

Three-Dimensional Bearings-Only Target Tracking: Comparison of Few Sigma Point Kalman Filters



U. Asfia, R. Radhakrishnan, and S. N. Sharma

Abstract Three-dimensional bearings-only target tracking problem is considered in this work. To solve this problem, highly accurate nonlinear filtering algorithms are required. This is because the nonlinearities in the noise corrupted measurements, combined with the other uncertainties, make the estimation problem highly challenging. In this work, the extended Kalman filter (EKF), the cubature Kalman filter (CKF), the unscented Kalman filter (UKF), and the new sigma point Kalman filter (NSKF) are used to solve this state estimation problem. The performance criteria chosen are root mean square error (RMSE) in position and velocity. To have a more meaningful study, RMSE at the end of the observation period and time-averaged RMSE after the observer manoeuvre is calculated, for varying measurement noise and initial uncertainty.

1 Introduction

The bearings-only tracking (BOT) problem has been a popular area of research for a sufficient period of time. In particular, it is a nonlinear filtering problem with a goal to estimate the target states (positions and velocities) using only bearing measurements that are corrupted with noise [1]. These measurements are obtained by a moving platform, also called as observer, with the help of one or more sensors, whereas the target can be a ship, an aircraft, or any mobile platform. The problem of target tracking, which considers only bearing angles as measurements, is sometimes referred to as bearings-only target motion analysis (TMA). The nonlinearities in the measurement equation and the associated noises only make the problem even

U. Asfia (✉) · R. Radhakrishnan · S. N. Sharma
Department of Electrical Engineering, SV National Institute of Technology Surat, Surat, India
e-mail: ds19e1006@eed.svnit.ac.in

R. Radhakrishnan
e-mail: r.rahul@eed.svnit.ac.in

S. N. Sharma
e-mail: sns@eed.svnit.ac.in

more challenging. Another significant factor is the observability. This problem is not observable, or the filtering solution does not converge until and unless the observer makes a manoeuvre [2].

Significant research has been conducted for BOT problem formulated in two-dimensional space [3, 4]. Since the measurement is nonlinear, the assumption that prior and posterior probability densities follow Gaussian distribution may not be true. Hence, for 2D BOT problems, nonlinear filters with both Gaussian as well as non-Gaussian assumption have been implemented for the solution. For nonlinear systems, the most widely used estimator is the extended Kalman filter (EKF) [5]. The EKF uses the traditional Kalman filter equations and applies it to the nonlinear system by linearising the nonlinear model [6]. In doing so, EKF proves to be computationally efficient over other advanced Kalman filters. Later, various deterministic sampling point filters such as the unscented Kalman filter (UKF) [7], cubature Kalman filter (CKF) [8], cubature quadrature Kalman filter (CQKF) [9], Gauss–Hermite filter (GHF) [10], etc., have been implemented. All these filters assume Gaussianity for the prior and posterior density functions. Filtering accuracy and their performance comparison are widely reported in the literature [11].

Now coming to the filters that follow non-Gaussian assumption, we have the particle filter (PF) [12] and algorithms that make use of Gaussian sum approach [13]. Most of the nonlinear filters mentioned above were implemented in the Gaussian sum approach and evaluated for solving BOT problems [11]. In the majority of scenarios described for the study of 2D BOT problems, it was observed that there was significant track-loss associated with the estimator performance. Hence, a track-loss condition was formulated and implemented so that the most accurate estimated values were considered for performance analysis. It was noted that without this condition, certain filters such as EKF diverged or performed with very less accuracy [4]. All these works focused on tracking a target that is assumed to be following a constant velocity motion. There are also numerous works available for tracking a manoeuvring target with bearings-only measurements, in 2D [4].

A pseudolinear estimator for 2D case was introduced [14] to deal with the nonlinearity and convergence problem directly. This approach reformulates the nonlinear bearing measurement into a linear form. But the disadvantage was that the correlation between the new form and the bearing noise leads to bias problems [15, 16]. To overcome this bias problem, some attempts were made, which resulted in the proposal of iterative instrumental variable estimator (IV) [17], and also a bias compensated IV [18]. Now, these were all recursive algorithms. With respect to IV, a batch iterative algorithm was proposed, termed as modified IV estimator [19]. Hence, from this discussion, it is obvious that BOT problem in 2D scenarios is widely studied and addressed.

However, there are relatively less number of works which addresses the BOT problem in 3D geometry [20–23]. Recently, much focus is on this problem, as researchers have decided to not to ignore the depth, or the height between the target and observer. So, the basic difference here is the addition of one more measurement, the elevation angle. Some of the nonlinear filters which were used for solving BOT problem have already been applied for 3D geometry [24]. In [25], in addition to the prob-

lem formulation in cartesian coordinates, tracking algorithms were developed for modified spherical coordinates too. Robust tracking algorithms were also developed for 3D BOT problems [26]. Similar to the PLE and IV estimators developed for 2D geometry, the same concepts with certain modifications were applied to problem formulation in 3D geometry [27].

Here, the authors focus their work on solving 3D BOT problem formulated in Cartesian coordinate system. A comparative study on the performance evaluation of certain sigma point Kalman filters was done. NSKF [28] was a recent addition to the sigma point Kalman filter family, which claims more accuracy than the conventional UKF. Rather than assigning same weights to all the sigma points, NSKF assigns more weights to the sigma points that are nearer to the mean value. The inherent disadvantage in this formulation is the increase in the number of sigma points.

This paper deals with a 3D bearings-only target tracking scenario for which the performance of various filters like EKF, UKF, CKF, and NSKF is studied. The filtering accuracy is analysed using RMSE of resultant position and velocity. To further analyse the accuracy, the effect of initial condition and varying measurement noise is taken into account. The rest of the paper is organised as follows: Sect. 2 formulates the 3D BOT problem in Cartesian coordinates, and the filtering algorithms are mentioned in Sect. 3. The simulation results are discussed in Sect. 4 followed by concluding remarks in Sect. 5.

2 Problem Formulation

The purpose of the 3D bearings-only tracking problem studied in this paper is to accurately estimate the position and velocity of the target from bearing (β) and elevation (ϵ) measurements obtained by an observer at fixed-time intervals. The target motion is assumed to be a constant velocity motion during the observation period. We assume that the observer's motion is deterministic, with known position and velocity. Figure 1 is a Cartesian coordinate frame representing the position of the target and observer.

