned problems could lead to filter instabilities, if the classical updating relation is used. Filter comparisons, in terms of computational time, is shown in Fig. 10. These results are obtained using an Intel Core i7-

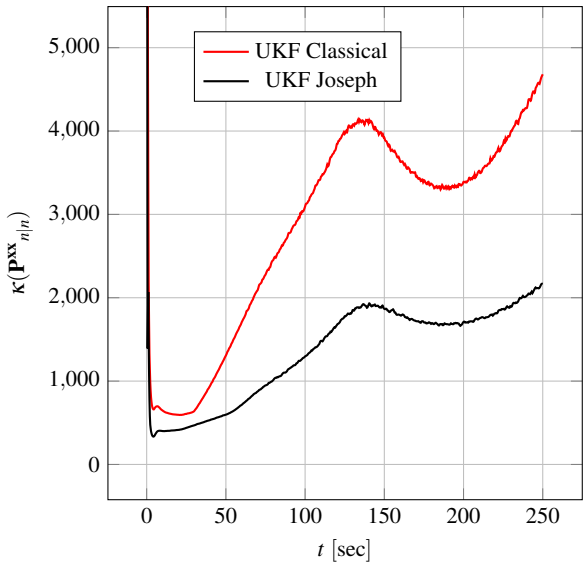

Fig. 7 Condition Number of the state covariance matrix $\mathbf{P}^{\mathbf{xx}}{}_{n|n}$ for the UKF using classical and Joseph formulation from 500 Monte Carlo runs

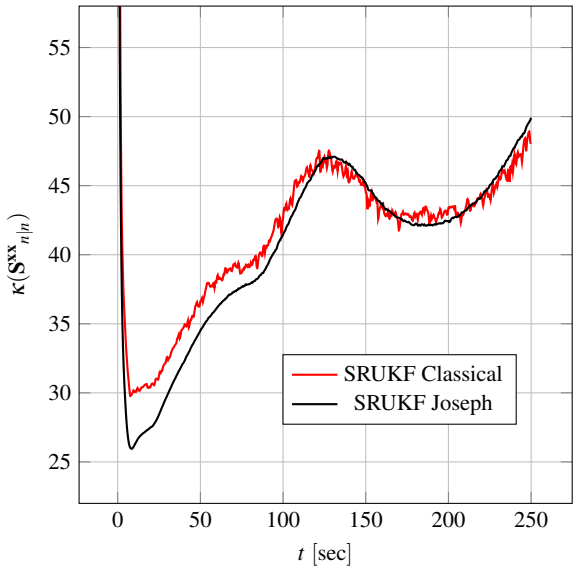

Fig. 8 Condition Number of $\mathbf{S}^{\mathbf{xx}}{}_{n|n}$ for the SRUKF using classical and Joseph formulation from 500 Monte Carlo runs

4710HQ 64 bit, 2.50 GHz with 16 GB RAM. It is evident that the proposed technique has a negligible impact, being the simulation time mainly affected by filter typology rather than covariance updating formula.

5 Test Cases

In the previous section the new algorithm has been validated using an ideal situation with a linear dynamics corrupted by a simplified noise model. This section has the aim to apply

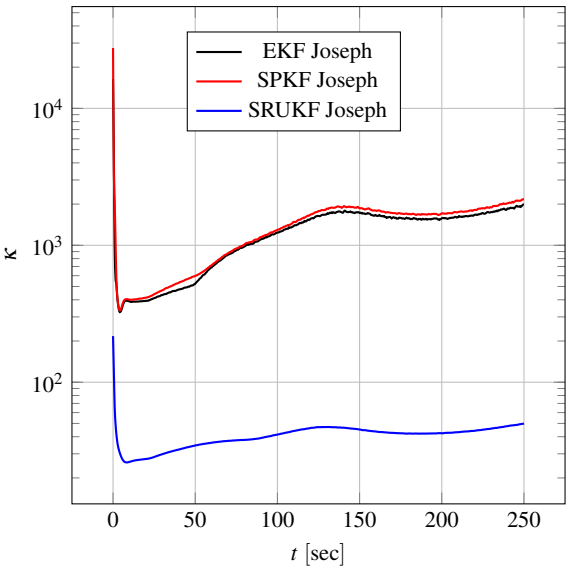

Fig. 9 Condition Number of the state covariance matrix $\mathbf{P}^{\mathbf{xx}}{}_{n|n}$ using Joseph formulation for EKF and UKF and of $\mathbf{S}^{\mathbf{xx}}{}_{n|n}$ using Joseph formulation for SRUKF from 500 Monte Carlo runs

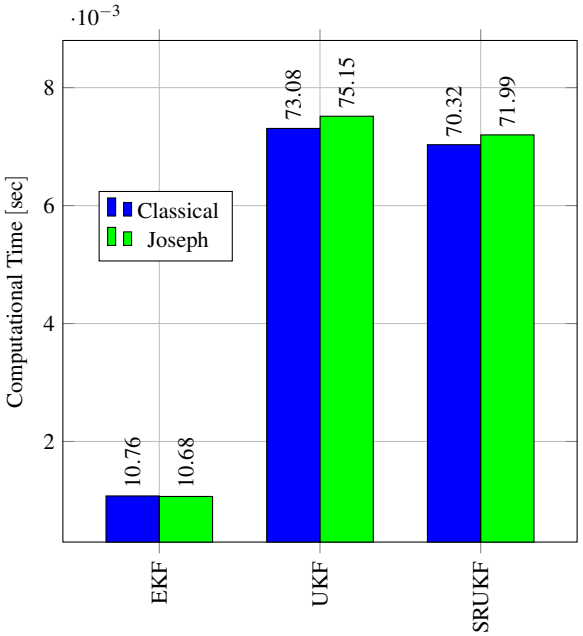

Fig. 10 Algorithms computational time from 5000 Monte Carlo runs

the proposed filter to two realistic situations where both dynamic and measurements models are non-linear and noise has a more realistic distribution. The first one is the tracking of an aircraft using a ground based radar, the second one is the tracking performed by means of an Automatic Dependant Surveillance-Broadcast (ADS-B) technique, where the cooperative intruder provides information about the proper GNSS position.

