

Article

Tracking an Underwater Object with Unknown Sensor Noise Covariance Using Orthogonal Polynomial Filters

Kundan Kumar ¹, Shovan Bhaumik ¹ and Sanjeev Arulampalam ^{2,3,*}

¹ Department of Electrical Engineering, Indian Institute of Technology Patna, Patna 801103, India; kundan.pee16@iitp.ac.in (K.K.); shovan.bhaumik@iitp.ac.in (S.B.)

² Maritime Division, Defence Science and Technology (DST) Group, Edinburgh, SA 5111, Australia

³ Faculty of Engineering, Computer & Mathematical Sciences, The University of Adelaide, Adelaide, SA 5005, Australia

* Correspondence: sanjeev.arulampalam@defence.gov.au

Abstract: In this manuscript, an underwater target tracking problem with passive sensors is considered. The measurements used to track the target trajectories are (i) only bearing angles, and (ii) Doppler-shifted frequencies and bearing angles. Measurement noise is assumed to follow a zero mean Gaussian probability density function with unknown noise covariance. A method is developed which can estimate the position and velocity of the target along with the unknown measurement noise covariance at each time step. The proposed estimator linearises the nonlinear measurement using an orthogonal polynomial of first order, and the coefficients of the polynomial are evaluated using numerical integration. The unknown sensor noise covariance is estimated online from residual measurements. Compared to available adaptive sigma point filters, it is free from the Cholesky decomposition error. The developed method is applied to two underwater tracking scenarios which consider a nearly constant velocity target. The filter's efficacy is evaluated using (i) root mean square error (RMSE), (ii) percentage of track loss, (iii) normalised (state) estimation error squared (NEES), (iv) bias norm, and (v) floating point operations (flops) count. From the simulation results, it is observed that the proposed method tracks the target in both scenarios, even for the unknown and time-varying measurement noise covariance case. Furthermore, the tracking accuracy increases with the incorporation of Doppler frequency measurements. The performance of the proposed method is comparable to the adaptive deterministic support point filters, with the advantage of a considerably reduced flops requirement.

Keywords: target motion analysis; bearings-only measurement; Doppler-shifted frequency; unknown measurement noise covariance; orthogonal polynomial; sigma point filters



Citation: Kumar, K.; Bhaumik, S.; Arulampalam, S. Tracking an Underwater Object with Unknown Sensor Noise Covariance Using Orthogonal Polynomial Filters. *Sensors* **2022**, *22*, 4970. <https://doi.org/10.3390/s22134970>

Academic Editor: Andrzej Stateczny

Received: 10 May 2022

Accepted: 28 June 2022

Published: 30 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Target tracking of an underwater object in a passive mode is a challenging problem owing to the nonlinear and low information content of the passive measurements, coupled with the complexities of the ocean environment [1–6]. A typical passive measurement used in such problems is bearings, and the resulting tracking process is termed bearings-only tracking (BOT), where the objective is to estimate the position and velocity of a moving target using bearings-only measurements. Another type of underwater tracking utilises both bearings and Doppler measurements received from the target and is known as Doppler-bearing tracking (DBT). Due to their importance in underwater target tracking, both BOT and DBT have received considerable attention in the literature [7–11]. Since the main aim in the solution to this problem is to estimate the position and velocity of a moving target, this problem is often known as target motion analysis (TMA) [2,7].

In this paper, we assume that the object is approaching towards the ownship following a nearly constant velocity path. The sonar mounted on the ownship measures only

bearings or Doppler and bearings. The TMA for such a scenario is challenging due to the following reasons:

- (i) With bearings-only measurements, the system is unobservable until the ownship executes a maneuver.
- (ii) In certain TMA applications, fast convergence of the solution is important, which is challenging with passive measurements.
- (iii) The measurement equation is highly nonlinear, which limits the ability of the estimator to provide reliable tracking performance.
- (iv) Sometimes, the measurement noise covariance is time-varying and unknown.

As we mentioned above, in the passive tracking problem, the measurement equations (both bearings-only and Doppler bearings) are nonlinear, and we need to implement a nonlinear filter to estimate the target's trajectory. The extended Kalman filter (EKF) [12,13] and its variants were initially applied for such problems [14–17]. However, they have limitations such as poor estimation accuracy and track divergence, particularly for high initial errors [13,18]. To improve the accuracy of estimation in general, a few deterministic sample point filters such as the unscented Kalman filter (UKF) [19,20], Gauss-Hermite filter (GHF) [21,22], and cubature Kalman filter (CKF) [23,24], etc., were developed. In these filters, the probability density functions are approximated with a few deterministic points and weights, which are propagated and updated using the process and measurement equation, respectively. The Cholesky decomposition operation on error covariance is required to be performed in all such filters. Due to processing software limitations, a round-off error arises, which frequently results in error covariances to be asymmetric and non-positive definite, which ultimately leads to unstable filtering. To circumvent this problem, square root filters [25–27] were proposed. Although the square root filters are numerically more stable, they are computationally more expensive.

Very recently, an orthogonal polynomial approximation-based filter was proposed [28], which can be seen as an alternative to the square root filtering. This filter uses an orthogonal polynomial approximation to linearise a nonlinear system, and subsequently the Kalman filtering approach is used. The coefficients of the linearised model are calculated by evaluating an intractable integral using a weighted sum of a few deterministic sample points. When these sample points are generated using the cubature rule, the filter is called cubature orthogonal polynomial-based EKF (CO-EKF), and similarly, when the unscented transform and Gauss-Hermite integration rules are used, the proposed filters are named UO-EKF and GHO-EKF, respectively. These filters are more robust and retain the desirable properties of error covariance matrices during software implementation [28], i.e., they are free from the Cholesky decomposition error. It is reported that the orthogonal polynomial filters provide almost similar accuracy to their square root counterparts with much less computation cost.

In this paper, we reformulate the orthogonal polynomial approximation-based filter proposed in [28] in an alternative way which is more straightforward and easier to follow. Further, we extend the orthogonal polynomial approximation filter in such a way that it can be applied to estimate the states when measurement noise covariance is not known. The proposed filter is also capable of estimating time-varying measurement noise covariance along with the states. We consider two tracking scenarios where the tracking is performed with bearings-only and Doppler-bearing measurements. In the first scenario, the ownship executes an evasive maneuver smoothly, and in the second case, it executes a sharp maneuver. The proposed adaptive orthogonal polynomial filters are implemented and their performance is compared with the corresponding square root filters in terms of (i) root mean square error (RMSE), (ii) percentage track loss, (iii) normalised (state) estimation error squared (NEES), (iv) bias norm, (v) floating point operations (flops) count, and (vi) computation time.

From the simulation results, it is observed that the proposed method tracks the target in both the scenarios even for unknown time-varying measurement noise covariance, and tracking accuracy increases with the incorporation of Doppler frequency measurements.

While the percentage of track loss of the developed method is comparable to adaptive deterministic support point filters, the flops requirement of our approach is considerably less. The robustness of the proposed method is tested by assuming a target maneuver with a small and constant turn rate, while the estimators wrongly modelled it as constant velocity. To compensate for this, the estimators use a higher process noise covariance, and it has been shown that the proposed filters work with acceptable accuracy, particularly for DBT.

The remaining part of the paper is organised as follows. An underwater tracking problem is formulated in Section 2. In Section 3, we derive the proposed filter. The simulation results are discussed in Section 4, followed by a brief discussion and conclusions.

2. Problem Formulation

2.1. System Model

The state vector of a target at time index k is given by $\mathcal{X}_k^t = [\mathcal{X}_{1,k}^t \quad \mathcal{X}_{2,k}^t \quad \dot{\mathcal{X}}_{1,k}^t \quad \dot{\mathcal{X}}_{2,k}^t]'$, where $(\mathcal{X}_{1,k}^t, \mathcal{X}_{2,k}^t)$ are the true positions and $(\dot{\mathcal{X}}_{1,k}^t, \dot{\mathcal{X}}_{2,k}^t)$ are the true velocities of the target along the x and y axis, respectively, and $'$ represents the transpose of a matrix. Similarly, the state vector of the observer (which is ownship) can be expressed as $\mathcal{X}_k^o = [\mathcal{X}_{1,k}^o \quad \mathcal{X}_{2,k}^o \quad \dot{\mathcal{X}}_{1,k}^o \quad \dot{\mathcal{X}}_{2,k}^o]'$.

The target dynamics in discrete time can be expressed as

$$\mathcal{X}_k^t = F\mathcal{X}_{k-1}^t + \eta_{k-1}, \quad (1)$$

where F is the state transition matrix. The process noise, η_{k-1} , follows a Gaussian distribution with zero mean and covariance,

$$Q_{k-1} = \tilde{q} \begin{bmatrix} \frac{T^3}{3} I_{2 \times 2} & \frac{T^2}{2} I_{2 \times 2} \\ \frac{T^2}{2} I_{2 \times 2} & T I_{2 \times 2} \end{bmatrix},$$

where \tilde{q} is the process noise intensity, and T is the sampling time. We define the relative state vector as

$$\mathcal{X}_k = \mathcal{X}_k^t - \mathcal{X}_k^o. \quad (2)$$

Using Equation (1), we can write the state equation as

$$\begin{aligned} \mathcal{X}_k &= F\mathcal{X}_{k-1}^t + \eta_{k-1} - \mathcal{X}_k^o \\ &= F(\mathcal{X}_{k-1} + \mathcal{X}_{k-1}^o) - \mathcal{X}_k^o + \eta_{k-1} \\ &= F\mathcal{X}_{k-1} - U_{k-1,k} + \eta_{k-1}, \end{aligned} \quad (3)$$

where $U_{k-1,k} = \mathcal{X}_k^o - F\mathcal{X}_{k-1}^o$ is known as the vector of deterministic inputs [7]. Here, we consider a single target which follows (i) a nearly constant velocity path described by a constant velocity (CV) model, or (ii) a maneuvering trajectory described by a constant turn (CT) model with constant and known turn rate. It is interesting to note that, for both cases, the process equation is linear.