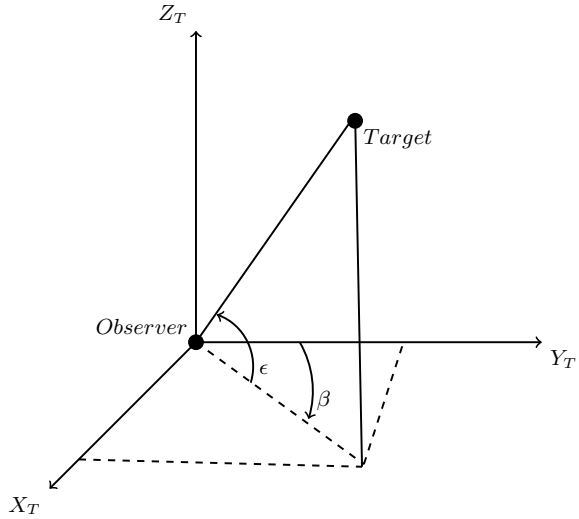
Let the target and observer state vector be defined as

$$\begin{aligned} X_k^t &= [x_k^t \ y_k^t \ z_k^t \ \dot{x}_k^t \ \dot{y}_k^t \ \dot{z}_k^t]^T \\ X_k^o &= [x_k^o \ y_k^o \ z_k^o \ \dot{x}_k^o \ \dot{y}_k^o \ \dot{z}_k^o]^T. \end{aligned}$$

Now, a relative state vector is defined as

$$\begin{aligned} \mathcal{X}_k &= X_k^t - X_k^o \\ &= [x_k^t - x_k^o \ y_k^t - y_k^o \ z_k^t - z_k^o \ \dot{x}_k^t - \dot{x}_k^o \ \dot{y}_k^t - \dot{y}_k^o \ \dot{z}_k^t - \dot{z}_k^o]^T \\ &= [x_k \ y_k \ z_k \ \dot{x}_k \ \dot{y}_k \ \dot{z}_k]^T. \end{aligned} \quad (1)$$

Fig. 1 Target–observer in Cartesian coordinate frame



Let the range vector be r and is defined as

$$\begin{aligned}
 r &= [x_k^t - x_k^o \quad y_k^t - y_k^o \quad z_k^t - z_k^o]^T \\
 &= [x_k \quad y_k \quad z_k]^T.
 \end{aligned}
 \tag{2}$$

Now, the actual range or the slant range at any instant is defined as $r_k = \sqrt{x_k^2 + y_k^2 + z_k^2}$. From the target–observer geometry shown in Fig. 1, r can be expressed in terms of β and ϵ as

$$r = r \begin{bmatrix} \cos \epsilon \sin \beta \\ \cos \epsilon \cos \beta \\ \sin \epsilon \end{bmatrix}.$$

Here, $\beta \in [0, 2\pi]$ and $\epsilon \in [-\frac{\pi}{2}, \frac{\pi}{2}]$. Moreover, the ground range or the range in x - y plane is defined as $r_g = r \cos \epsilon$.

2.1 Process Model

The discrete time state equation describing the constant velocity target dynamics can be expressed as

$$X_k^t = F_{k-1} X_{k-1}^t + w_{k-1}^t,
 \tag{3}$$

where F_{k-1} is the state transition matrix defined as

$$F_{k-1} = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

T is the sampling interval, and w_{k-1}^t is a zero mean Gaussian noise, uncorrelated with X .

Now, the relative state vector dynamics using (3) and (1) is defined as [24]

$$\mathcal{X}_k = F_{k-1}\mathcal{X}_{k-1} + w_{k-1}^t - X_k^o + F_{k-1}X_{k-1}^o, \tag{4}$$

where $w_{k-1}^t \sim N(0, Q_{k-1})$ and

$$Q_{k-1} = \begin{bmatrix} \frac{T^3}{3}q_x & 0 & 0 & \frac{T^2}{2}q_x & 0 & 0 \\ 0 & \frac{T^3}{3}q_y & 0 & 0 & \frac{T^2}{2}q_y & 0 \\ 0 & 0 & \frac{T^3}{3}q_z & 0 & 0 & \frac{T^2}{2}q_z \\ \frac{T^2}{2}q_x & 0 & 0 & Tq_x & 0 & 0 \\ 0 & \frac{T^2}{2}q_y & 0 & 0 & Tq_y & 0 \\ 0 & 0 & \frac{T^2}{2}q_z & 0 & 0 & Tq_z \end{bmatrix}.$$

Here, q_x, q_y, q_z are the power spectral densities of the process noise along the $X, Y,$ and Z axes, respectively. Figure 2 shows the target–observer dynamics considered in this study. It can be seen that the target follows a constant velocity model and the observer manoeuvre is modelled in different constant velocity phases.

2.2 Measurement Model

The measurement model involving the bearing and elevation angle is defined as

$$z_k = h(\mathcal{X}_k) + v_k, \tag{5}$$

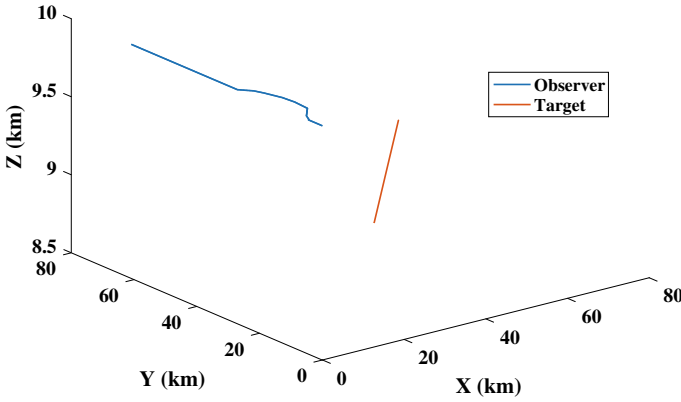


Fig. 2 3D BOT scenario

where

$$h(\mathcal{X}_k) = \begin{bmatrix} \beta_k \\ \epsilon_k \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\frac{x_k}{y_k}\right) \\ \tan^{-1}\left(\frac{z_k}{\sqrt{x_k^2 + y_k^2}}\right) \end{bmatrix}.$$

Here, $\mathbf{v}_k = [\mathbf{v}_{1,k} \ \mathbf{v}_{2,k}]^T$ is a zero mean white Gaussian noise with covariance matrix R_k , that is $\mathbf{v}_k \sim N(0, R_k)$, $R_k = \text{diag}(\sigma_\beta^2, \sigma_\epsilon^2)$. Here, σ_β and σ_ϵ are the standard deviations of error in bearing and elevation angles, respectively.