5.1 Ground based RADAR data filtering

5.1.1 Dynamic model

In order to improve the modelling of the tracked object dynamics, a non-linear variable turn model has been assumed. Using the Fundamental Relation of Kinematics (FRK), it is possible to obtain the derivative of a time-dependent vector as shown in the following equation

$$\frac{d\mathbf{u}^I}{dt} = \frac{d\mathbf{u}^B}{dt} + \boldsymbol{\Omega}_{BI} \times \mathbf{u} \tag{38}$$

where $\mathbf{u}^I$ and $\mathbf{u}^B$ represent the vector expressed respectively in the inertial and body reference frame, and $\boldsymbol{\Omega}_{BI}$ is the angular velocity vector of B with respect to I frame. Applying the FRK to the target velocity $\mathbf{v}$ we get the target acceleration

$$\mathbf{a} = \frac{d\mathbf{v}^B}{dt} + \boldsymbol{\Omega} \times \mathbf{v}. \tag{39}$$

Applying again the FRK, it is possible to obtain the target jerk equation

$$\dot{\mathbf{a}} = \frac{d^2\mathbf{v}^B}{dt^2} + \dot{\boldsymbol{\Omega}} \times \mathbf{v} + 2\boldsymbol{\Omega} \times \mathbf{v} - \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \mathbf{v}). \tag{40}$$

With the hypothesis of $\boldsymbol{\Omega} \perp \mathbf{v}$, $\boldsymbol{\Omega}$ can be written as follow

$$\boldsymbol{\Omega} = \frac{\mathbf{v} \times \mathbf{a}}{\|\mathbf{v}\|^2}. \tag{41}$$

Substituting Equation 41 into 40, it is possible to model the target acceleration as a second order Markov process with state dependent coefficients [32]

$$\dot{\mathbf{a}} = -2\alpha\mathbf{a} - (2\alpha^2 + \omega^2)\mathbf{v} + \mathbf{v}_{n-1} \tag{42}$$

where

$$\omega = \frac{\|\mathbf{v} \times \mathbf{a}\|}{\|\mathbf{v}\|^2}, \quad \alpha = -\frac{\mathbf{v} \cdot \mathbf{a}}{\|\mathbf{a}\|^2}, \quad \mathbf{v}_{n-1} = \frac{d^2\mathbf{v}^B}{dt^2} + \dot{\boldsymbol{\Omega}} \times \mathbf{v}.$$

Rearranging Equation 42 in a matrix form, the state equation becomes

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{A}(\omega_c, \alpha) & 0 & 0 \\ 0 & \mathbf{A}(\omega_c, \alpha) & 0 \\ 0 & 0 & \mathbf{A}(\omega_c, \alpha) \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{B} \\ \mathbf{B} \\ \mathbf{B} \end{bmatrix} \mathbf{v}_{n-1} \tag{43}$$

where

$$\mathbf{A}(\omega_c, \alpha) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -(\alpha^2 + \omega_c^2) & -2\alpha \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

and $\omega_c = \alpha^2 + \omega^2$. During filtering process, this non-linear differential equation is integrated using a 4th order Runge-Kutta algorithm. For this dynamic model, $\mathbf{Q_1}$ matrix is different from that defined in the previous paragraph. In order to find an analytical form for it, ω_c and α have been considered constant within each time step. Applying Equation 7 for this model, it is possible to calculate $\mathbf{Q_1}$ as follows

$$\mathbf{Q_1}(\omega_c, \alpha, \Delta t) = \int_0^{\Delta t} \left(e^{\mathbf{A}(\omega_c,\alpha)\tau}\mathbf{B}\right)\left(e^{\mathbf{A}(\omega_c,\alpha)\tau}\mathbf{B}\right)^T d\tau \tag{44}$$

where $e^{\mathbf{A}(\omega_c,\alpha)\tau}$ is the state space exponential matrix. Each term of $\mathbf{Q_1}$ is in Appendix A.

5.1.2 Measurement model

The measurement equations are the same shown in the previous paragraph, where the sensor provides information about the target range, elevation and azimuth. Differences are in a more accurate and realistic noise modelling and in considering a non-constant covariance matrix $\mathbf{R}$. The radar model considered here is a phased array radar. Radar accuracy is given by the following set of equations

$$\sigma_i = \sqrt{\sigma_{i,N}^2 + \sigma_{i,F}^2 + \sigma_{i,B}^2} \qquad \text{with } i = \{R, \theta, \phi\} \tag{45}$$

$\sigma_{i,N}$ is the Signal-to-Noise S/N-dependent random range measurement error, $\sigma_{i,F}$ is the fixed random error and $\sigma_{i,B}$ is the bias error. In explicit form, the terms in Equations 45 are

$$\begin{aligned}
&\sigma_{RN} = \frac{\Delta R}{\sqrt{2\,(S/N)}} && \sigma_{RF} \simeq \frac{\Delta R}{25} && \sigma_{RB} \approx 0 \\
&\sigma_{\theta N} = \frac{\Delta\theta}{k_M \cos\varphi_\theta \sqrt{2\,(S/N)}} && \sigma_{\theta F} \simeq \frac{\Delta\theta}{40\cos\varphi_\theta} && \sigma_{\theta B} \approx 0 \\
&\sigma_{\phi N} = \frac{\Delta\phi}{k_M \cos\varphi_\phi \sqrt{2\,(S/N)}} && \sigma_{\phi F} \simeq \frac{\Delta\phi}{40\cos\varphi_\phi} && \sigma_{\phi B} \approx 0
\end{aligned}$$

where $\Delta R = c/2B$ is the range resolution, B the signal bandwidth defined for a phased array as

$$B = \frac{max\{\Delta\theta, \Delta\phi\}\, c}{0.886\lambda \sin(min\{\varphi_\theta, \varphi_\phi\})},$$

c is the speed of light, $\Delta\theta$ and $\Delta\phi$ are the phased-array beamwidth on broadside, φ_θ and φ_ϕ are the off-broadside

scan angles. k_M is the monopulse pattern difference slope, λ the signal wavelength, the Signal-to-Noise ratio is

$$S/N = \frac{P_p \tau G_t^2 \lambda^2 \sigma}{(4\pi)^3 R^4 k T_s L} \tag{46}$$

where P_p is the peak power of the antenna at the transmitter, τ the pulse duration, G_t is the antenna gain, σ is the target Radar Cross Section (RCS), $k = 1.3807 \times 10^{-23}\,\mathrm{JK}^{-1}$ is the Boltzmann constant, T_s is the system noise temperature and L the radar losses in dB, given by