2.1.1. Constant Velocity Model

The state transition matrix for the CV model is given by

$$F = \begin{bmatrix} I_{2 \times 2} & T I_{2 \times 2} \\ 0_{2 \times 2} & I_{2 \times 2} \end{bmatrix}, \quad (4)$$

and the vector of deterministic inputs [2,7] becomes

$$U_{k-1,k} = \begin{bmatrix} \mathcal{X}_{1,k}^0 - \mathcal{X}_{1,k-1}^0 - T\dot{\mathcal{X}}_{1,k-1}^0 \\ \mathcal{X}_{2,k}^0 - \mathcal{X}_{2,k-1}^0 - T\dot{\mathcal{X}}_{2,k-1}^0 \\ \dot{\mathcal{X}}_{1,k}^0 - \dot{\mathcal{X}}_{1,k-1}^0 \\ \dot{\mathcal{X}}_{2,k}^0 - \dot{\mathcal{X}}_{2,k-1}^0 \end{bmatrix}. \quad (5)$$

2.1.2. Constant Turn Model

In this case, it is assumed that the target maneuvers with a constant and known turn rate, ω . The state transition matrix of this model is expressed as [23,29,30]

$$F = \begin{bmatrix} 1 & 0 & \frac{\sin(\omega T)}{\omega} & -\frac{1 - \cos(\omega T)}{\omega} \\ 0 & 1 & \frac{1 - \cos(\omega T)}{\omega} & \frac{\sin(\omega T)}{\omega} \\ 0 & 0 & \cos(\omega T) & -\sin(\omega T) \\ 0 & 0 & \sin(\omega T) & \cos(\omega T) \end{bmatrix}, \quad (6)$$

and the expression of $U_{k-1,k}$ becomes

$$U_{k-1,k} = \begin{bmatrix} \mathcal{X}_{1,k}^0 - \mathcal{X}_{1,k-1}^0 - \frac{\sin(\omega T)}{\omega} \dot{\mathcal{X}}_{1,k-1}^0 + \frac{1 - \cos(\omega T)}{\omega} \dot{\mathcal{X}}_{2,k-1}^0 \\ \mathcal{X}_{2,k}^0 - \mathcal{X}_{2,k-1}^0 - \frac{1 - \cos(\omega T)}{\omega} \dot{\mathcal{X}}_{1,k-1}^0 - \frac{\sin(\omega T)}{\omega} \dot{\mathcal{X}}_{2,k-1}^0 \\ \dot{\mathcal{X}}_{1,k}^0 - \cos(\omega T) \dot{\mathcal{X}}_{1,k-1}^0 + \sin(\omega T) \dot{\mathcal{X}}_{2,k-1}^0 \\ \dot{\mathcal{X}}_{2,k}^0 - \sin(\omega T) \dot{\mathcal{X}}_{1,k-1}^0 - \cos(\omega T) \dot{\mathcal{X}}_{2,k-1}^0 \end{bmatrix}. \quad (7)$$

It is to be noted that in the limiting condition $\omega \rightarrow 0$, Equations (6) and (7) reduce to Equations (4) and (5), respectively.

2.2. Measurement Model

Measurement sensors are mounted on the ownship and all the sensors are passive so that some specific tactical advantages can be achieved. It is assumed that the sonar mounted on the ownship is only capable of measuring the bearing angle. We assume the following two cases.

2.2.1. Case I—Bearings-Only Measurement

The sensors measure the direction of the target location with respect to true north. Such measurements can be expressed as

$$\mathcal{Y}_k = \gamma(\mathcal{X}_k) + \nu_{\theta_k}, \quad (8)$$

where $\gamma(\mathcal{X}_k) = \tan^{-1}(\mathcal{X}_{1,k}/\mathcal{X}_{2,k})$, and ν_{θ_k} is noise in bearing measurement, which is assumed to be zero mean, white Gaussian with standard deviation, σ_θ , i.e., $\nu_{\theta_k} \sim \mathcal{N}(0, \sigma_\theta^2)$. For such measurements, the system is unobservable [2,7] and the estimation of the target state is only possible when the ownship starts maneuvering. It is to be noted that even if, sometimes, the sensor measures the bearing angle with respect to its own local coordinates, it is easy to transform the measurement with respect to true north.

2.2.2. Case II—Doppler-Bearing Measurement

As the target is moving, a Doppler shift occurs on the tonal frequency emitted from the target, which can also be measured. In such cases, both the noise-corrupted bearing angles and Doppler-shifted frequencies are available. This is popularly known as Doppler-bearing (DB) measurement and the overall measurement function $\gamma(\mathcal{X}_k)$ becomes

$$\gamma(\mathcal{X}_k) = \begin{bmatrix} \tan^{-1}(\mathcal{X}_{1,k}/\mathcal{X}_{2,k}) \\ f_T(1 - \frac{\dot{\mathcal{X}}_{1,k}\mathcal{X}_{1,k}}{c r_k} - \frac{\dot{\mathcal{X}}_{2,k}\mathcal{X}_{2,k}}{c r_k}) \end{bmatrix}, \quad (9)$$

where f_T is the constant tonal frequency emitted from the target, $r_k^2 = \mathcal{X}_{1,k}^2 + \mathcal{X}_{2,k}^2$, c is the speed of the sound in water, and the measurement noise vector in this case is $v_k = [v_{\theta_k} \ v_{f_k}]'$. Note that v_{f_k} is the sensor noise in the Doppler measurement and assumed to be zero mean, white, Gaussian with variance σ_f^2 , i.e., $v_{f_k} \sim \mathcal{N}(0, \sigma_f^2)$. Further, we assume that the noises in bearing and Doppler measurements is uncorrelated, with overall measurement noise covariance $R_k = \text{diag}(\sigma_\theta^2, \sigma_f^2)$.

It is to be noted that for Doppler-bearing (DB) measurements, the system is observable and maneuvering of the ownship is not required for estimators to converge [9]. Further, it is expected that if noise variance for Doppler measurement is reasonably low, the Doppler-bearing tracking (DBT) should provide more accurate results than bearings-only tracking (BOT).

3. Tracking Methodology

In this section, we formulate a tracking filter for underwater TMA when sensor noise covariance is unknown. The proposed filter linearises the nonlinear measurement function using a first-order orthogonal polynomial approximation with a Gaussian measure as a weight function and estimates the measurement noise covariance R_k at each step.

3.1. Orthogonal Polynomial Filter

The proposed orthogonal polynomial filter is a kind of linearised Kalman filter, where, unlike the extended Kalman filter, the linearisation is not done using the Taylor series; rather, it is done with a first-order Hermite polynomial. The method was developed for a general nonlinear state estimation problem in our earlier publication [28]. Here, we provide a more straightforward derivation of the filter for an underwater tracking problem.

The problem considered here consists of a linear process model, so the time update step can be done by the Kalman filter (KF). The available measurements are assumed to be (i) bearings, (ii) bearings with Doppler-shifted frequencies. Thus, our measurement $\mathcal{Y}_k \in \mathbb{R}^{n_y}$, where n_y is 1 or 2 for bearings and Doppler-bearing measurements, respectively. As the measurement equation is nonlinear, it is linearised with the first-order Hermite polynomial, and subsequently the KF scheme is applied for the measurement update.

Let us assume an arbitrary function of Gaussian random variable (r.v.) \mathcal{X} , $f(\mathcal{X})$, which maps \mathcal{X} to a real space of arbitrary dimension n_p , i.e., $f(\mathcal{X}) : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_p}$ is approximated with a first-order Hermite polynomial [28,31–33]; so,

$$f(\mathcal{X}) \approx A_0 H_0(\mathcal{X}) + A_1 H_1(\mathcal{X}), \quad (10)$$

where the coefficients A_0 and A_1 are matrices with dimension $n_p \times 1$ and $n_p \times n_x$, respectively. $H_0(\mathcal{X})$ and $H_1(\mathcal{X})$ are the zero and first-order Hermite polynomial, respectively. Please note that in Equation (10), we neglect higher-order terms so that the function is approximated by a linear equation. All the terms with higher-order Hermite polynomials that are neglected in Equation (10) combined together form the residual error. We choose the Hermite polynomial since its weighting function expansion is the same as the Gaussian [31,32]. As the Hermite polynomial is orthogonal [33],

$$\langle H_i(\mathcal{X}), H_j(\mathcal{X}) \rangle = \begin{cases} I_{n_x} & \text{for } i = j, \\ 0 & \text{otherwise,} \end{cases} \quad (11)$$

where \langle, \rangle represents the dot product, $\mathcal{X} \sim \mathcal{N}(\hat{\mathcal{X}}, P)$, and I_{n_x} is the identity matrix of dimension n_x . The coefficient A_i can be calculated as

$$\begin{aligned} A_i &= \langle f(\mathcal{X}), H_i(\mathcal{X}) \rangle \\ &= \int_{-\infty}^{\infty} f(\mathcal{X}) H_i(\mathcal{X})' \mathcal{N}(\mathcal{X}, \hat{\mathcal{X}}, P) d\mathcal{X}. \end{aligned} \quad (12)$$

It is easy to verify that $H_0(\mathcal{X}) = 1$ and $H_1(\mathcal{X}) = S^{-1}(\mathcal{X} - \hat{\mathcal{X}})$, where $P = SS'$. Substituting the values of $H_0(\mathcal{X})$ and $H_1(\mathcal{X})$ in Equation (10), the approximation of $f(\mathcal{X})$ becomes

$$f(\mathcal{X}) \approx A_0 + A_1 S^{-1}(\mathcal{X} - \hat{\mathcal{X}}). \quad (13)$$