3 Recursive Bayesian Algorithms

Even though certain batch estimation algorithms have been proposed recently for 3D BOT problem [27], we resort to the implementation of three recursive deterministic sampling point filters. They are the cubature Kalman filter (CKF), unscented Kalman filter (UKF), and the recently proposed new sigma point Kalman filter (NSKF). In addition to this, the extended Kalman filter is also implemented. In this section, Bayesian filtering, the formulation of these filtering algorithms, and their implementation with respect to the problem considered are discussed.

3.1 Bayesian Approach to Filtering

Suppose the discrete state-space process and measurement models are represented as in (4) and (5). In Bayesian filtering, the unknown states and parameters to be estimated are assumed as random variables that follow a particular probability distribution. This

probability distribution is termed as prior probability density. The prior information regarding the states to be estimated is combined with the measurements received, via a conditional density function of the states, given the measurements. This probability density function (pdf) is usually termed as posterior probability density function.

In this framework, the two steps involved in calculating the prior and posterior pdfs are the time update step and the measurement update step.

Time update step: The prior pdf, $p(\mathcal{X}_k | z_{1:k-1})$, is obtained using the Chapman–Kolmogorov equation

$$p(\mathcal{X}_k | z_{1:k-1}) = \int p(\mathcal{X}_k | \mathcal{X}_{k-1})p(\mathcal{X}_{k-1} | z_{1:k-1})d\mathcal{X}_{k-1}. \tag{6}$$

Measurement update step: The measurement at time instant k , z_k and the prior pdf is used to obtain the posterior pdf using Bayes’ rule

$$p(\mathcal{X}_k | z_{1:k}) = \frac{p(z_k | \mathcal{X}_k)p(\mathcal{X}_k | z_{1:k-1})}{\int p(z_k | \mathcal{X}_k)p(\mathcal{X}_k | z_{1:k-1})d\mathcal{X}_k}. \tag{7}$$

When the process or measurement model becomes nonlinear, the integrals become intractable, and we need to look for suboptimal solutions. This is due to the fact that prior and posterior pdfs may not be Gaussian anymore. To overcome this, one approach is to approximate them with Gaussian distribution and take the mean as a point estimate. Under this assumption, the prior and posterior pdfs can be expressed as $p(\mathcal{X}_k | z_{1:k-1}) = N(\hat{\mathcal{X}}_{k|k-1}, P_{k|k-1})$ and $p(\mathcal{X}_k | z_{1:k}) = N(\hat{\mathcal{X}}_{k|k}, P_{k|k})$. The posterior mean $\hat{\mathcal{X}}_{k|k}$ is the estimated value of state \mathcal{X} .

3.2 Extended Kalman Filter

The extended Kalman filter is an immediate extension of the Kalman filter to nonlinear state estimation problems. The nonlinear process and measurement models are linearised, and then the Kalman filter equations are used. In this problem, since the process model is linear, the same Kalman filter equations for the time update step can be implemented. But for the nonlinear measurement model, a corresponding linearised model has to be computed in the measurement update step. With a suitable assumption of the posterior estimate $\hat{\mathcal{X}}_{k-1|k-1}$ and covariance matrix $P_{k-1|k-1}$ at time $k - 1$, the recursive algorithm can be given as

$$\begin{aligned} \hat{\mathcal{X}}_{k|k-1} &= F_{k-1}\hat{\mathcal{X}}_{k-1|k-1} - X_k^o + F_{k-1}X_{k-1}^o \\ \bar{P}_{k|k-1} &= F_{k-1}P_{k-1|k-1}F_{k-1}^T + Q \end{aligned}$$

Now, the measurement update step involves the formation of Jacobian matrix followed by innovation covariance and Kalman gain, given as

$$\begin{aligned} H_k &= \frac{\partial h(\hat{\mathcal{X}}_{k|k-1})}{\partial \hat{\mathcal{X}}_{k|k-1}} \\ S_k &= H_k \bar{P}_{k|k-1} H_k^T + R_k \\ K_k &= \bar{P}_{k|k-1} H_k^T S_k^{-1} \end{aligned}$$

Then, the posterior mean and covariance are found out as

$$\begin{aligned} \hat{\mathcal{X}}_{k|k} &= \hat{\mathcal{X}}_{k|k-1} + K_k [z_k - h(\hat{\mathcal{X}}_{k|k-1})] \\ P_{k|k} &= \bar{P}_{k|k-1} - K_k S_k K_k^T \end{aligned}$$

3.3 Unscented Kalman filter

Unscented Kalman filter (UKF)[7] was the first suboptimal filter that introduced the concept of numerical integration of the intractable integrals using a set of deterministic points and weights. The major advantage over the already existing EKF was that the algorithm was derivative free. It proved to be more accurate and less sensitive to initial condition mismatch when compared with the EKF. The main concept was to implement the numerical approximation using the unscented transformation.

Suppose that the integral to be approximated is of the form

$$\mathbb{I}(\mathcal{X}) = \int h(\mathcal{X}) p_{\mathcal{X}}(\mathcal{X}) d\mathcal{X}$$

where \mathcal{X} is a random variable that is assumed to follow a Gaussian density $p_{\mathcal{X}}(\mathcal{X}) \sim N(\hat{\mathcal{X}}, P)$. Now, since $\mathbb{I}(\mathcal{X})$ is not a tractable one, a numerical approximation with the help of unscented transformation is defined as

$$\mathbb{I}(\mathcal{X}) \cong \sum_{i=1}^{2n+1} w_i h(\hat{\mathcal{X}}_i), \quad \text{where } n \text{ is the order of the system.}$$

Here

$$\begin{aligned} \hat{\mathcal{X}}_1 &= \hat{\mathcal{X}} \\ \hat{\mathcal{X}}_i &= \hat{\mathcal{X}} + (\sqrt{(n+\kappa)P})_i, \quad i = 1, \dots, n \\ \hat{\mathcal{X}}_i &= \hat{\mathcal{X}} - (\sqrt{(n+\kappa)P})_i, \quad i = 1, \dots, n \quad \text{and} \\ w_1 &= \frac{\kappa}{n+\kappa}, \quad w_i = \frac{1}{2(n+\kappa)}, \quad \text{for } i = 1, \dots, 2n. \end{aligned} \tag{8}$$