$$L = L_{mt} + L_{mr} + L_s + L_p + L_{lens} + L_{sp} + L_{rain} + L_{bs}.$$

In the previous relation, L_{mt} and L_{mr} are the microwave losses at transmitting and receiving antenna that, in this case, are coincident, L_s is the scan loss

$$L_s \approx [\cos(\mid \phi_r - \phi \mid)\cos(\mid \theta_r - \Phi \mid)]^{-2.5}$$

where ϕ_r and θ_r are the azimuth and elevation of the radar beam with zero scan angle, and Φ is the target elevation angle above the horizon

$$\Phi = \arcsin\left[-\frac{R^2 + R_e^2 - (R_e + z)^2}{2RR_e}\right].$$

R_e in the above equation is $4/3$ the Earth radius. L_p is the path loss, L_{lens} the lens loss, L_{rain} the rain loss, L_{sp} is the loss due to suboptimal signal processing and L_{bs} is the beamshape loss. To calculate the extent of these losses refer to [31]. Once the S/N ratio for a single pulse is calculated, it is possible to integrate it for n_p pulses assuming a coherent integration

$$(S/N)_{CI} = n_p S/N \frac{S/N}{(1 + S/N) L_i}$$

where L_i is the integration loss. This value of S/N can be used to calculate the radar accuracy in Equation 45. Once the standard deviation has been obtained using the previous algorithm, the $\mathbf{R}$ matrix can be computed for each time step. The Jacobian matrix $\mathbf{H}_J$ is equal to that defined in Equation 37, being the measurement equations the same. In order to test the filter performance outside the nominal conditions, the value of the rain intensity has been changed and the signal has been jammed using the following relation for calculating the new S/N value

$$\frac{S}{J+N} = \frac{\frac{P_p \tau G_t G_r \lambda^2 \sigma}{(4\pi)^3 R^4 L_t L_r}}{\frac{ERP G_r \lambda^2}{(4\pi R)^2 B_J L_{pol} L_r} + kT_s}$$

where G_r is the receiving antenna gain, L_t and L_r are the losses at transmitting and receiving antenna, L_{pol} is the polarization loss assumed to be 0 dB, B_J is the bandwidth of the jamming signal and ERP is the Effective Radiated Power.

5.1.3 Numerical results

For the radar test case, sensor parameters have been listed in Table 4. In Fig. 11 the simulated trajectory is illustrated, together with the radar noisy measurements. The figure also shows when the jamming and signal losses occur. Consequences of these two faults can be seen from the first graph of Fig. 12. The radar measurements have an update rate of 10 Hz. As it can be seen from the second plot of Fig. 12, the condition number κ of the EKF diverges when the signal is jammed, about 70 sec after the simulation starts, and UKF diverges when the signal is lost. On the contrary, the Joseph formulation of the Square Root UKF prevents filter divergence over the entire simulation, confirming the higher stability features, already proved previously. Better filter performance is also confirmed by Fig. 13, where deviation of the filtered acceleration from the real value is lower than those estimated by EKF and UKF. Fig. 14 shows how the Root Mean Square Error of the SRUKF is halved with respect to the input radar noise also during out-nominal condition.

Table 4 RADAR input parameters

Param.	Value	Units	Param.	Value	Units
θ_r	0	deg	B_J	0.15	GHz
ϕ_r	20	deg	n_p	10	-
P_p	2.5	kW	$\varphi_{\theta\,max}$	± 60	deg
τ	50	μ-sec	$\varphi_{\phi\,max}$	± 60	deg
G_t	39.5	dB	L_{mt}	1.5	dB
G_r	39.5	dB	L_{mr}	1.5	dB
f_0	3.3	GHz	L_{sp}	1	dB
T_s	500	K	L_i	1	dB
k_M	1.6	-	L_{bs}	3.2	dB
$\Delta\theta$	2	deg	σ	0.1	dBsm
$\Delta\phi$	2	deg	ERP	15	W

5.2 GNSS data filtering

5.2.1 Dynamic model

The dynamic model assumed here is the non-linear second order Markov process already used for the radar test case, where the state vector is augmented with two state variables, in order to estimate the clock error and the clock error drift [27] of the GNSS receiver

$$\mathbf{x}_{GNSS} = \left[\mathbf{x}^T \; C_b \; C_d\right]^T. \tag{47}$$

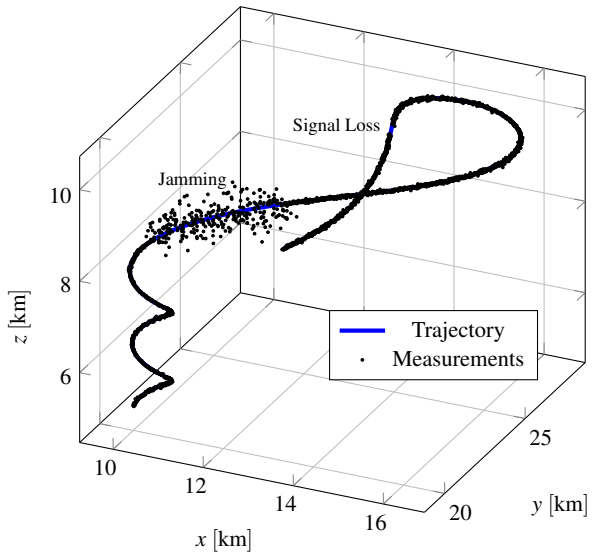

Fig. 11 Aircraft trajectory

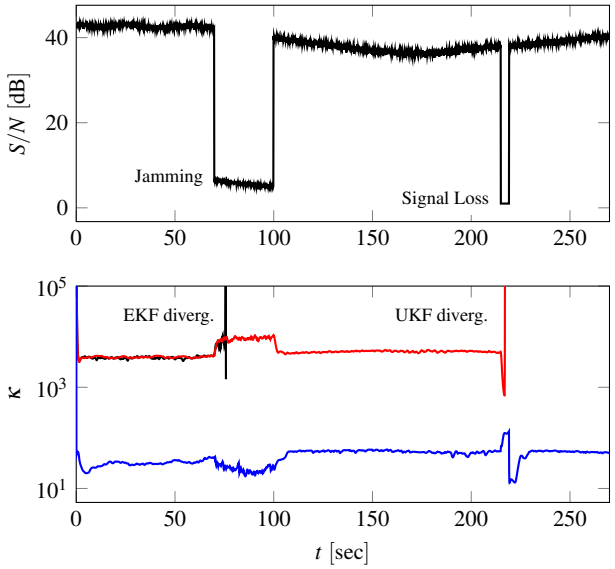

Fig. 12 RADAR Signal-to-Noise ratio and Condition Number κ of the state covariance matrix $\mathbf{P}^{\mathbf{xx}}{}_{n|n}$ using Joseph formulation for the —EKF, —UKF and $\mathbf{S}^{\mathbf{xx}}{}_{n|n}$ for —SRUKF