Following Equation (13), we can linearise the measurement function $\gamma(\mathcal{X}_k)$ as

$$\gamma(\mathcal{X}_k) \approx A_k + B_k S_{k|k-1}^{-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k-1}), \quad (14)$$

where the state $\mathcal{X}_k \in \mathbb{R}^4$ follows a Gaussian distribution with mean $\hat{\mathcal{X}}_{k|k-1}$ and covariance $P_{k|k-1}$, and the coefficients A_k and B_k are $n_y \times 1$ and $n_y \times 4$ matrices, respectively. $S_{k|k-1}$ is the square root of the covariance matrix $P_{k|k-1}$ (i.e., $P_{k|k-1} = S_{k|k-1} S_{k|k-1}'$), which can be calculated by the Cholesky decomposition. Using Equation (12), the coefficient matrices A_k and B_k can be expressed as

$$A_k = \int_{-\infty}^{\infty} \gamma(\mathcal{X}_k) \mathcal{N}(\mathcal{X}_k; \hat{\mathcal{X}}_{k|k-1}, P_{k|k-1}) d\mathcal{X}_k, \quad (15)$$

$$B_k = \int_{-\infty}^{\infty} \gamma(\mathcal{X}_k) (S_{k|k-1}^{-1}(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k-1}))' \mathcal{N}(\mathcal{X}_k; \hat{\mathcal{X}}_{k|k-1}, P_{k|k-1}) d\mathcal{X}_k. \quad (16)$$

The above integrals cannot be solved analytically for any arbitrary nonlinear function [34]. Various methods such as the Gauss-Hermite (GH) quadrature rule [21], unscented transform [19,20], cubature quadrature (CQ) rule [23,24], and high-degree cubature quadrature (HDCQ) rule [35,36] are available to solve such integrals. Here, in brief, we discuss the GH quadrature rule.

Gauss-Hermite quadrature rule: A single-dimensional Gauss-Hermite (GH) quadrature rule [21,34] is given by

$$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} f(x) \exp(-x^2) dx \approx \sum_{i=1}^m f(\xi_i) \omega_i, \quad (17)$$

where ξ_i are the quadrature points with associated weights ω_i for $i = 1, \dots, m$. The above integral can be extended for a multidimensional variable using the product rule [21]. To calculate these quadrature points, we take a symmetric tridiagonal matrix J , with zero diagonal elements and $J_{i,i+1} = \sqrt{i/2}, 1 \leq i \leq (m-1)$ [21,37]. The quadrature points are located at $\xi_i = \sqrt{2}\psi_i$, where ψ_i is the i -th eigenvalue of the matrix J . The weight $\omega_i = |v_{i,1}|^2$, where $v_{i,1}$ is the first element of the i -th normalised eigenvector of J . Thus, a multidimensional integral

$$I = \int_{-\infty}^{\infty} f(\mathcal{X}) \mathcal{N}(\mathcal{X}, 0, I_{n_x}) d\mathcal{X} \quad (18)$$

can be approximately evaluated as [38]

$$I = \sum_{i_1=1}^m \cdots \sum_{i_{n_x}=1}^m f(\xi_{i_1}, \xi_{i_2}, \dots, \xi_{i_{n_x}}) \omega_{i_1} \omega_{i_2} \cdots \omega_{i_{n_x}}. \quad (19)$$

The total number of required points in the above method for any n_x dimensional system is m^{n_x} . Here, we see that the number of points in the GH quadrature rule increases exponentially with the system's dimension. For more about the point and weight generation methods of the UKF, CQKF, HDCQKF, readers are referred to [19,21,24,36].

Calculation of A_k and B_k : To calculate A_k and B_k (Equations (15) and (16)) using the above-discussed GH quadrature rule, we need to transform Equations (15) and (16) into standard Gaussian integrals. Transforming the Gaussian r.v. \mathcal{X}_k into a standard Gaussian r.v. x_k following $\mathcal{X}_k = \hat{\mathcal{X}}_{k|k-1} + S_{k|k-1}x_k$, the integrals (Equations (15) and (16)) become

$$A_k = \int_{-\infty}^{\infty} \gamma(\hat{\mathcal{X}}_{k|k-1} + S_{k|k-1}x_k)\mathcal{N}(x_k; 0, I_4)dx_k, \tag{20}$$

$$B_k = \int_{-\infty}^{\infty} \gamma(\hat{\mathcal{X}}_{k|k-1} + S_{k|k-1}x_k)x_k'\mathcal{N}(x_k; 0, I_4)dx_k. \tag{21}$$

Using the points (ζ_i) and weights (ω_i), the coefficients A_k (Equation (20)) can be expressed as

$$A_k = \sum_{i=1}^m \gamma(\hat{\mathcal{X}}_{k|k-1} + S_{k|k-1}\zeta_i)\omega_i = \sum_{i=1}^m \gamma(\chi_{i,k|k-1})\omega_i, \tag{22}$$

where $\chi_{i,k|k-1} = \hat{\mathcal{X}}_{k|k-1} + S_{k|k-1}\zeta_i$. Similarly, the B_k is calculated as

$$B_k = \sum_{i=1}^m \gamma(\chi_{i,k|k-1})\zeta_i'\omega_i. \tag{23}$$

For bearings-only measurements given in Equation (8), $A_k = \sum_{i=1}^m \tan^{-1}(\chi_{\{1,i,k|k-1\}} / \chi_{\{2,i,k|k-1\}})$ and $B_k = \sum_{i=1}^m \tan^{-1}(\chi_{\{1,i,k|k-1\}} / \chi_{\{2,i,k|k-1\}})\zeta_i'\omega_i$, where $\chi_{1,i}$ represents the first element of the i -th transformed quadrature point. For the DB measurements given in Equation (9), the expressions of A_k and B_k become

$$A_k = \sum_{i=1}^m \left[\tan^{-1}(\chi_{\{1,i,k|k-1\}} / \chi_{\{2,i,k|k-1\}}) \quad f_{s_k}^i \right]' \omega_i,$$

$$B_k = \sum_{i=1}^m \left[\tan^{-1}(\chi_{\{1,i,k|k-1\}} / \chi_{\{2,i,k|k-1\}}) \quad f_{s_k}^{(i)} \right]' \zeta_i'\omega_i,$$

where $f_{s_k}^i = f_T(1 - \frac{\hat{\chi}_{1,i,k|k-1}\chi_{1,i,k|k-1}}{c r_k^i} - \frac{\hat{\chi}_{2,i,k|k-1}\chi_{2,i,k|k-1}}{c r_k^i})$, $r_k^i = \sqrt{\hat{\chi}_{1,i,k|k-1}^2 + \hat{\chi}_{2,i,k|k-1}^2}$.

Remark 1. Even with the assumption of Gaussian pdf of states, the approximation arises due to (i) neglecting the terms containing second- and higher-order Hermite polynomials; (ii) approximate evaluation of the coefficients A_0 and A_1 by the summation method. However, if the nonlinear function is in polynomial form, the exact evaluation of coefficients can be done with the Gaussian integral [17].

Remark 2. On neglecting higher-order Hermite polynomial terms in the approximation (Equation (10)), an accumulative error may occur. It can be compensated by the following approaches [12] (pp. 395–402): (i) increasing the process noise artificially; (ii) multiplying the error covariance by a factor slightly greater than unity. In both cases, the Kalman gain increases and more weight is given to the present innovation, thus compensating for the error which occurs during linearisation.

Remark 3. From the above discussion, we have seen that A_k and B_k can be evaluated using the weighted sum of deterministic sample points. If we use the GH quadrature rule [21,34] to obtain the points, the estimation method is called Gauss-Hermite orthogonal polynomial-based EKF (GHO-EKF). The extension EKF is used because the proposed method linearises the nonlinear function (but not with Taylor series expansion). Similarly, when the unscented transform [19] and CQ rule [23,24] are used to generate the points, the filters are called unscented orthogonal polynomial-based EKF (UO-EKF) and cubature orthogonal polynomial-based EKF (CO-EKF), respectively.

Measurement Update

As we have linearised the measurement equation, the measurement update step can be performed with the Kalman filter (KF) algorithm. The estimated value of the measurement and its covariance can be calculated as

$$\begin{aligned}\hat{\mathcal{Y}}_{k|k-1} &= E[(\gamma(\mathcal{X}_k) + \nu_k) | \mathcal{Y}_{1:k-1}] \\ &= \int_{\mathbb{R}^{n_x}} (A_k + B_k S_{k|k-1}^{-1} (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k-1})) \mathcal{N}(\mathcal{X}_k; \hat{\mathcal{X}}_{k|k-1}, P_{k|k-1}) d\mathcal{X}_k \\ &= A_k,\end{aligned}\quad (24)$$

and

$$\begin{aligned}P_{k|k-1}^{\mathcal{Y}\mathcal{Y}} &= E[(\mathcal{Y}_k - \hat{\mathcal{Y}}_{k|k-1})(\mathcal{Y}_k - \hat{\mathcal{Y}}_{k|k-1})'] \\ &= E[(B_k S_{k|k-1}^{-1} (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k-1}) + \nu_k)(B_k S_{k|k-1}^{-1} (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k-1}) + \nu_k)'] \\ &= B_k S_{k|k-1}^{-1} P_{k|k-1} (S'_{k|k-1})^{-1} B'_k + R_k \\ &= B_k B'_k + R_k.\end{aligned}\quad (25)$$

The cross-covariance between the state and measurement can be calculated as

$$\begin{aligned}P_{k|k-1}^{\mathcal{X}\mathcal{Y}} &= E[(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k-1})(\mathcal{Y}_k - \hat{\mathcal{Y}}_{k|k-1})'] \\ &= E[(\mathcal{X}_k - \hat{\mathcal{X}}_{k|k-1})(B_k S_{k|k-1}^{-1} (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k-1}) + \nu_k)'] \\ &= P_{k|k-1} (S'_{k|k-1})^{-1} B'_k \\ &= S_{k|k-1} B'_k.\end{aligned}\quad (26)$$

Remark 4. Although, throughout our work, we consider the measurement noises to be Gaussian, it may not always be so [30,39,40]. In the case of non-Gaussian noise, it can be approximated by a Gaussian distribution and we can apply the proposed method. However, in this case, since there is a mismatch of noise statistics with the assumed one, robust filters might be helpful. Further, the noise can also be represented as a sum of Gaussians, and Gaussian sum filters [41] with the proposed method may help to achieve the desired results.