3.4 New Sigma Point Kalman Filter

In UKF, the sigma point equal to mean is given the maximum weight, whereas all other sigma points are assigned equal weights. In [28], this weight assignment was revisited because of the fact that the sigma points which are nearer to the mean value should have higher probability of occurrence. Accordingly, a new sigma point transformation was formulated where sigma points closer to the mean were given more weights when compared to the sigma points that are further away from mean. This new transformation also took care of the criteria that first two moments are matched exactly. Taking the same assumption above, $\mathcal{X} \sim N(\hat{\mathbf{x}}, P)$, the new transformation is defined as

$$\begin{aligned} \hat{\mathcal{X}}_1 &= \hat{\mathbf{x}}, \quad w_1 = 1 - \frac{\sum_{i=1}^n \alpha_i}{2(\sum_{i=1}^n \alpha_i + \mathfrak{b})} \\ \hat{\mathcal{X}}_{i+1} &= \hat{\mathbf{x}} + \sqrt{\frac{\sum_{i=1}^n \alpha_i + \mathfrak{b}}{m\alpha_i}} S_i, \quad w_{i+1} = \frac{m\alpha_i}{4(\sum_{i=1}^n \alpha_i + \mathfrak{b})}, \quad i = 1, \dots, n \\ \hat{\mathcal{X}}_{i+1} &= \hat{\mathbf{x}} - \sqrt{\frac{\sum_{i=1}^n \alpha_i + \mathfrak{b}}{m\alpha_{i-n}}} S_{i-n}, \quad w_{i+1} = \frac{m\alpha_{i-n}}{4(\sum_{i=1}^n \alpha_i + \mathfrak{b})}, \quad i = n + 1, \dots, 2n \\ \hat{\mathcal{X}}_{i+1} &= \hat{\mathbf{x}} + \sqrt{\frac{\sum_{i=1}^n \alpha_i + \mathfrak{b}}{(1-m)\alpha_{i-2n}}} S_{i-2n}, \quad w_{i+1} = \frac{(1-m)\alpha_{i-2n}}{4(\sum_{i=1}^n \alpha_i + \mathfrak{b})}, \quad i = 2n + 1, \dots, 3n \\ \hat{\mathcal{X}}_{i+1} &= \hat{\mathbf{x}} - \sqrt{\frac{\sum_{i=1}^n \alpha_i + \mathfrak{b}}{(1-m)\alpha_{i-3n}}} S_{i-3n}, \quad w_{i+1} = \frac{(1-m)\alpha_{i-3n}}{4(\sum_{i=1}^n \alpha_i + \mathfrak{b})}, \quad i = 3n + 1, \dots, 4n. \end{aligned} \tag{9}$$

Here, S is defined such that $SS^T = P$, and S_i and P_i denote the i^{th} column of S and P , respectively. The variable α is defined as $\alpha_i = \frac{|\langle X_i, P_i \rangle|}{\|X_i\|_2 \|P_i\|_2}$. \mathfrak{b} is a real constant such that $\mathfrak{b} > \{\frac{1}{4} \max(m\alpha_i) - \frac{1}{2} \sum_{i=1}^n \alpha_i\}$, and m is chosen as $m \in (0.5, 1)$.

If we represent the total number of sigma points with \mathcal{N} , then from this transformation, the value of \mathcal{N} becomes $4n + 1$. But in case of the UKF, value of $\mathcal{N} = 2n + 1$. Compared to the UKF, NSKF has an advantage of having two tuning parameters, \mathfrak{b} and m . The algorithm for implementing UKF and NSKF for the 3D BOT problem is given below.

4 Simulation Results

The simulation and results comprise a performance comparison of EKF, UKF, CKF, and NSKF for 1000 Monte Carlo runs. The entire tracking scenario is implemented and simulated in MATLAB software. The initial parameter values required for gen-

Algorithm 1: For UKF and NSKF

Initialise $\hat{\mathcal{X}}_{k-1|k-1}$ and $P_{k-1|k-1}$

$$\hat{\mathcal{X}}_{k|k-1} = F_{k-1}\hat{\mathcal{X}}_{k-1|k-1} - X_k^o + F_{k-1}X_{k-1}^o$$

$$\bar{P}_{k|k-1} = F_{k-1}P_{k-1|k-1}F_{k-1}^T + Q$$

Calculate $\hat{\mathcal{X}}_i$ using (8) or (9), $i = 1, \dots, \mathcal{N}$

$$Z_{i,k|k-1} = h(\hat{\mathcal{X}}_i)$$

$$\hat{z}_k = \sum_{i=1}^{\mathcal{N}} w_i Z_{i,k|k-1}$$

$$P_{z_k} = \sum_{i=1}^{\mathcal{N}} w_i [Z_{i,k|k-1} - \hat{z}_k][Z_{i,k|k-1} - \hat{z}_k]^T + R_k$$

$$P_{\mathcal{X}_k z_k} = \sum_{i=1}^{\mathcal{N}} w_i [\hat{\mathcal{X}}_i - \hat{\mathcal{X}}_{k|k-1}][Z_{i,k|k-1} - \hat{z}_k]^T$$

$$K_k = P_{\mathcal{X}_k z_k} P_{z_k}^{-1}$$

$$\hat{\mathcal{X}}_{k|k} = \hat{\mathcal{X}}_{k|k-1} + K_k [z_k - \hat{z}_k]$$

$$P_{k|k} = \bar{P}_{k|k-1} - K_k P_{z_k} K_k^T$$

Table 1 Target and observer initial parameters

Parameters	Value
Initial target position	$\begin{bmatrix} 138/\sqrt{2} & 138/\sqrt{2} & 9 \end{bmatrix}$ (km)
Initial target speed (s)	0.297 (km/s)
Target course	-135 ($^\circ$)
Elevation angle	0.415 ($^\circ$)
Initial observer position	$\begin{bmatrix} 0 & 0 & 10 \end{bmatrix}$ (km)
Initial observer speed (s)	0.297 (km/s)

erating the scenario shown in Fig. 2 are given in Table 1. Here, target and observer initial position in the three coordinates are given. The resultant velocity of the target and observer is given, which is used for finding the velocities in the three coordinate axes. From Fig. 2, it should be noted that target position in Z position is decreasing. The bearing angle β is calculated with reference to the Y -axis.

For each Monte Carlo run, the target is assumed to start at a position defined by the initial measurement received. According to this measurement, a suitable assumption for range r , target speed s , bearing, and elevation angle is considered. Here, these values are mentioned in Table 2. The variables in the process noise, defined for calculating process noise covariance matrix Q , are taken as $q_x = q_y = 0.01 \text{ m}^2/\text{s}^3$ and $q_z = 10^{-4} \text{ m}^2/\text{s}^3$. The motion of the observer is deterministic and moves in a plane parallel to the X - Y plane at a fixed height of 10 km.