The state vector $\mathbf{x}_{GNSS}$ is expressed in the ECEF reference frame and hereinafter it will be indicated simply by $\mathbf{x}$. The state representation of the augmented model is

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{A}(\omega_c,\alpha) & 0 & 0 & 0\ 0 \\ 0 & \mathbf{A}(\omega_c,\alpha) & 0 & 0\ 0 \\ 0 & 0 & \mathbf{A}(\omega_c,\alpha) & 0\ 0 \\ 0 & 0 & 0 & 0\ 1 \\ 0 & 0 & 0 & 0\ 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{B} \\ \mathbf{B} \\ \mathbf{B} \\ 0 \\ 1 \end{bmatrix} \mathbf{v}_{n-1} \quad (48)$$

where $\mathbf{A}(\omega_c,\alpha)$ and $\mathbf{B}$ are those expressed in the previous section. The system noise covariance matrix $\mathbf{Q}$ instead is

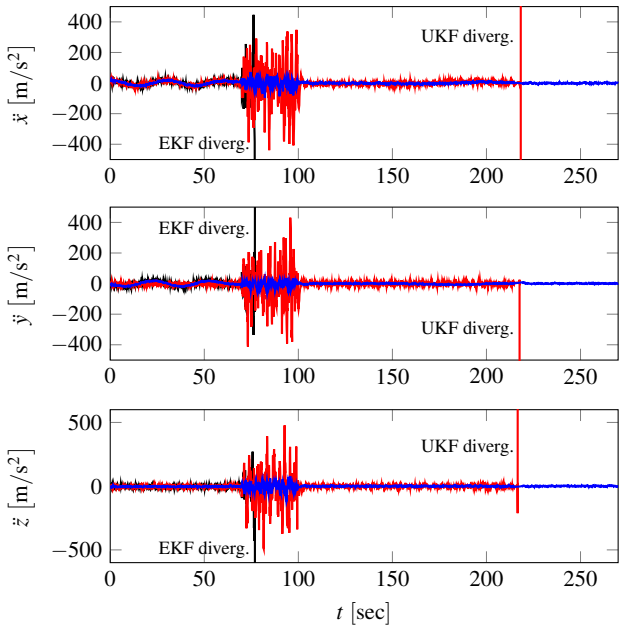

Fig. 13 Filtered acceleration using Joseph formulation for —EKF, —UKF and —SRUKF for RADAR measurements

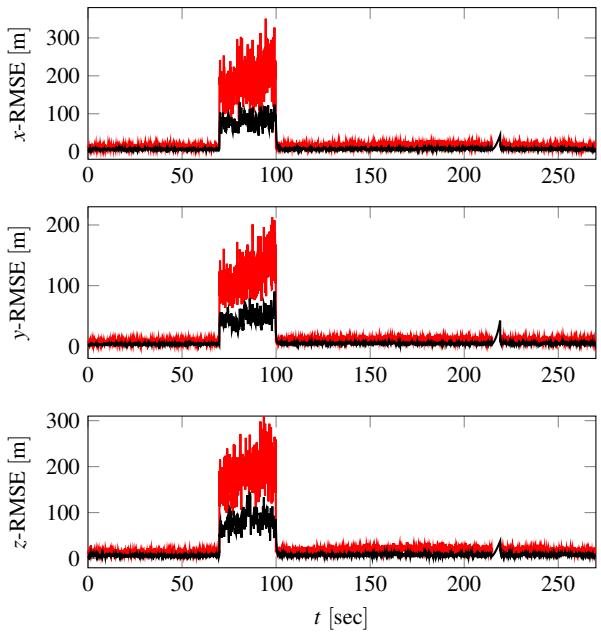

Fig. 14 RMSE between —RADAR measurements and filtered solution from 10 Monte Carlo runs using Joseph formulation of the —SRUKF

$$\mathbf{Q} = \begin{bmatrix} q_x\mathbf{Q_1}(\omega_c,\alpha,\Delta t) & 0 & 0 & 0 \\ 0 & q_y\mathbf{Q_1}(\omega_c,\alpha,\Delta t) & 0 & 0 \\ 0 & 0 & q_z\mathbf{Q_1}(\omega_c,\alpha,\Delta t) & 0 \\ 0 & 0 & 0 & q_c\mathbf{Q_c}(\Delta t) \end{bmatrix} \quad (49)$$

where $\mathbf{Q_c}$ is

$$\mathbf{Q_c} = \begin{bmatrix} \frac{\Delta t^3}{3} & \frac{\Delta t^2}{2} \\ \frac{\Delta t^2}{2} & \Delta t \end{bmatrix}$$

and q_c is the Power Spectral Density of the clock drift.

5.2.2 Measurement model

The non-linear measurement model is composed by the pseudorange and the pseudorange rate between the GNSS constellation [28] and the on-board receiver as shown below

$$\mathbf{z} = \begin{bmatrix} \rho_1 \ \dots \ \rho_{Ns} \ \dot{\rho}_1 \ \dots \ \dot{\rho}_{Ns} \end{bmatrix}^T \tag{50}$$

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{w}_n$$

$$\mathbf{h}(\mathbf{x}) = \begin{bmatrix} \hat{\rho}_1 \ \dots \ \hat{\rho}_{Ns} \ \dot{\hat{\rho}}_1 \ \dots \ \dot{\hat{\rho}}_{Ns} \end{bmatrix}^T . \tag{51}$$