3.2. Estimation of Measurement Noise Covariance

The measurement noise considered here is time-varying, zero mean white Gaussian with unknown covariance. In the literature, filtering for such problems is known as adaptive filtering [42–48]. Several adaptive filters have been proposed based on the KF [42,44], EKF [48], UKF [46,49], CKF [50], and GHF [46]. A few works on the TMA using BOT for unknown time-varying measurement noise are found in [51,52]. In this paper, we have developed a noise adaptation technique using the orthogonal polynomial filter for both BOT and DBT. Here, we derive an expression for online R_k estimation compatible with the orthogonal polynomial filter and presented in Lemma 1.

Lemma 1. The covariance of measurement noise, R_k can be expressed as

$$R_k = P_{k|k}^{\mathcal{Y}\mathcal{Y}} + b_k P_{k|k} b'_k, \quad (27)$$

where $P_{k|k}^{\mathcal{Y}\mathcal{Y}}$ is the covariance of residual measurement, $P_{k|k}$ is the posterior error covariance, and $b_k = B_k S_{k|k-1}^{-1}$

Proof. The measurement equation using Equation (14) can be written as

$$\mathcal{Y}_k = A_k + B_k S_{k|k-1}^{-1} (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k-1}) + \nu_k. \quad (28)$$

The expression for the posterior estimate of measurement is

$$\hat{\mathcal{Y}}_{k|k} = A_k + B_k S_{k|k-1}^{-1} (\hat{\mathcal{X}}_{k|k} - \hat{\mathcal{X}}_{k|k-1}). \quad (29)$$

Subtracting Equation (29) from Equation (28), we obtain the residual of measurement

$$\mathcal{Y}_k - \hat{\mathcal{Y}}_{k|k} = b_k (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}) + \nu_k = b_k \epsilon_{k|k} + \nu_k,$$

where $\epsilon_{k|k} = \mathcal{X}_k - \hat{\mathcal{X}}_{k|k}$. The covariance of residual measurement becomes

$$\begin{aligned} P_{k|k}^{\mathcal{Y}\mathcal{Y}} &= E[(\mathcal{Y}_k - \hat{\mathcal{Y}}_{k|k})(\mathcal{Y}_k - \hat{\mathcal{Y}}_{k|k})'] \\ &= E[(b_k \epsilon_{k|k} + \nu_k)(b_k \epsilon_{k|k} + \nu_k)'] \\ &= b_k P_{k|k} b_k' + E[b_k \epsilon_{k|k} \nu_k'] + E[\nu_k \epsilon_{k|k}' b_k'] + R_k. \end{aligned} \quad (30)$$

As \mathcal{X}_k , $\hat{\mathcal{X}}_{k|k-1}$, and $\epsilon_{k|k-1}$ are uncorrelated with measurement noise ν_k , $E[\mathcal{X}_k \nu_k'] = E[\hat{\mathcal{X}}_{k|k-1} \nu_k'] = E[\epsilon_{k|k-1} \nu_k'] = 0$. Using these relations, we can write the following

$$\begin{aligned} E[b_k \epsilon_{k|k} \nu_k'] &= E[b_k (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}) \nu_k'] \\ &= b_k E[\mathcal{X}_k \nu_k' - \hat{\mathcal{X}}_{k|k} \nu_k'] \\ &= -b_k E[(\hat{\mathcal{X}}_{k|k-1} + K_k (\mathcal{Y}_k - \hat{\mathcal{Y}}_{k|k-1})) \nu_k'] \\ &= -b_k E[(\hat{\mathcal{X}}_{k|k-1} + K_k b_k \epsilon_{k|k-1} + K_k \nu_k) \nu_k'] \\ &= -b_k K_k R_k, \end{aligned} \quad (31)$$

where the Kalman gain $K_k = P_{k|k}^{\mathcal{X}\mathcal{Y}} (P_{k|k}^{\mathcal{Y}\mathcal{Y}})^{-1}$. Further, we can evaluate

$$E[\nu_k \epsilon_{k|k}' b_k'] = E[b_k \epsilon_{k|k} \nu_k']' = -R_k K_k' b_k'. \quad (32)$$

Now,

$$\begin{aligned} b_k P_{k|k} b_k' - b_k K_k R_k &= b_k (I - K_k b_k) P_{k|k-1} b_k' - b_k K_k R_k \\ &= b_k P_{k|k-1} b_k' - b_k K_k (b_k P_{k|k-1} b_k' + R_k) \\ &= b_k P_{k|k-1} b_k' - b_k P_{k|k-1} b_k' (b_k P_{k|k-1} b_k' + R_k)^{-1} (b_k P_{k|k-1} b_k' + R_k) \\ &= 0. \end{aligned}$$

or,

$$b_k K_k R_k = b_k P_{k|k} b_k'. \quad (33)$$

Substituting Equations (31)–(33) in Equation (30), we have Equation (27). \square

From the measurement data sequence, $P_{k|k}^{\mathcal{Y}\mathcal{Y}}$ is calculated using the following equation:

$$P_{k|k}^{\mathcal{Y}\mathcal{Y}} = \frac{1}{k - i_0 + 1} \sum_{i=i_0}^k (\mathcal{Y}_i - \hat{\mathcal{Y}}_{i|i})(\mathcal{Y}_i - \hat{\mathcal{Y}}_{i|i})', \quad (34)$$

where i_0 is the user's choice to fix a window length of $(k - i_0 + 1)$. Substituting Equation (34) into Equation (27),

$$R_k = \frac{1}{k - i_0 + 1} \sum_{i=i_0}^k (\mathcal{Y}_i - \hat{\mathcal{Y}}_{i|i})(\mathcal{Y}_i - \hat{\mathcal{Y}}_{i|i})' + b_k P_{k|k} b_k'. \quad (35)$$

The detailed algorithm to implement the proposed adaptive orthogonal polynomial-based filter is presented in Algorithm 1.

Algorithm 1: Pseudo code for adaptive polynomial filter for target tracking

Step 1: Initialisation

- 1 Initialise the filter with $\hat{\mathcal{X}}_{0|0}$, $P_{0|0}$, and R_0 .
- 2 Compute the sample points ζ_i , and their respective weights ω_i .

Step 2: Time update

- 3 Compute the predicted mean and covariance as

$$\hat{\mathcal{X}}_{k|k-1} = F\hat{\mathcal{X}}_{k-1|k-1} - U_{k-1,k},$$

$$P_{k|k-1} = FP_{k-1|k-1}F' + Q_{k-1}.$$

Step 3: Measurement update

- 4 Compute $\hat{\mathcal{Y}}_{k|k-1}$, $P_{k|k-1}^{\mathcal{Y}\mathcal{Y}}$, and $P_{k|k-1}^{\mathcal{X}\mathcal{Y}}$ using Equations (24)–(26).
- 5 Calculate the Kalman gain, $K_k = P_{k|k-1}^{\mathcal{X}\mathcal{Y}} (P_{k|k-1}^{\mathcal{Y}\mathcal{Y}})^{-1}$.
- 6 Estimate the posterior state,

$$\hat{\mathcal{X}}_{k|k} = \hat{\mathcal{X}}_{k|k-1} + K_k(\mathcal{Y}_k - \hat{\mathcal{Y}}_{k|k-1}).$$

- 7 Compute the posterior error covariance,

$$P_{k|k} = P_{k|k-1} - K_k P_{k|k-1}^{\mathcal{Y}\mathcal{Y}} K_k'.$$

Step 4: Estimate unknown R_k

- 8 Calculate the posterior estimate of the measurement, $\hat{\mathcal{Y}}_{k|k}$ using Equation (29).
 - 9 Calculate the residual of measurement as $\mathcal{Y}_k - \hat{\mathcal{Y}}_{k|k}$.
 - 10 Calculate the covariance of measurement residual ($P_{k|k}^{\mathcal{Y}\mathcal{Y}}$) using Equation (34).
 - 11 Estimate R_k using Equation (35).
-

3.3. Computational Advantage

3.3.1. Alternative to the Square Root Filters

In sigma point filters, to calculate the sigma point at each time step, the Cholesky factorisation of the covariance matrix (P), i.e., $P = SS'$, is required. The accumulated round-off error associated with the limited word length arithmetics of the processing software [25–27] sometimes leads to an asymmetric, non-positive definite covariance matrix, which forces the filter to stop. To circumvent such a problem, the square root sigma point filters such as square root UKF (SRUKF), square root quadrature Kalman filter (SRQKF), square root CKF (SRCKF), etc., were developed [23,25–27]. In square root filtering, at each time step, the square root of P , i.e., S , is propagated directly using QR decomposition. The QR decomposition guarantees positive definite covariance matrices at the expense of increased computational burden.

The orthogonal polynomial filters are free from such problems and less sensitive to the round-off error and preserve the symmetry and positive definiteness property [28]. Furthermore, the computational cost of this filter is less than the square root filters, which we show using the flops count in the next subsection.

3.3.2. Flops Count

We analyze the computational complexity of the filters by counting the number of floating point operations (flops) [26,28,53]. A flop is defined as one basic algebraic operation, such as addition, subtraction, multiplication, and division, between any two floating point numbers [26,53]. The filter algorithm consists of the following matrix operations: (i) addition, (ii) multiplication, (iii) inverse, and (iv) Cholesky decomposition. The addition of two matrices with dimension $m \times n$ requires mn flops. The multiplication of an $m \times n$ matrix with $n \times p$ matrix requires $mp(2n - 1)$ number of flops [53]. The inverse and Cholesky operation of any matrix $A \in \mathbb{R}^{n \times n}$ require n^3 and $n^3/3 + 2n^2$ flops, respectively [53]. The QR decomposition of any matrix with dimension $m \times n$ using the Householder method requires $2mn^2 - 2/3n^3$ flops.