Table 2 Parameters for defining $\hat{\mathcal{X}}_{0|0}$ & $P_{0|0}$

Parameter	Mean	Standard deviation
Range (km)	150	13.6
Speed (m/s)	258	41.6
Bearing heading (rad/s)	$\beta + \pi$	$\pi/\sqrt{12}$
Elevation heading (rad/s)	0	$\pi/60$

4.1 Filter Initialisation

All the filters in this paper are initialised using the method given in [24] with the parameters shown in Table 2. The relative state is initialised depending on the first measurement of bearing and elevation angle and the prior range estimate of the target. The $\bar{r} \sim N(r, \sigma_r^2)$ is the prior range estimate of the target, with r as the assumed true initial range of the target from the observer. The true initial bearing and elevation measurement estimate are $\hat{\beta}_1$ and $\hat{\epsilon}_1$ with headings $\bar{\alpha}_1$ and $\bar{\gamma}_1$, respectively. Also, $\bar{s} \sim N(s, \sigma_s^2)$ is the initial speed estimate for the target.

The initial estimate ($\hat{\mathcal{X}}_{0|0}$) for relative state vector is given as [24]

$$\hat{\mathcal{X}}_{0|0} = \begin{bmatrix} \bar{r} \eta_{1,0}(\hat{\epsilon}_1, \sigma_\epsilon^2) \eta_{0,1}(\hat{\beta}_1, \sigma_\beta^2) \\ \bar{r} \eta_{1,0}(\hat{\epsilon}_1, \sigma_\epsilon^2) \eta_{1,1}(\hat{\beta}_1, \sigma_\beta^2) \\ \bar{r} \eta_{0,1}(\hat{\epsilon}_1, \sigma_\epsilon^2) \\ \bar{s} \eta_{1,0}(\bar{\gamma}_1, \sigma_\gamma^2) \eta_{0,1}(\bar{\alpha}_1, \sigma_\alpha^2) - \dot{x}_1^0 \\ \bar{s} \eta_{1,0}(\bar{\gamma}_1, \sigma_\gamma^2) \eta_{1,1}(\bar{\alpha}_1, \sigma_\alpha^2) - \dot{y}_1^0 \\ \bar{s} \eta_{0,1}(\bar{\gamma}_1, \sigma_\gamma^2) - \dot{z}_1^0 \end{bmatrix},$$

where $\eta_{1,0}(\mu, \sigma^2) = \cos \mu \exp(-\sigma^2/2)$ and $\eta_{0,1}(\mu, \sigma^2) = \sin \mu \exp(-\sigma^2/2)$. Also, $(\dot{x}_1^0, \dot{y}_1^0, \dot{z}_1^0)$ is the initial velocity components of the observer state vector.

Let P be the covariance matrix, whose (i, j) th element is represented as $P_{i,j}$. The upper triangular nonzero elements of the covariance matrix P are [24]

$$\begin{aligned} P_{11} &= (\sigma_r^2 + \bar{r}^2) \eta_{2,0}(\hat{\epsilon}_1, \sigma_\epsilon^2) \eta_{0,2}(\hat{\beta}_1, \sigma_\beta^2) - \bar{r}^2 \eta_{1,0}^2(\hat{\epsilon}_1, \sigma_\epsilon^2) \eta_{0,1}^2(\hat{\beta}_1, \sigma_\beta^2) \\ P_{12} &= (\sigma_r^2 + \bar{r}^2) \eta_{2,0}(\hat{\epsilon}_1, \sigma_\epsilon^2) \eta_{1,1}(\hat{\beta}_1, \sigma_\beta^2) - \bar{r}^2 \eta_{1,0}^2(\hat{\epsilon}_1, \sigma_\epsilon^2) \eta_{1,0}(\hat{\beta}_1, \sigma_\beta^2) \eta_{0,1}(\hat{\beta}_1, \sigma_\beta^2) \\ P_{13} &= [(\sigma_r^2 + \bar{r}^2) \eta_{1,1}(\hat{\epsilon}_1, \sigma_\epsilon^2) - \bar{r}^2 \eta_{1,0}(\hat{\epsilon}_1, \sigma_\epsilon^2) \eta_{0,1}(\hat{\epsilon}_1, \sigma_\epsilon^2)] \eta_{0,1}(\hat{\beta}_1, \sigma_\beta^2) \\ P_{22} &= (\sigma_r^2 + \bar{r}^2) \eta_{2,0}(\hat{\epsilon}_1, \sigma_\epsilon^2) \eta_{2,0}(\hat{\beta}_1, \sigma_\beta^2) - \bar{r}^2 \eta_{1,0}^2(\hat{\epsilon}_1, \sigma_\epsilon^2) \eta_{1,0}^2(\hat{\beta}_1, \sigma_\beta^2) \\ P_{23} &= [(\sigma_r^2 + \bar{r}^2) \eta_{1,1}(\hat{\epsilon}_1, \sigma_\epsilon^2) - \bar{r}^2 \eta_{1,0}(\hat{\epsilon}_1, \sigma_\epsilon^2) \eta_{0,1}(\hat{\epsilon}_1, \sigma_\epsilon^2)] \eta_{1,0}(\hat{\beta}_1, \sigma_\beta^2) \\ P_{33} &= (\sigma_r^2 + \bar{r}^2) \eta_{0,2}(\hat{\epsilon}_1, \sigma_\epsilon^2) - \bar{r}^2 \eta_{0,1}(\hat{\epsilon}_1, \sigma_\epsilon^2) \\ P_{44} &= (\sigma_s^2 + \bar{s}^2) \eta_{2,0}(\bar{\gamma}_1, \sigma_\gamma^2) \eta_{0,2}(\bar{\alpha}_1, \sigma_\alpha^2) - \bar{s}^2 \eta_{1,0}^2(\bar{\gamma}_1, \sigma_\gamma^2) \eta_{0,1}^2(\bar{\alpha}_1, \sigma_\alpha^2) \end{aligned}$$