$\hat{\rho}_i$ and $\dot{\hat{\rho}}_i$ are defined as

$$\hat{\rho}_{i=1\dots Ns} = R_{i=1\dots Ns} + C_b$$
$$\dot{\hat{\rho}}_{i=1\dots Ns} = \dot{R}_{i=1\dots Ns} + C_d$$

where Ns is the number of satellites, R_i and $\dot{R}_i$ are respectively

$$R_i = \sqrt{[x_{s,i}(t) - x(t)]^2 + [y_{s,i}(t) - y(t)]^2 + [z_{s,i}(t) - z(t)]^2}$$
$$\dot{R}_i = \frac{1}{R_i} [(\dot{x}_{s,i} - \dot{x})(x_{s,i} - x) + (\dot{y}_{s,i} - \dot{y})(y_{s,i} - y) + (\dot{z}_{s,i} - \dot{z})(z_{s,i} - z)]$$

where for $\dot{R}_i$ the time dependency has been omitted for clarity. In order to derive a simpler form for the Jacobian matrix $\mathbf{H_J}$, a linear dynamic model has been assumed for the state variables,

$$\left\{ \begin{array}{l} \xi(t) = \xi + \dot{\xi}\Delta t + \frac{1}{2}\ddot{\xi}\Delta t^2 \\ \dot{\xi}(t) = \dot{\xi} + \ddot{\xi}\Delta t \end{array} \middle| \ \xi \in \{ x \ y \ z \} \right\} \tag{52}$$

obtaining the components shown in Appendix B.
GNSS accuracy can be expressed as a function of different error sources

$$\sigma_\rho = \sigma_\rho \left(\delta_{\rho,clock} \ \delta_{\rho,atm} \ \delta_{\rho,rec} \ \delta_{\rho,mp} \right)$$
$$\sigma_{\dot{\rho}} = \sigma_{\dot{\rho}} \left(\delta_{\dot{\rho},clock} \ \delta_{\dot{\rho},atm} \ \delta_{\dot{\rho},rec} \right)$$

where $\delta_{i,clock}$, $\delta_{i,atm}$, $\delta_{i,rec}$ and $\delta_{i,mp}$, with $i = \{\rho, \dot{\rho}\}$, are respectively the clock, tropospheric, receiver and multipath errors. Clock and receiver errors are modelled as a zero mean white noise, whereas $\delta_{\rho,atm}$ is obtained by means of Niell Mapping Function (NMF)

$$\delta_{\rho,atm} = T_{dry}(z_{NED}) M_{dry}(E) + T_{wet} M_{wet}(E) \tag{53}$$

where T_{dry} and T_{wet} represent respectively dry and wet delay at altitude z_{NED}, which are given by

$$T_{dry} = a e^{-b z_{NED}}$$
$$T_{wet} = T_{0,wet} + \Delta T_{wet}$$

where $a = 2.3\,\text{m}$, $b = 0.116 \times 10^{-3}\,\text{m}^{-1}$, $T_{0,wet} = 0.1\,\text{m}$ and ΔT_{wet} is a random walk process which features a PSD of $1\,\text{cm}^2/\text{hour}$. For M_{dry} and M_{wet} formulation see [30,38]. Applying the Standard Multipath Error Model (SMEM) endorsed by International Civil Aviation Organization (ICAO), it is possible to calculate $\delta_{\rho,mp}$

$$\delta_{\rho,mp} = 0.13 + 0.53 e^{-\frac{E}{10}}$$

where E is the satellite elevation in radians [29]. In order to test the filter performance outside nominal conditions, two different kind of faults have been injected. The first one is a clock degradation and the second one is a Phase Lock Loop (PLL) fault.

5.2.3 Numerical results

Parameters for GNSS sensor have been chosen in order to simulate a Commercial Off-The-Shelf receiver, with clock accuracy of 500 ns, frequency jitter of 0.3 Hz, bit resolution of 1 % of the chip rate and an update rate of 1 Hz. The simulated trajectory is the same of Paragraph 5.1.3 with the difference that the NED reference frame origin is fictitiously located in a point of latitude 45.2°-N and longitude 7.6°-E. The trajectory geo-localization is necessary to express it in ECEF coordinates. The GPS constellation is obtained by the almanac of the 29th February 2016. As an update rate of 10 Hz has been chosen for the filter, whereas the actual measurement occurs with a rate of 1 Hz, it has been necessary to estimate the state vector in absence of measurements propagating the dynamic equation 43. Similarly to the previous case, two faults have been simulated: a clock accuracy degradation (Fig. 15) and a PLL failure, which corresponds to loosing all the satellites in view. The clock degradation failure affects the filter performance more critically, causing the divergence of the EKF and UKF even if the Joseph formulation is used. Nevertheless, the algorithm proposed in this paper remains stable after both failures. Evidence of the

improved filter performance is shown also in Fig. 16 and 17, except for the z-coordinate of the RMSE, for which the state estimation is noisier when the clock degradation is injected.

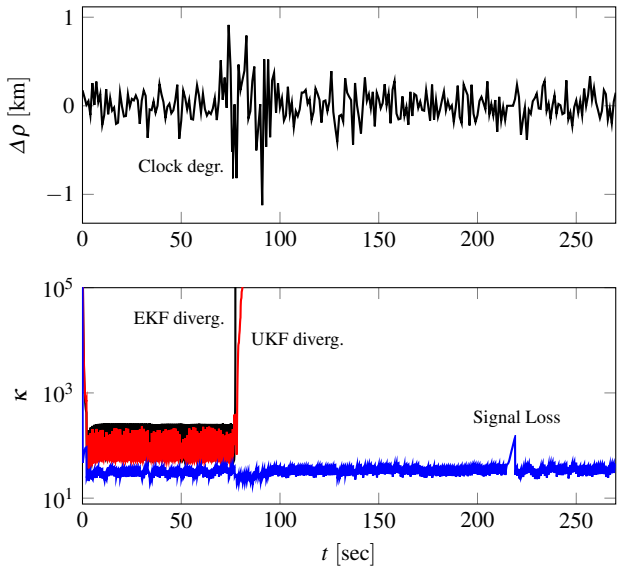

Fig. 15 GNSS pseudorange error and Condition Number κ of the state covariance matrix $\mathbf{P}^{\mathbf{xx}}{}_{n|n}$ using Joseph formulation for the —EKF, —UKF and $\mathbf{S}^{\mathbf{xx}}{}_{n|n}$ for —SRUKF