We calculate the flops count for all the filters (both ordinary and adaptive) for BOT and DBT measurements, and these are summarised in Table 1. From the table, we see that the filters with DBT measurements require more flops than those with the BOT. The adaptation of the unknown measurement noise covariance brings an extra computational burden on the filtering algorithm. Among all filters, the EKF has the lowest flops count. The orthogonal polynomial filters such as CO-EKF, UO-EKF, and GHO-EKF have lower flops counts than their square root counterparts (SRCKF, SRUKF, and square root GHF (SRGHF)).

Table 1. Flops count of the implemented filters for BOT and DBT.

| Filter | BOT | | DBT | |
|---------|----------|----------|----------|----------|
| | Ordinary | Adaptive | Ordinary | Adaptive |
| EKF | 409 | 516 | 589 | 914 |
| SRCKF | 1197 | 1548 | 1475 | 2053 |
| CO-EKF | 775 | 966 | 1011 | 1440 |
| SRUKF | 1323 | 1711 | 1659 | 2282 |
| UO-EKF | 817 | 1008 | 1065 | 1444 |
| SRGHF | 8090 | 11,142 | 10,299 | 14,162 |
| GHO-EKF | 3913 | 4104 | 4953 | 5382 |

4. Simulation Results

4.1. Tracking Scenarios

In this paper, we consider two tracking scenarios, as shown in Figure 1. From the figure, we see that in both scenarios, the target follows a nearly constant velocity path. In the first engagement scenario (Figure 1a), the ownship maneuvers smoothly during the period 61–420 s with a turn rate of $0.4^\circ/\text{s}$. In the second scenario (Figure 1b), the observer sharply maneuvers at $t = 181$ s and changes its course to 320° from 180° . In the second scenario, the rate of change for the bearing angle during the maneuver is high, which means that it is more difficult for any suboptimal filter to track. It is well known that the maneuvering of ownship makes the system observable. We took two scenarios from the earlier literature [2,7] as benchmark problems so that we could demonstrate the efficacy of our proposed algorithm against the background of existing available results. For both scenarios, the estimators use (i) bearings-only and (ii) bearings and Doppler-shifted frequency measurements, and tracking is performed for 15 min. The speed of sound in water (c) is taken to be 1.5 km/s, and the tonal frequency (f_T) of the target is 385 Hz. Although, in an underwater scenario, the tonal frequency may not be available continuously and it may change after some time interval, here, we assume that the tonal frequency is fixed and available throughout the simulation. The parameters used in the simulation are provided in Table 2.

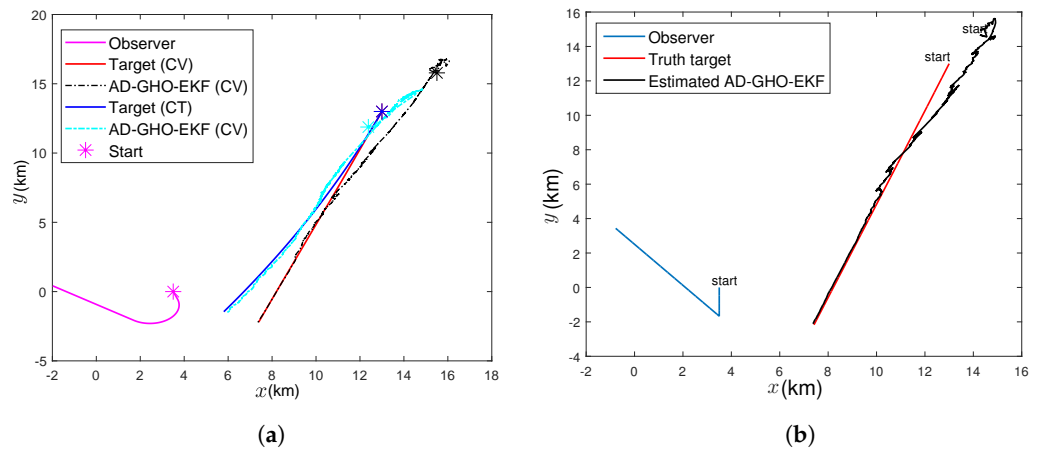


Figure 1. The tracking scenarios: (a) moderately nonlinear; (b) highly nonlinear.

Table 2. Parameters used in tracking scenarios.

| Parameters | Scenario I | Scenario II |
|--|---------------------|---------------------|
| Initial range | 16.1 km | 16.1 km |
| Target speed | 35 knots | 35 knots |
| Target course | -160° | -160° |
| Observer speed | 18 knots | 18 knots |
| Observer initial course | 160° | 180° |
| Observer maneuver | 61th to 420th s | 181th s |
| σ_r | 2 km | 2 km |
| σ_s | 2 knots | 2 knots |
| σ_θ | 1.5° | 1.5° |
| σ_f | 1 Hz | 1 Hz |
| σ_c | $\pi/\sqrt{5}$ | $\pi/\sqrt{5}$ |
| \tilde{q} (km^2/s^3) | 9×10^{-12} | 9×10^{-12} |
| T | 1 s | 1 s |

4.2. Performance Metrics

We evaluate the performance of the tracking filters using the following performance metrics.

4.2.1. Root Mean Square Error

The position root mean square error (RMSE) at any time index k can be expressed as

$$\text{RMSE}_{\text{pos},k} = \sqrt{\frac{1}{M} \sum_{i=1}^M (\mathcal{X}_{1,k}^{i,t} - \hat{\mathcal{X}}_{1,k}^{i,t})^2 + (\mathcal{X}_{2,k}^{i,t} - \hat{\mathcal{X}}_{2,k}^{i,t})^2},$$

where M is the number of ensembles, and $(\mathcal{X}_{1,k}^{i,t}, \mathcal{X}_{2,k}^{i,t})$ is the position at the k -th time step of i -th ensemble with corresponding estimated value $(\hat{\mathcal{X}}_{1,k}^{i,t}, \hat{\mathcal{X}}_{2,k}^{i,t})$.

4.2.2. Normalised (State) Estimation Error Squared

The normalised (state) estimation error squared (NEES) at the k -th time index can be defined as [12] (p. 234)

$$\text{NEES}_k = (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k})' P_{k|k}^{-1} (\mathcal{X}_k - \hat{\mathcal{X}}_{k|k}). \quad (36)$$

If we assume that the estimators are consistent, and the system is linear with Gaussian noise, then the NEES follows a chi-square distribution with n_x degree of freedom, and

$E[\text{NEES}_k] = n_x$, the dimension of the state. The average value of the NEES (ANEES) for M Monte Carlo (MC) runs can be calculated as [12] (p. 234) [54]

$$\text{ANEES}_k = \frac{1}{n_x M} \sum_{i=1}^M \text{NEES}_k^i. \tag{37}$$

The estimators are called consistent if the $\text{ANEES}_k \in [l_b, u_b]$, where l_b and u_b are the lower and upper bounds of the acceptance interval, respectively. Furthermore, the estimator is considered optimistic [12] (p. 245) if $\text{ANEES}_k > u_b$, because, in this case, the value of $P_{k|k}$ is too small, whereas the estimator is pessimistic [12] (p. 245) when the value of ANEES_k is lower than l_b . The calculation of l_b and u_b can be found in [28,55] and references therein.

4.2.3. Bias Norm

The position bias norm at the k -th time step for M number of MC runs is expressed as [56]

$$\text{Bias norm}_k = \left\| \frac{1}{M} \sum_{i=1}^M \hat{\mathcal{X}}_{p,k}^i - \frac{1}{M} \sum_{i=1}^M \mathcal{X}_{p,k}^i \right\|_2, \tag{38}$$

where $\|\cdot\|_2$ denotes the vector norm, $\mathcal{X}_{p,k} = (\mathcal{X}_{1,k}, \mathcal{X}_{2,k})$ is the true position, and $\hat{\mathcal{X}}_{p,k} = (\hat{\mathcal{X}}_{1,k}, \hat{\mathcal{X}}_{2,k})$ is its posterior estimate.

4.2.4. Track Loss

The terminal error of estimation is defined as

$$e_b = \sqrt{(\mathcal{X}_{1,kmax}^t - \hat{\mathcal{X}}_{1,kmax}^t)^2 + (\mathcal{X}_{2,kmax}^t - \hat{\mathcal{X}}_{2,kmax}^t)^2},$$

where $\mathcal{X}_{1,kmax}^t$ and $\hat{\mathcal{X}}_{1,kmax}^t$ are the true and estimated x-position at the final time step, with similar definitions for these variables with subscript 2, which correspond to the y-position. A track is deemed lost if the terminal error (e_b) is above a threshold, e_T .