$$\begin{aligned}
 P_{45} &= (\sigma_s^2 + \bar{s}^2) \eta_{2,0}(\bar{\gamma}_1, \sigma_\gamma^2) \eta_{1,1}(\bar{\alpha}_1, \sigma_\alpha^2) - \bar{s}^2 \eta_{1,0}^2(\bar{\gamma}_1, \sigma_\gamma^2) \eta_{1,0}(\bar{\alpha}_1, \sigma_\alpha^2) \eta_{0,1}(\bar{\alpha}_1, \sigma_\alpha^2) \\
 P_{46} &= [(\sigma_s^2 + \bar{s}^2) \eta_{1,1}(\bar{\gamma}_1, \sigma_\gamma^2) - \bar{s}^2 \eta_{1,0}(\bar{\gamma}_1, \sigma_\gamma^2) \eta_{0,1}(\bar{\gamma}_1, \sigma_\gamma^2)] \eta_{0,1}(\bar{\alpha}_1, \sigma_\alpha^2) \\
 P_{55} &= (\sigma_s^2 + \bar{s}^2) \eta_{2,0}(\bar{\gamma}_1, \sigma_\gamma^2) \eta_{2,0}(\bar{\alpha}_1, \sigma_\alpha^2) - \bar{s}^2 \eta_{1,0}^2(\bar{\gamma}_1, \sigma_\gamma^2) \eta_{1,0}^2(\bar{\alpha}_1, \sigma_\alpha^2) \\
 P_{56} &= [(\sigma_s^2 + \bar{s}^2) \eta_{1,1}(\bar{\gamma}_1, \sigma_\gamma^2) - \bar{s}^2 \eta_{1,0}(\bar{\gamma}_1, \sigma_\gamma^2) \eta_{0,1}(\bar{\gamma}_1, \sigma_\gamma^2)] \eta_{1,0}(\bar{\alpha}_1, \sigma_\alpha^2) \\
 P_{66} &= (\sigma_s^2 + \bar{s}^2) \eta_{0,2}(\bar{\gamma}_1, \sigma_\gamma^2) - \bar{s}^2 \eta_{0,1}^2(\bar{\gamma}_1, \sigma_\gamma^2),
 \end{aligned}$$

where

$$\begin{aligned}
 \eta_{2,0}(\mu, \sigma^2) &= [\sinh \sigma^2 + \cos^2 \mu \exp(-\sigma^2)] \exp(-\sigma^2) \\
 \eta_{0,2}(\mu, \sigma^2) &= [\sinh \sigma^2 + \sin^2 \mu \exp(-\sigma^2)] \exp(-\sigma^2) \\
 \eta_{1,1}(\mu, \sigma^2) &= \cos \mu \sin \mu \exp(-2\sigma^2).
 \end{aligned}$$

Since P is a symmetric matrix, the property that $P_{ij} = P_{ji}$ gives the lower triangular elements of the covariance matrix. Hence, $P_{0|0}$ is defined as

$$P_{0|0} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & 0 & 0 & 0 \\ P_{12} & P_{22} & P_{23} & 0 & 0 & 0 \\ P_{13} & P_{32} & P_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & P_{44} & P_{45} & P_{46} \\ 0 & 0 & 0 & P_{45} & P_{55} & P_{56} \\ 0 & 0 & 0 & P_{64} & P_{56} & P_{66} \end{bmatrix}.$$

4.2 Performance Criteria

Two measures of performance using root mean square error (RMSE) are used to compare the various filters. RMSE was calculated for the target states, which were found out from the estimated relative states and from the deterministic observer values. We found out the RMSE for both resultant position as well as for the resultant velocity. The first measure defined was the RMSE for position and velocity averaged from the end of the last observer manoeuvre to the end of observation or simulation period. The final RMSE for position and velocity at the end of the simulation period was considered to be the second measure of performance. These statistics are computed by averaging over 1000 realisations, for bearing and elevation measurement noise standard deviations of 0.057° , 0.65° , 0.8° , and 1° . The effect of increase in initial uncertainty on filter accuracy is also studied. The expression for RMSE in position is given as

$$\text{RMSE}_k = \sqrt{\frac{1}{M} \sum_{j=1}^M [(x_{j,k}^t - \hat{x}_{j,k}^t)^2 + (y_{j,k}^t - \hat{y}_{j,k}^t)^2 + (z_{j,k}^t - \hat{z}_{j,k}^t)^2]}$$

where k and M denote the total number of time steps and Monte Carlo runs, respectively. A similar expression for RMSE in velocity was implemented for the study. As compared to the 2D BOT scenarios widely available in the literature, the 3D scenario considered in this study did not incur any track-loss. Hence, the need for implementing any track-loss condition for performance analysis is ruled out.

4.3 Performance Comparison

For simulation, the sampling time is considered as $T = 1$ s, and the total observation period lasted for $T = 210$ s. The tracking performance of NSKF is shown in Fig. 3 for a single run. The estimated target path is plotted along with the truth target path and the observer path in this figure. It is observed that the NSKF accurately tracks the target.

RMSE in position and velocity is plotted in Figs. 4 and 5 to compare the estimation accuracy of various filters. From these figures, it can be observed that all filters performed with comparable accuracy. The results are shown in Tables 3 and 4. For the particular target–observer scenario discussed in the paper, it can be seen that all the filters give comparable results. The effect of increase in the measurement noise can be seen in the increase of the RMSE value at the final time for both position and velocity resulting in lower filter accuracy. From Table 3, when the measurement noise is low (i.e. standard deviation is 0.057°), EKF gives comparable response. For higher measurement noise standard deviations (i.e. 1°), it can be seen that NSKF