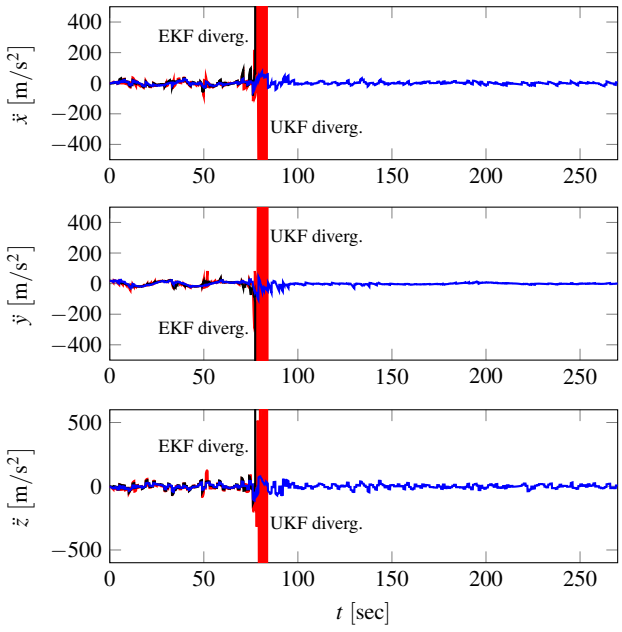

Fig. 16 Filtered acceleration using Joseph formulation for —EKF, —UKF and —SRUKF for GNSS measurements

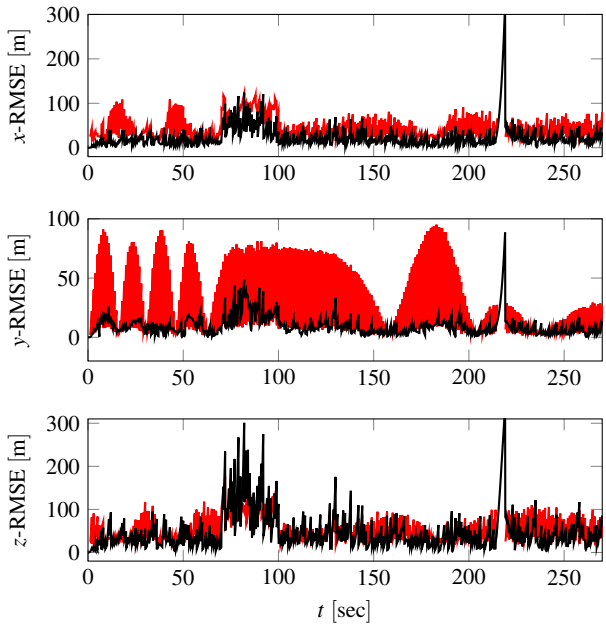

Fig. 17 RMSE between —GNSS measurements and filtered solution from 10 Monte Carlo runs using Joseph formulation of the —SRUKF

6 Conclusions

This paper presents an alternative approach to improve filter numerical stability for non-linear estimation and sensor fusion. This technique is based on the Joseph formulation of the state covariance matrix. It guarantees better numerical properties, such as improved numerical stability and preservation of symmetry, as well as a higher order accuracy in the state estimation. A further positive aspect is that it prevents divergences when the equations are strongly non-linear as shown in radar and GNSS simulation. This approach is also of great interest in solving positive definiteness issues of the Square-Root algorithm, avoiding the negative Cholesky updating. The numerical results show how estimated values by SRUKF are much more accurate than those obtained by EKF and UKF. The matrix condition number is improved using the Joseph formula and becomes even better if the latter is combined with the Square-Root algorithm, that never diverges even in case of signal degradation. Furthermore, another important aspect is that the computational time is not affected by the Joseph approach.

Conflict of Interest

Conflict of Interest: The authors declare that they have no conflict of interest.

Appendix A $\mathbf{Q_1}$ Matrix

Solving the Equation 44 we get the following expressions for the $\mathbf{Q_1}$ matrix components. ω_c, α and Δt are the same defined in Paragraph 5.1.1.

$$q_{1,1} = \frac{4\Delta t\omega_c^2\alpha + \omega_c^2 + 4\Delta t\alpha^3 - 11\alpha^2}{4\alpha(\omega_c^2+\alpha^2)^3} - \frac{e^{-2\alpha\Delta t}}{4\omega_c^2\alpha(\omega_c^2+\alpha^2)^3}$$
$$\left[\omega_c\alpha\sin(2\omega_c\Delta t)\left(3\alpha^2-\omega_c^2\right)+\alpha^2\cos(2\omega_c\Delta t)\left(3\omega_c^2-\alpha^2\right)\right.$$
$$\omega_c^4+\alpha^4+2\omega_c^2\alpha^2+8\omega_c\alpha e^{\alpha\Delta t}\sin(\omega_c\Delta t)\left(\omega_c^2-\alpha^2\right)$$
$$\left.-16\omega_c^2\alpha^2 e^{\alpha\Delta t}\cos(\omega_c\Delta t)\right]$$

$$q_{1,2} = \frac{e^{-2\alpha\Delta t}}{2\omega_c^2(\omega_c^2+\alpha^2)^2}\left[\omega_c^2 e^{2\alpha\Delta t}+\cos(\omega_c\Delta t)^2\left(\omega_c^2-\alpha^2\right)\right.$$
$$+\alpha^2-2\omega_c^2 e^{\alpha\Delta t}\cos(\omega_c\Delta t)+\omega_c\alpha\sin(2\omega_c\Delta t)+$$
$$\left.-2\omega_c\alpha e^{\alpha\Delta t}\sin(\omega_c\Delta t)\right]$$

$$q_{1,3} = \frac{e^{-2\alpha\Delta t}}{4\omega_c^2\alpha(\omega_c^2+\alpha^2)}\left[\alpha^2\cos(2\omega_c\Delta t)-\omega_c^2 e^{2\alpha\Delta t}+\omega_c^2\right.$$
$$\left.-\alpha^2-\omega_c\alpha\sin(2\omega_c\Delta t)+4\omega_c\alpha e^{\alpha\Delta t}\sin(\omega_c\Delta t)\right]$$

$$q_{2,1} = \frac{e^{-2\alpha\Delta t}}{2\omega_c^2(\omega_c^2+\alpha^2)^2}\left[\omega_c^2 e^{2\alpha\Delta t}+\alpha^2+\omega_c^2\cos(\omega_c\Delta t)^2\right.$$
$$-\alpha^2\cos(\omega_c\Delta t)^2-2\omega_c^2 e^{\alpha\Delta t}\cos(\omega_c\Delta t)+\omega_c\alpha\sin(2\omega_c\Delta t)$$
$$\left.-2\omega_c\alpha e^{\alpha\Delta t}\sin(\omega_c\Delta t)\right]$$

$$q_{2,2} = \frac{e^{-2\alpha\Delta t}}{4\omega_c^2\alpha(\omega_c^2+\alpha^2)}\left[\omega_c^2-\omega_c^2 e^{2\alpha\Delta t}-\alpha^2\cos(2\omega_c\Delta t)\right.$$
$$\left.+\alpha^2+\omega_c\alpha\sin(2\omega_c\Delta t)\right]$$