4.3. Initialisation of the Filters

For a fair comparison, all the filters are initialised with the same mean and covariance [2] (pp. 115–117), [7]. We initialise the range with a random number $\hat{r} \sim \mathcal{N}(r, \sigma_r^2)$, where r is the true initial range and σ_r is the standard deviation of the range. Similarly, we initialise the first bearing angle to be $\theta_0 \sim \mathcal{N}(\theta, \sigma_\theta^2)$ and target speed $\hat{s} \sim \mathcal{N}(s, \sigma_s^2)$, where θ and s are the true bearing angle and target speed, respectively. σ_θ and σ_s are the standard deviation of the bearing angle and target speed, respectively. Moreover, in initialising the target course, the filters assume that the target moves towards the observer, so the initial target course estimate is calculated by $\hat{c} = \theta_0 + \pi$. As our state vector consists of relative positions and velocities, the initial estimate of it can be expressed as $\hat{\mathcal{X}}_{0|0} = [\hat{r} \sin(\theta_0) \quad \hat{r} \cos(\theta_0) \quad \hat{s} \sin(\hat{c}) - \dot{\mathcal{X}}_{1,0}^0 \quad \hat{s} \cos(\hat{c}) - \dot{\mathcal{X}}_{2,0}^0]^T$, where $(\dot{\mathcal{X}}_{1,0}^0, \dot{\mathcal{X}}_{2,0}^0)$ is the initial velocity of the observer. The initial error covariance can be written as [2] (pp. 116–117)

$$P_{0|0} = \begin{bmatrix} P_{\mathcal{X}\mathcal{X}} & P_{\mathcal{X}\mathcal{Y}} & 0 & 0 \\ P_{\mathcal{Y}\mathcal{X}} & P_{\mathcal{Y}\mathcal{Y}} & 0 & 0 \\ 0 & 0 & P_{\dot{\mathcal{X}}\dot{\mathcal{X}}} & P_{\dot{\mathcal{X}}\dot{\mathcal{Y}}} \\ 0 & 0 & P_{\dot{\mathcal{Y}}\dot{\mathcal{X}}} & P_{\dot{\mathcal{Y}}\dot{\mathcal{Y}}} \end{bmatrix}, \tag{39}$$

where

$$\begin{aligned} P_{\mathcal{X}\mathcal{X}} &= \hat{r}^2 \sigma_\theta^2 \cos^2(\theta_0) + \sigma_r^2 \sin^2(\theta_0), \\ P_{\mathcal{Y}\mathcal{Y}} &= \hat{r}^2 \sigma_\theta^2 \sin^2(\theta_0) + \sigma_r^2 \cos^2(\theta_0), \\ P_{\mathcal{X}\mathcal{Y}} &= P_{\mathcal{Y}\mathcal{X}} = (\sigma_r^2 - \hat{r}^2 \sigma_\theta^2) \sin(\theta_0) \cos(\theta_0), \end{aligned}$$

$$P_{\dot{x}\dot{x}} = \hat{s}^2 \sigma_c^2 \cos^2(\hat{c}) + \sigma_s^2 \sin^2(\hat{c}),$$

$$P_{\dot{y}\dot{y}} = \hat{s}^2 \sigma_c^2 \sin^2(\hat{c}) + \sigma_s^2 \cos^2(\hat{c}),$$

$$P_{\dot{x}\dot{y}} = P_{\dot{y}\dot{x}} = (\sigma_s^2 - \hat{s}^2 \sigma_c^2) \sin(\hat{c}) \cos(\hat{c}).$$

4.4. Performance Comparison

All the filters, namely the orthogonal polynomial filters and deterministic sample point filters, are implemented with BOT and DBT measurements for Scenario I and Scenario II in the following cases: (i) known and time-invariant measurement noise (we call it $R_{k,known}$), (ii) known and time-varying measurement noise (we call it $R_{k,kknown}$), (iii) unknown and time-invariant measurement noise (we denote it $R_{k,unknown}$), (iv) unknown and time-varying measurement noise (we denote it $R_{k,unkknown}$). It is needless to say that we replace the filters with their adaptive counterparts whenever we encounter unknown measurement noise covariance. For the $R_{k,unknown}$ case, we vary R_k from $(1.5^\circ)^2$ to $(4^\circ)^2$ linearly based on the range of target from ownship. Results are summarised in figures and tables showing the RMSE, ANEES, bias norm, track loss, and computational time for various cases. Throughout the simulation, for the SRUKF, we set $\kappa = 1$ as the choice $\kappa = 3 - n_x$ was found to be problematic due to the negative weights. However, such problems did not appear in UO-EKF and so, for this filter, we set $\kappa = -1$. For SRGHF and GHO-EKF, we used three univariate points (i.e., $m = 3$), which resulted in a total of $3^4 = 81$ support points.

4.4.1. Scenario I : Case I—Bearings-Only Tracking

Figure 2a,b show the RMSEs of position and velocity for $R_{k,known}$ obtained from 500 MC runs (excluding track loss cases with $e_T = 500$ m). From these figures, we see that except for the EKF, all the filters provide similar RMSE. To see the filters' consistency, the average NEES (ANEES) obtained from 500 MC runs are plotted in Figure 3a with 95% probability regions, for which the lower bound and upper bound become $l_b = 0.939$ and $u_b = 1.0629$, respectively [28,55]. From this figure, we see that all the filters' ANEES values fall inside the concentration region only by the end of the simulation time. Position bias norms obtained from 500 MC runs are plotted in Figure 3b. From this figure, we see that the bias norms of all filters are sufficiently low, and among them, the SRGHF and GHO-EKF attain the lowest bias norm.

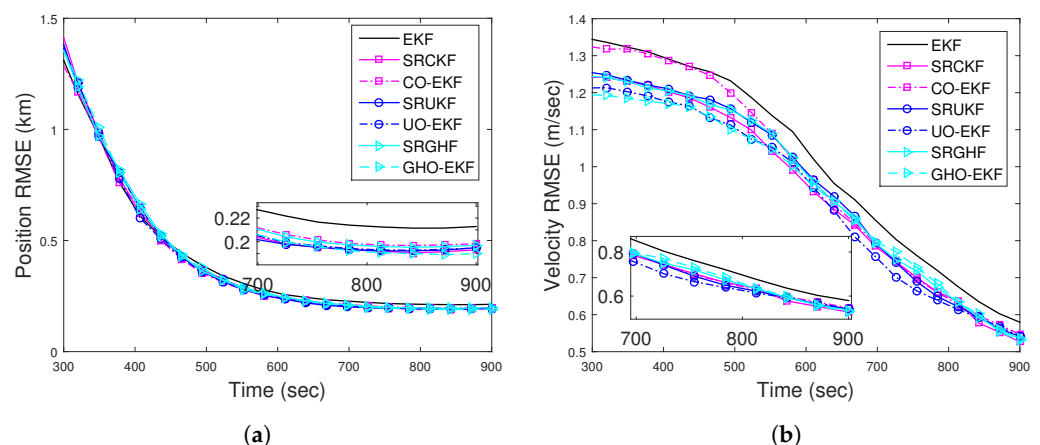


Figure 2. RMSEs of the BOT for Scenario I: (a) position; (b) velocity.

To show the efficacy of the proposed measurement noise adaptation technique, in Figure 4, we plot $R_{unknown}$ and $R_{k,unknown}$ and their estimates obtained from the adaptive GHO-EKF (AD-GHO-EKF). From this figure, we see that the estimated noise covariances converge to their true values. Very similar estimates are obtained from other adaptive polynomial filters and are not plotted in the figure. To observe the effect of unknown R in state estimation (both BOT and DBT cases), position and velocity RMSEs (excluding track loss cases, with error threshold $e_T = 500$ m) obtained from a single representative

filter, say AD-GHO-EKF, are plotted in Figure 5a,b, respectively. Here, we also provide the RMSE results for the DBT case in order to show the improvement in RMSE that we could achieve in comparison to BOT if Doppler shifts are incorporated as measurements. The findings of the result for the DBT case are explicitly discussed in Scenario I: Case II. The results are compared with GHO-EKF for both the time-varying and time-invariant unknown measurement noise covariance. From these figures, we see that the GHO-EKF provides lower RMSEs compared to AD-GHO-EKF in this situation. Furthermore, the RMSE for the R_{unknown} case is slightly less than $R_{k,\text{unknown}}$, as expected.

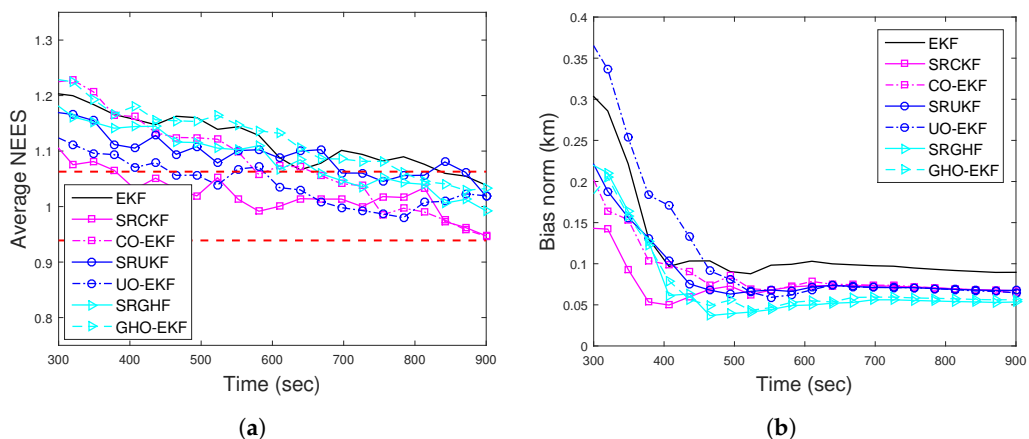


Figure 3. (a) ANEES; (b) position bias norm of the BOT for Scenario I.

ANEES metrics are calculated for various adaptive filters for $e_T \in \{200 \text{ m}, 500 \text{ m}\}$. For $e_T = 500 \text{ m}$, none of the filters' ANEES is inside the concentration region. In Figure 6a, the ANEES of various adaptive filters are plotted from 500 MC runs $e_T = 200 \text{ m}$, and for this case, we notice that the ANEES of the GHO-EKF (DBT, R_{known}) is consistently inside the probability region, whereas the same obtained from other filters fall inside the region only after 650 s. The poor ANEES results are mainly due to low observability and unknown time-varying measurement noise covariance. Furthermore, we provide the position bias norm of GHO-EKF and AD-GHO-EKF for unknown measurement noise covariance obtained from 500 MC runs in Figure 6b. From this figure, we observe that GHO-EKF (DBT) has the lowest bias norm, whereas the AD-GHO-EKF for $R_{k,\text{unknown}}$ attains the highest.

Next, we study the effect of sampling time on the accuracy of estimation. In Figure 7, we plot the position and velocity RMSE (excluding the track divergence with $e_T = 500 \text{ m}$) of AD-GHO-EKF obtained from 500 MC runs for different sampling times, T , ranging from 1 s to 60 s. From these figures, we observe that for lower sampling time T , the AD-GHO-EKF exhibits better performance, as expected. However, it is to be noted that the sampling time of the sensor cannot be very small as certain time is required to accumulate the energy received in order to measure the bearing angle.