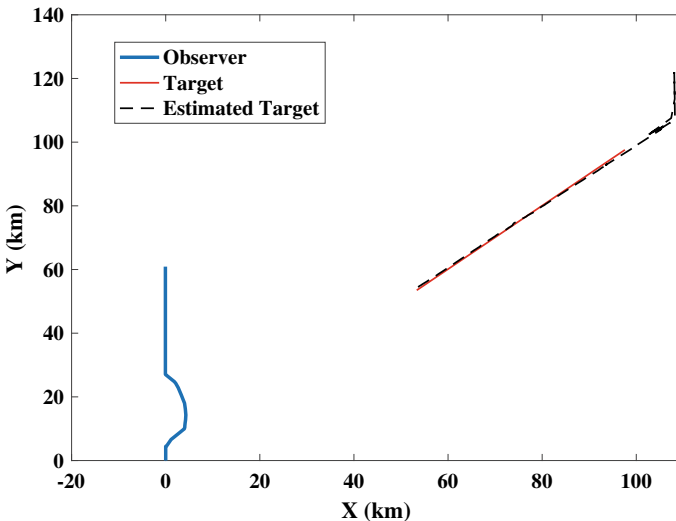


Fig. 3 Estimated and truth target path in X-Y plane

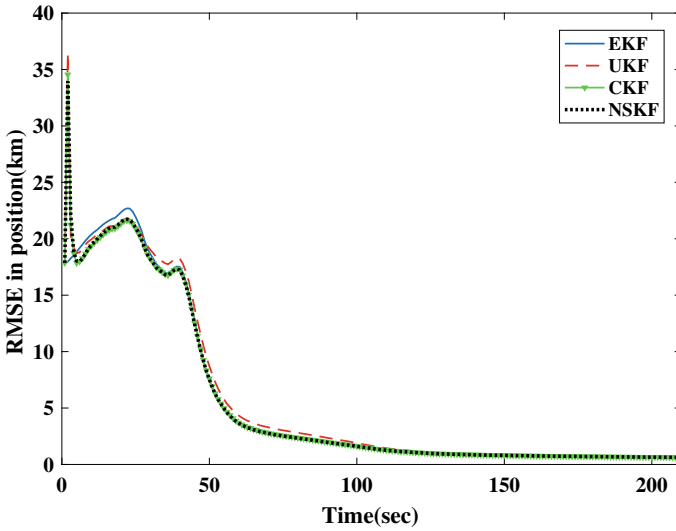


Fig. 4 RMSE in position

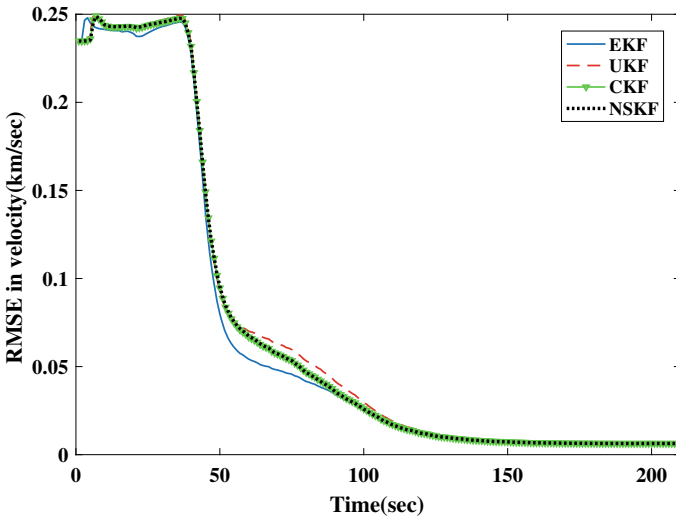


Fig. 5 RMSE in velocity

performs with slightly higher accuracy. However, in the case of RMSE in velocity from Table 4 EKF and NSKF, both have comparable accuracy for lower measurement noise. But with the increase in measurement noise, NSKF in comparison with other filters performs with slightly better accuracy. It should be also noted that, when the noise in bearing and elevation angles is increased from 0.057° to 1° , the error in

Table 3 RMSE in position

Final RMS position error (km)					Time-averaged RMS position error (km)				
Filter	Measurement noise Standard deviation (°)				Filter	Measurement noise Standard deviation (°)			
	0.057	0.65	0.8	1		0.057	0.65	0.8	1
EKF	0.626	4.932	5.467	5.979	EKF	0.900	6.710	7.859	9.293
UKF	0.671	4.968	5.483	5.955	UKF	0.983	6.821	7.995	9.450
CKF	0.634	4.951	5.482	5.946	CKF	0.906	6.802	7.993	9.441
NSKF	0.631	4.948	5.477	5.944	NSKF	0.903	6.787	7.979	9.226

Table 4 RMSE in velocity

Final RMS velocity error (km/s)					Time-averaged RMSE in velocity (km/s)				
Filter	Measurement noise Standard deviation (°)				Filter	Measurement noise Standard deviation (°)			
	0.057	0.65	0.8	1		0.057	0.65	0.8	1
EKF	0.0063	0.0491	0.0539	0.0578	EKF	0.0099	0.0566	0.0618	0.0662
UKF	0.0067	0.0494	0.0538	0.0575	UKF	0.108	0.0572	0.0622	0.0665
CKF	0.0063	0.0492	0.0538	0.0575	CKF	0.010	0.0570	0.0622	0.0667
NSKF	0.0063	0.0491	0.0537	0.0575	NSKF	0.0099	0.0570	0.0622	0.0660

Table 5 RMSE in position with varying $P_{0|0}$

Final RMS position error (km)				Time-averaged RMS position error (km)			
Filter	$P = 1.5 * P$	$P = 2 * P$	$P = 3 * P$	Filter	$P = 1.5 * P$	$P = 2 * P$	$P = 3 * P$
EKF	6.247	7.012	8.217	EKF	9.072	9.799	10.912
UKF	6.171	6.775	7.731	UKF	9.176	9.727	10.569
CKF	6.169	6.785	7.858	CKF	9.177	9.710	10.681
NSKF	6.204	6.835	7.970	NSKF	9.212	9.760	10.815

RMS position shifted from 631 m to 5.9 km. This indicates a significant decrease in the accuracy of the nonlinear filters implemented.

The initial covariance matrix was computed as $P = \nu \times P_{0|0}$ where the value of $\nu = 1.5, 2, 3$ to analyse the effect of initial uncertainty. Here, the measurement standard deviation for bearing and elevation angle was set to be 1° . From Tables 5 and 6, it can be seen that EKF in comparison with NSKF shows more error with the increase in the uncertainty. But still, it cannot be considered as a significant increase in estimation accuracy, as it is comparable.

Table 6 RMSE in velocity with varying $P_{0|0}$

Final RMSE in velocity (km/s)				Time-averaged RMSE in velocity (km/s)			
Filter	$P = 1.5 * P$	$P = 2 * P$	$P = 3 * P$	Filter	$P = 1.5 * P$	$P = 2 * P$	$P = 3 * P$
EKF	0.0618	0.0690	0.0820	EKF	0.0678	0.0740	0.0839
UKF	0.0609	0.0663	0.0765	UKF	0.0679	0.0734	0.0807
CKF	0.0608	0.0664	0.0779	CKF	0.0680	0.0732	0.0815
NSKF	0.0612	0.0669	0.0790	NSKF	0.0684	0.0737	0.0827

5 Conclusion

This paper presents a comparative study of certain nonlinear sigma point Kalman filters for solving the problem of 3D bearings-only target tracking. The filters used were EKF, UKF, CKF, and NSKF. The filtering accuracy is analysed using the RMSE in both position and velocity. From the RMSE, we focused on two criteria: (1) RMS error in position and velocity at the final simulation time step (2) the time-averaged RMS position and velocity error, after the end of observer manoeuvre. To make it more meaningful, the two criteria were evaluated by increasing the measurement noise standard deviation in bearing as well as elevation angles. It is observed that the accuracy of all filters decreases with the increase in measurement noise, but at a similar rate without any significant difference. A very small increase in accuracy was observed for the sigma point filters when compared to the EKF. We would also like to emphasise the fact that these results are valid only for the particular tracking scenario considered, and not a generalised one for 3D BOT problems. Except for highly nonlinear scenarios (where the rate at which the bearing angle rate is very high)[11], we expect the filters to perform in a similar fashion as described in the paper. An attempt was made to evaluate the two criteria with varying initial uncertainty. Here also, comparable results were obtained for sigma point filters with respect to the EKF.