$$q_{2,3} = \frac{e^{-2\alpha\Delta t}}{4\omega_c^2}\left[\cos(2\omega_c\Delta t)-1\right]$$

$$q_{3,1} = \frac{e^{-2\alpha\Delta t}}{4\omega_c^2\alpha(\omega_c^2+\alpha^2)}\left[\alpha^2\cos(2\omega_c\Delta t)-\omega_c^2 e^{2\alpha\Delta t}\right.$$
$$\left.+\omega_c^2-\alpha^2-\omega_c\alpha\sin(2\omega_c\Delta t)+4\omega_c\alpha e^{\alpha\Delta t}\sin(\omega_c\Delta t)\right]$$

$$q_{3,2} = q_{2,3}$$

$$q_{3,3} = \frac{e^{-2\alpha\Delta t}}{4\omega_c^2\alpha}\left[\alpha^2\cos(2\omega_c\Delta t)-\omega_c^2-\alpha^2+\right.$$
$$\left.+\omega_c\alpha\sin(2\omega_c\Delta t)\right]$$

Appendix B Jacobian Matrix

Applying the Jacobian matrix definition of Equation 17 to the measurement model defined in Equation 51 we get the following expressions for the $\mathbf{H_J}$ components, where symbols have been defined in Paragraph 5.2.2.

$$\frac{\partial\hat{\rho}_i}{\partial\xi} = -\frac{\xi_{s,i}-\xi}{R_i}, \quad \frac{\partial\hat{\rho}_i}{\partial\dot{\xi}} = \frac{\partial\hat{\rho}_i}{\partial\xi}\Delta t, \quad \frac{\partial\hat{\rho}_i}{\partial\ddot{\xi}} = \frac{1}{2}\frac{\partial\hat{\rho}_i}{\partial\xi}\Delta t^2$$

$$\frac{\partial\hat{\rho}_i}{\partial C_b} = 1 \qquad \frac{\partial\hat{\rho}_i}{\partial C_d} = 0$$

$$\frac{\partial\dot{\hat{\rho}}_i}{\partial\xi} = -\frac{\dot{\xi}_{s,i}-\dot{\xi}}{R_i} + \frac{\dot{R}_i\left(\xi_{s,i}-\xi\right)}{R_i^2}$$

$$\frac{\partial\dot{\hat{\rho}}_i}{\partial\dot{\xi}} = -\frac{\xi_{s,i}-\xi+\left(\dot{\xi}_{s,i}-\dot{\xi}\right)\Delta t}{R_i} + \frac{\dot{R}_i\left(\xi_{s,i}-\xi\right)\Delta t}{R_i^2}$$

$$\frac{\partial\dot{\hat{\rho}}_i}{\partial\ddot{\xi}} = -\frac{\left(\xi_{s,i}-\xi\right)\Delta t}{R_i} - \frac{1}{2}\frac{\Delta t^2}{R_i}\left(\dot{\xi}_{s,i}-\dot{\xi}+\frac{\dot{R}_i\left(\xi_{s,i}-\xi\right)}{R_i}\right)$$

$$\frac{\partial\dot{\hat{\rho}}_i}{\partial C_b} = 0, \qquad \frac{\partial\dot{\hat{\rho}}_i}{\partial C_d} = 1$$

References

1. Ramasamy, S. and Sabatini, R.: "A unified approach to cooperative and non-cooperative Sense-and-Avoid", Unmanned Aircraft Systems (ICUAS), 2015 International Conference on, Denver, CO, 2015, pp. 765-773. doi: 10.1109/ICUAS.2015.7152360
2. Ramasamy, S., Sabatini, R., Gardi, A.: Avionics sensor fusion for small size unmanned aircraft Sense-and-Avoid. Metrology for Aerospace (MetroAeroSpace), 2014 IEEE, Benevento, 2014, pp. 271-276. doi: 10.1109/MetroAeroSpace.2014.6865933
3. Williams, Ed.: 2Airborne collision avoidance system", In Proceedings of the 9th Australian workshop on Safety critical systems and software (SCS '04), Tony Cant (Ed.), 47. Australian Computer Society, Inc., Darlinghurst, Australia, 97-110, (2004)
4. Ramasamy, S., Sabatini, R. and Gardi, A.: "Novel flight management system for improved safety and sustainability in the CNS+A context," 2015 Integrated Communication, Navigation and Surveillance Conference (ICNS), Herdon, VA, 2015, pp. G3-1-G3-11. doi: 10.1109/ICNSURV.2015.7121225
5. Havangi, R.: "Robust SLAM: SLAM base on H_∞ square root unscented Kalman filter", Nonlinear Dynamics, Vol. 83.1, 767-779, (2016). doi: 10.1007/s11071-015-2365-x