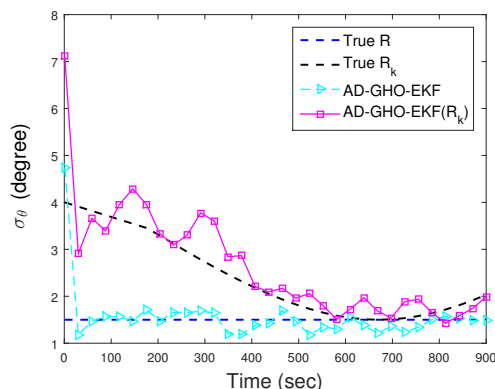


Figure 4. True and estimated R (both time-varying and time-invariant) for Scenario I.

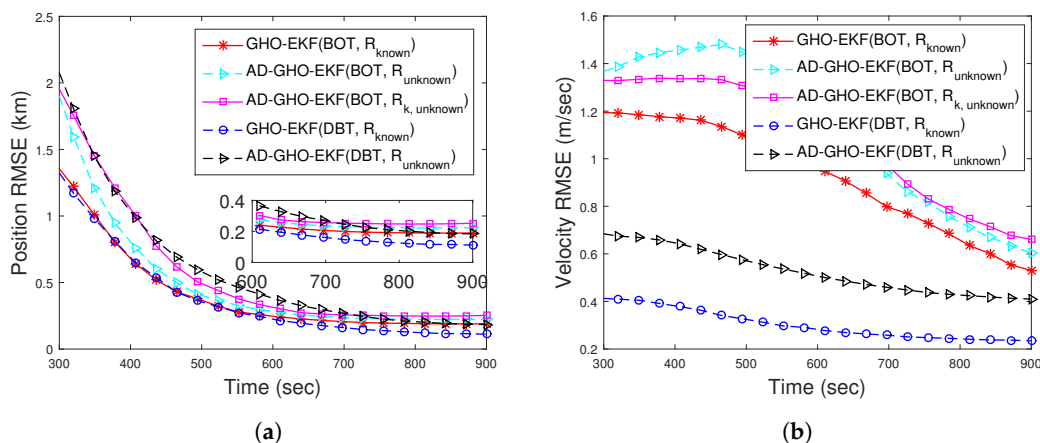


Figure 5. RMSEs of the BOT and DBT for Scenario I for unknown R .

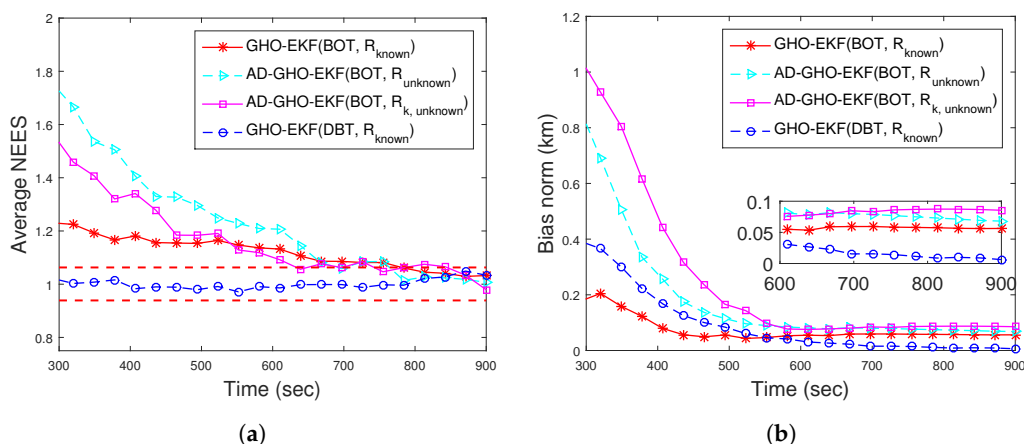


Figure 6. (a) ANEES and (b) position bias norm for BOT and DBT for Scenario I.

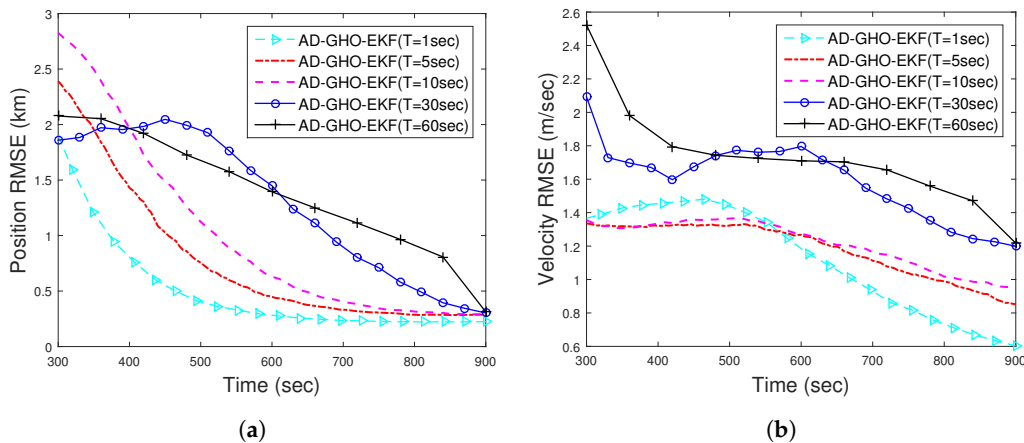


Figure 7. RMSEs of the (a) position; (b) velocity of the BOT for varying T of $R_{unknown}$ for Scenario I.

4.4.2. Scenario I : Case II—Doppler-Bearing Tracking

Figure 8a,b show the position and velocity RMSEs for DBT obtained from square root filters and orthogonal polynomial filters out of 500 MC runs. They are calculated with known $R = (1.5^\circ)^2$, excluding track divergence cases. All the RMSEs are comparable and lower than the RMSE obtained from BOT. The ANEES with the 500 m track loss threshold are within the provided probability region. The position bias norms of the filters approach zero and are much below that of BOT. From these figures (Figures 8 and 9), we can conclude that the filters with DBT measurements track more effectively than those with BOT measurements.

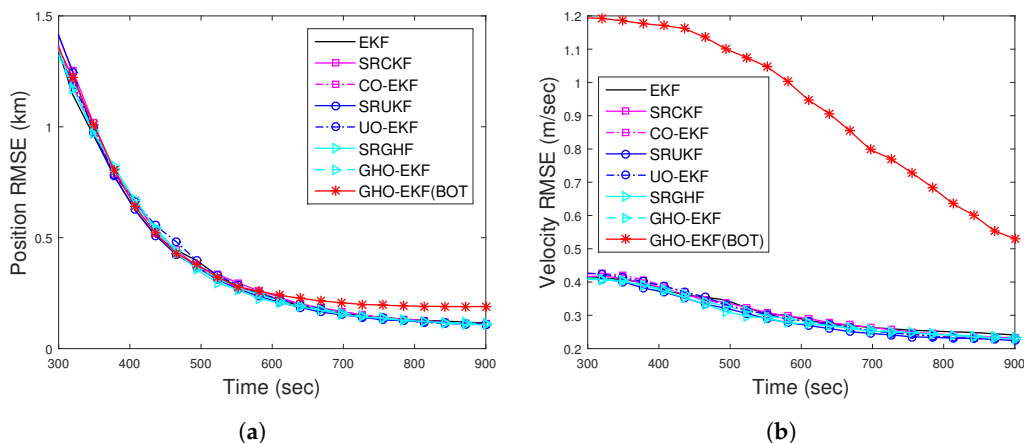


Figure 8. RMSEs of the (a) position; (b) velocity of the DBT for Scenario I.

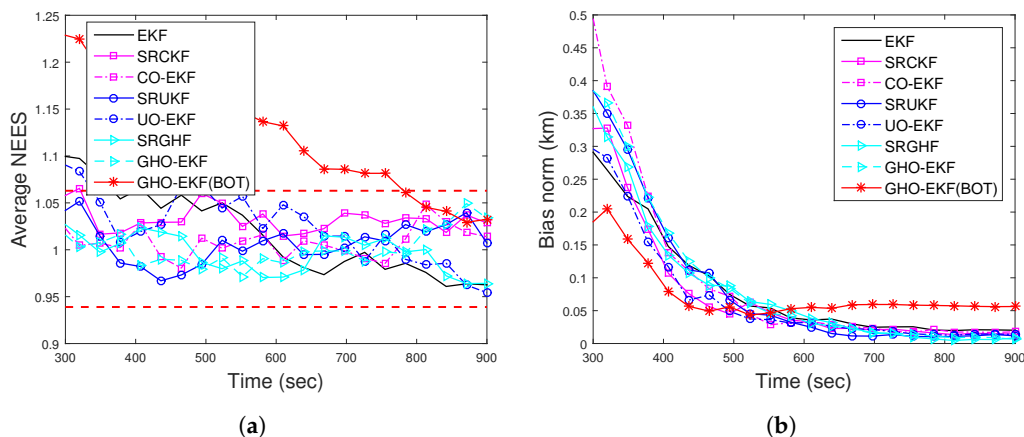


Figure 9. (a) ANEES; (b) position bias norm of the DBT for Scenario I.

The efficacy of the proposed adaptive estimator is also checked for R_{unknown} using DBT measurements. We provide the position and velocity RMSEs of GHO-EKF and AD-GHO-EKF (R_{unknown}) obtained from 500 MC runs (excluding the track divergence cases, $e_T = 500$ m) in Figure 5. From this figure, we see that GHO-EKF attains lower RMSEs than AD-GHO-EKF. We also see that the incorporation of the Doppler-shifted frequency improves the filters’ estimation accuracy.