References

1. Nardone, S., Lindgren, A.G., Gong, K.: Fundamental properties and performance of conventional bearings-only target motion analysis. *IEEE Trans. Autom. Control* **29**(9), 775–787 (1984)
2. Radhakrishnan, R., Singh, A.K., Bhaumik, S., Tomar, N.K.: Quadrature filters for underwater passive bearings-only target tracking. In: 2015 Sensor Signal Processing for Defence (SSPD), pp. 1–5. IEEE, Edinburgh (2015). <https://doi.org/10.1109/SSPD.2015.7288519>
3. Blackman, S., Popoli, R.: *Design and Analysis of Modern Tracking Systems*. Artech House, Norwood, MA (1999)
4. Ristic, B., Arulampalam, S., Gordon, N.: *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, Boston (2003)

5. Bar-Shalom, Y., Li, X.R., Kirubarajan, T.: Estimation, Tracking and Navigation: Theory, Algorithms and Software. Wiley, New York (2001)
6. Peach, N.: Bearings-only tracking using a set of range-parameterised extended Kalman filters. In: IEE Proceedings-Control Theory and Applications, vol. 142, issue 1, pp. 73–80. IET (1995)
7. Simon, J., Uhlmann, J., Hugh, F., Durrant, W.: A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Trans. Autom. Control* **45**(3), 477–482 (2000)
8. Arasaratnam, I., Haykin, S.: Cubature Kalman filter. *IEEE Trans. Autom. Control* **54**(6), 1254–1269 (2009)
9. Bhaumik, S., Swati.: Cubature quadrature Kalman filter. *IET Signal Process.* **7**(7), 533–541 (2013). <https://doi.org/10.1049/iet-spr.2012.0085>
10. Ito, K., Xiong, K.: Gaussian filters for nonlinear filtering problems. *IEEE Trans. Autom. Control* **45**(5), 910–927 (2000)
11. Radhakrishnan, R., Bhaumik, S., Tomar, N.K.: Gaussian sum shifted Rayleigh filter for underwater bearings-only target tracking problems. *IEEE J. Ocean. Eng.* **44**(2), 492–501 (2018)
12. Kotecha, J.H., Djuric, P.M.: Gaussian sum particle filtering. *IEEE Trans. Signal Process.* **51**(10), 2602–2612 (2003)
13. Alspach, D., Sorenson, H.: Nonlinear Bayesian estimation using Gaussian sum approximations. *IEEE Trans. Autom. Control* **17**(4), 439–448 (1972)
14. Lingren, A.G., Gong, K.F.: Position and velocity estimation via bearing observations. *IEEE Trans. Aerosp. Electron. Syst.* **AES-14**(4), 564–577 (1978)
15. Doğançay, K.: On the bias of linear least squares algorithms for passive target localization. *Signal Process.* **84**(3), 475–486 (2004) (Elsevier)
16. Aidala, V.J., Nardone, S.C.: Biased estimation properties of the pseudolinear tracking filter. *IEEE Trans. Aerosp. Electron. Syst.* **AES-18**(4), 432–441 (1978)
17. Chan, Y.T., Rudnicki, S.W.: Bearings-only and Doppler-bearing tracking using instrumental variables. *IEEE Trans. Aerosp. Electron. Syst.* **28**(4), 1076–1083 (1992)
18. Doğançay, K.: Passive emitter localization using weighted instrumental variables. *Signal Process.* **84**(4), 487–497 (2004) (Elsevier)
19. Lindgren, A.G., Gong, K.F.: Properties of a bearings-only motion analysis estimator: an interesting case study in system observability. In: Proceedings of the 12th Asilomar Conference on Circuits Systems, vol. 78, issue 1, pp. 61–78. Monterey, CA (1978)
20. Badriasl, L., Arulampalam, S., Finn, A.: A novel batch Bayesian WIV estimator for three-dimensional TMA using bearing and elevation measurements. *IEEE Trans. Signal Process.* **66**(4), 1023–1036 (2017)
21. Mallick, M., Morelande, M., Mihaylova, L., Arulampalam, S., Yan, Y.: Comparison of angle-only filtering algorithms in 3D using Cartesian and modified spherical coordinates. In: 2012 15th International Conference on Information Fusion, pp. 1392–1399. IEEE (2011)
22. Badriasl, L., Doğançay, K.: Three-dimensional target motion analysis using azimuth/elevation angles. *IEEE Trans. Aerosp. Electron. Syst.* **50**(4), 3178–3194 (2014)
23. Badriasl, L., Sathyan, T., Arulampalam, S., Finn, A.: A novel closed-form estimator for 3D TMA using heterogeneous sensors. *IEEE Trans. Signal Process.* **63**(8), 1895–1910 (2015)
24. Mallick, M., Morelande, M., Mihaylova, L., Arulampalam, S., Yan, Y.: Angle-only filtering in three dimensions. In: Integrated Tracking, Classification, and Sensor Management. Wiley Online Library (2013)
25. Gupta, S.D., Yu, J.Y., Mallick, M., Coates, M., Morelande, M.: Comparison of angle-only filtering algorithms in 3D using EKF, UKF, PF, PFF, and ensemble KF. In: 18th International Conference on Information Fusion (Fusion), pp. 1649–1656. IEEE, Washington (2015)
26. Mehrjouyan, A., Alfi, A.: Robust adaptive unscented Kalman filter for bearings-only tracking in three dimensional case. *Appl. Ocean Res.* **87**, 223–232 (2019)
27. Badriasl, L., Arulampalam, S., Hoek, J., Finn, A.: Bayesian WIV estimators for 3-D bearings-only TMA with speed constraints. *IEEE Trans. Signal Process.* **67**(13), 3576–3591 (2019)
28. Radhakrishnan, R., Yadav, A., Date, P., Bhaumik, S.: A new method for generating sigma points and weights for nonlinear filtering. *IEEE Control Syst. Lett.* **2**(3), 519–524 (2018)