6. Amidi, O., Kanade, T. and Fujita, K.: "A visual odometer for autonomous helicopter flight," Journal of Robotics and Autonomous Systems, Vol. 28, pp. 185-193, August, (1999)
7. Wu, A.D., Johnson, E.N. and Proctor, A.A.: "Vision-Aided inertial navigation for flight control", Journal of Aerospace Computing, Information, and Communication, Vol. 2, No. 9 (2005), pp. 348-360. doi: http://dx.doi.org/10.2514/1.16038
8. Jincheol, H. and Ramachandra, S.: "Vision-based obstacle avoidance based on monocular SLAM and image segmentation for UAVs", Infotech@Aerospace 2012, Infotech@Aerospace Conferences, Garden Grove, California, 1-9 (2012). doi: http://dx.doi.org/10.2514/6.2012-2464
9. Ramasamy, S., Gardi, A., Liu, J., Sabatini, R.: "A laser obstacle detection and avoidance system for manned and unmanned aircraft applications", Unmanned Aircraft Systems (ICUAS), 2015 International Conference on, Denver, CO, 2015, pp. 526-533. doi: 10.1109/ICUAS.2015.7152332
10. Sabatini, R., Gardi, A., Richardson, M.A.: LIDAR obstacle warning and avoidance system for unmanned aerial vehicle sense-and-avoid. Aerosp. Sci. Technol., 55, 344-358 (2016). doi: 10.1016/j.ast.2016.05.020
11. Cappello, F., Ramasamy, S., Sabatini, R. and Liu, J.: "Low-cost sensors based Multi-Sensor Data Fusion techniques for RPAS Navigation and Guidance", Unmanned Aircraft Systems (ICUAS), 2015 International Conference on, Denver, CO, 2015, pp. 714-722. doi: 10.1109/ICUAS.2015.7152354
12. Cappello, F, Sabatini, R., Ramasamy, S. and Marino, M.: "Particle filter based multi-sensor data fusion techniques for RPAS navigation and guidance," Metrology for Aerospace (MetroAeroSpace), 2015 IEEE, Benevento, 2015, pp. 395-400. doi: 10.1109/MetroAeroSpace.2015.7180689
13. Durrant-Whyte, Hugh, and Thomas C. Henderson: "Multisensor data fusion," Springer Handbook of Robotics. Springer Berlin Heidelberg, 585-610 (2008),
14. Fasano, G., et al. "Data fusion for UAS collision avoidance: results from flight testing." Proceedings of the AIAA Infotech at Aerospace Technical Conference, Hyatt Regency Saint Louis, 2011. doi: http://dx.doi.org/10.2514/6.2011-1458
15. Jamshaid, A., Mirza, U.B. and Rasheeq, M.: "Performance comparison among some nonlinear filters for a low cost SINS/GPS integrated solution", Nonlinear Dynamics, Vol. 61.3, 491-502 (2010). doi: 10.1007/s11071-010-9665-y
16. Dah-Jing, J., et al.: "Performance enhancement for ultra-tight GPS/INS integration using a fuzzy adaptive strong tracking unscented Kalman filter", Nonlinear Dynamics, Vol. 73.1, 377-395 (2013). doi: 10.1007/s11071-013-0793-z
17. Jaechan, L.: "Particle filtering for nonlinear dynamic state systems with unknown noise statistics", Nonlinear Dynamics, Vol 78.2, 1369-1388 (2014). doi:10.1007/s11071-014-1523-x
18. Haug, A.J.: Bayesian estimation and tracking: a practical guide. John Wiley & Sons, 2012
19. Bar-Shalom, Yaakov, X. Rong Li, and Thiagalingam Kirubarajan: Estimation with applications to tracking and navigation: theory algorithms and software. John Wiley & Sons, 2004
20. Brookner, Eli: "Tracking and Kalman Filtering Made Easy", John Wiley & Sons, 2002, doi: 10.1002/0471224197
21. Grewal, Mohinder S: "Kalman filtering," Springer Berlin Heidelberg, 2011
22. De Vivo, F., Battipede, M., Gili, P. & Brandl, A. "Ill-conditioned problems improvement adapting Joseph covariance formula to nonlinear Bayesian filters", WSEAS TRANSACTIONS ON ELECTRONICS, vol. 7, pp. 18-25, 2016, doi: 10.13140/RG.2.1.3027.0960
23. Särkkä, S.: Bayesian filtering and smoothing. Vol. 3. Cambridge University Press, 2013
24. Guoliang, L., Worgotter, F. and Markelic, I.: "Square-root sigma-point information filtering," IEEE Transactions on Automatic Control 57.11 (2012): 2945-2950. doi: 10.1109/TAC.2012.2193708
25. Bierman, G.J.: "Factorization methods for discrete sequential estimation," Courier Corporation, 2006
26. Ward, C. and Kincaid D.: "Numerical methods and computing," Pacific Grove: California (1985)
27. Grewal, Mohinder S., Weill, L. R., Andrews, A. P.: "Global Positioning Systems, Inertial Navigation, and Integration", John Wiley & Sons, Inc, Wiley-Interscience, 2001, doi: 10.1002/0470099720
28. Dunn, M. J.: "Navstar GPS Space Segment/Navigation User Segment L1C Interface Interface Specification IS-GPS-800", Technical report, Global Positioning System Directorate Systems Engineering and Integration, 2013
29. Macabiau, C., Moriella, L., Raimondi, M., Dupouy, C., Steingass, A. and Lehner, A.: "GNSS airborne multipath errors distribution using the high resolution aeronautical channel model and comparison to SARPs error curve", ION NTM 2006, National Technical Meeting of The Institute of Navigation, pp. 454, 2006
30. Niell, A. E.: "Global mapping functions for the atmosphere delay at radio wavelengths", Journal of Geophysical Research, 1996, doi: 10.1029/95JB03048
31. Curry, G. R.: "Radar System Performance Modeling, Second Edition", ARTECH HOUSE, INC., 2005, ISBN: 1-58053-816-9
32. Li, X. R. and Jilkov, V. P.: "Survey of maneuvering target tracking. Part I. Dynamic models", IEEE Transactions on aerospace and electronic systems, Vol. 39, N. 4, pp. 1333–1364, 2003, doi: 10.1109/TAES.2003.1261132
33. Tirri, A.E., Fasano, G., Accardo, D., Moccia, A., De Lellis, E., "Advanced sensing issues for UAS collision avoidance", In Proceedings of the 2nd International Conference on Application and Theory of Automation in Command and Control Systems (ATACCS '12). IRIT Press, Toulouse, France, pp. 12-19, ISBN: 978-2-917490-20-4
34. Salazar, L. R., Sabatini, R., Ramasamy, S., & Gardi, A.: "A Novel System for Non-Cooperative UAV Sense-And-Avoid" In proceedings of European Navigation Conference, April, 2013, doi: 10.13140/2.1.2164.4488
35. Noth, K. R.: "Modeling and simulation of a ground based sense and avoid architecture for Unmanned Aircraft System operations", In Integrated Communications, Navigation and Surveilance Conference (ICNS), IEEE 2011, pp. 7-1, doi: 10.1109/ICNSURV.2011.5935356
36. Spriesterbach, T.P., Bruns, K.A., Baron, L.I. and Sohlke, J.E.: Unmanned Aircraft System Airspace Integration in the National Airspace Using a Ground-Based Sense and Avoid System, Johns Hopkins APL Technical Digest, vol. 32, no. 3, 2013
37. Holmes, S. A., Klein, G., & Murray, D. W.: "An $O(N^2)$ Square Root Unscented Kalman Filter for Visual Simultaneous Localization and Mapping". IEEE transactions on pattern analysis and machine intelligence, 2009, vol. 31(7), pp. 1251-1263, doi: 10.1109/TPAMI.2008.189
38. J. Sanz Subirana, J.M., Zornoza, J. & Hernndez-Pajares M.: "Tropospheric Delay", Technical University of Catalonia, Spain, 2011, `http://www.navipedia.net/index.php/Tropospheric_Delay`