Now, we compare the performance of filters (EKF, orthogonal polynomial filters, deterministic square root sample point filters, and their adaptive counterparts) for BOT and DBT in terms of percentage track loss obtained from 10,000 MC runs, presented in Table 3. The track divergence bound is set to $e_T = 500$ m. From this table, we see that the filters with DBT measurements have lower track loss than those with BOT. As expected, filters with known R show better performance than those with unknown R , i.e., adaptive filters. The orthogonal polynomial filters provide almost similar track loss as their square root counterparts. We also observe that the filters with time-varying R_k (known and unknown) show higher track loss than time-invariant R , and these results are not provided. Furthermore, we observe that the track loss count of all the filters increases with the increase in sampling time. This happens due to the fact that with the increase in sampling time, the observability of the system decreases, which further deteriorates the filters’ performance.

Table 3. Percentage track loss of different filters of the BOT and DBT for both scenarios.

| Filter | Scenario I | | Scenario II | |
|------------|------------|------|-------------|------|
| | BOT | DBT | BOT | DBT |
| EKF | 2.30 | 0 | 6.73 | 0 |
| AD-EKF | 9.86 | 2.27 | 17.35 | 2.09 |
| SRCKF | 2.25 | 0 | 5.72 | 0 |
| CO-EKF | 2.03 | 0 | 6.17 | 0 |
| AD-SRCKF | 8.28 | 1.3 | 16.51 | 0.91 |
| AD-CO-EKF | 8.78 | 1.35 | 16.23 | 1.01 |
| SRUKF | 2.16 | 0 | 6.12 | 0 |
| UO-EKF | 2.22 | 0 | 6.26 | 0 |
| AD-SRUKF | 8.48 | 1.41 | 17.73 | 1.05 |
| AD-UO-EKF | 8.69 | 1.38 | 15.61 | 0.95 |
| SRGHF | 1.86 | 0 | 6.35 | 0 |
| GHO-EKF | 2.24 | 0 | 6.27 | 0 |
| AD-SRGHF | 8.20 | 0.72 | 16.96 | 0.74 |
| AD-GHO-EKF | 8.35 | 0.80 | 16.41 | 0.80 |

4.4.3. Scenario II : Case I—Bearings-Only Tracking

For BOT, the position and velocity RMSE of different filters for R_{known} from 500 MC runs are plotted in Figure 10a,b, respectively. From these figures, we see that except for the EKF, all other filters attain similar RMSEs. Figure 11a shows that except EKF, the ANEES of all filters (with $e_T = 500$ m) fall within the provided probability region. The position bias norm plot (Figure 11b) shows that all the filters attain a sufficiently low bias norm.

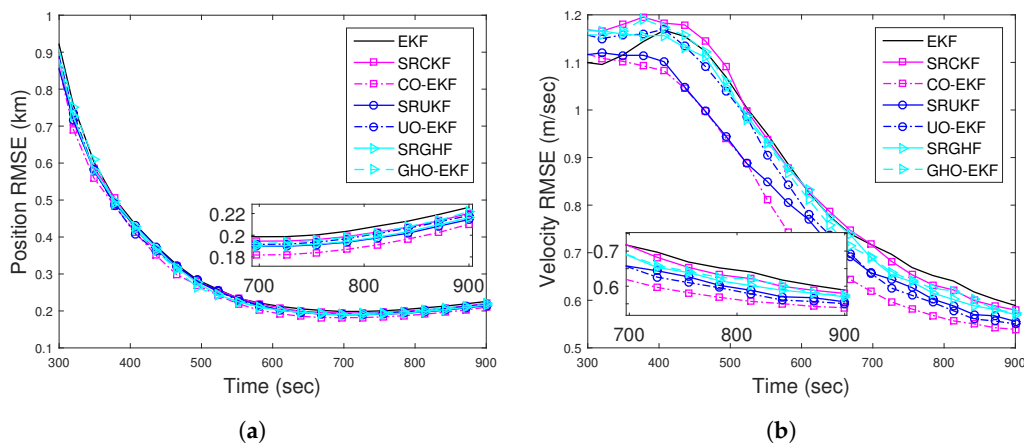


Figure 10. (a) Position; (b) velocity RMSE of the BOT for Scenario II.

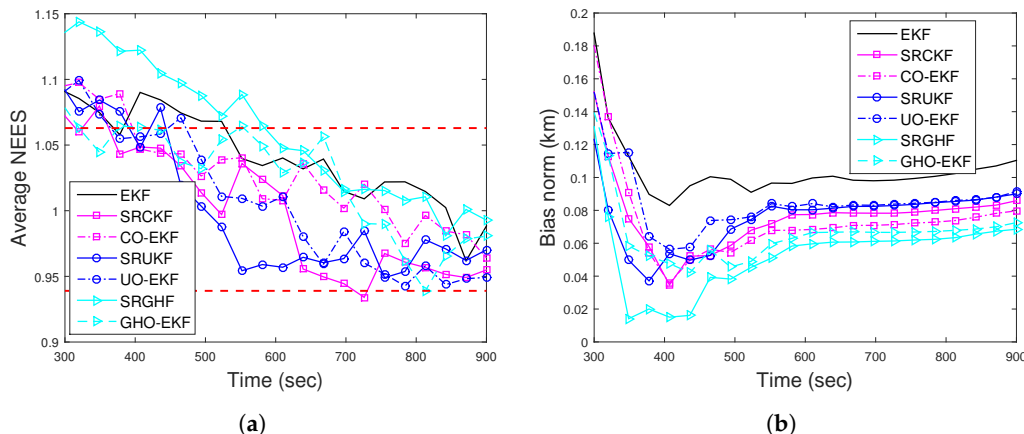


Figure 11. (a) ANEES; (b) position bias norm of the BOT for Scenario II.

4.4.4. Scenario II : Case II—Doppler-Bearing Tracking

For DBT, the position and velocity RMSE of different filters for R_{known} obtained from 500 MC runs are plotted in Figure 12a,b, respectively. From this figure, we see that the RMSEs of all filters are comparable and lower than the RMSEs obtained from BOT.

The estimation accuracy of the proposed AD-GHO-EKF for R_{unknown} for BOT and DBT is examined in Figure 13 in terms RMSEs (obtained from 500 MC runs). For comparison, we also include the RMSEs of GHO-EKF for R_{known} . From this figure, it can be seen that the GHO-EKF provides better estimation accuracy than the AD-GHO-EKF for both cases of measurements. We also see that the RMSE with DBT measurements is lower than the RMSE obtained from the BOT.

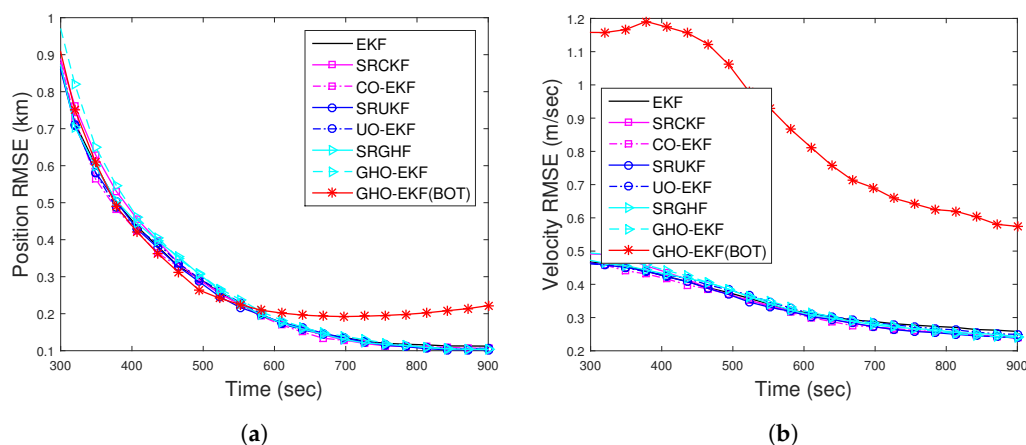


Figure 12. RMSEs of the (a) position; (b) velocity of the DBT for Scenario II.

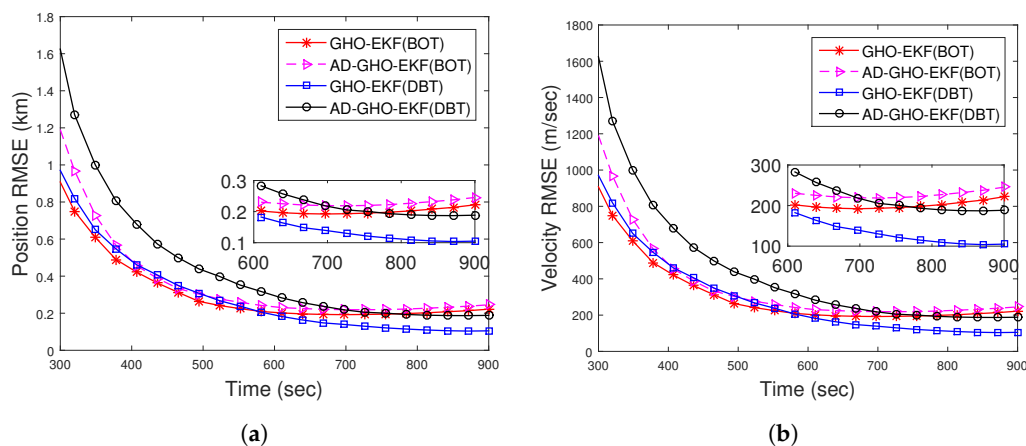


Figure 13. RMSEs of the (a) position; (b) velocity of the BOT and DBT for Scenario II for unknown fixed R .

Now, we calculate the percentage track loss of the EKF, square root filters, orthogonal polynomial filters, and their adaptive counterparts out of 10,000 MC runs for BOT and DBT, the results of which are presented in Table 3. The track loss bound is set to be $e_T = 500$ m. From this table, it can be seen that the filters with DBT measurements perform better than those with BOT. We also see that for known measurement noise covariance, the filters have lower track loss. It can be also observed that the orthogonal polynomial filters have almost similar track loss to the square root filters.

Finally, we compare the filtering performance in terms of their execution time. The relative execution times of all filters with respect to the EKF for both BOT and DBT are reported in Table 4. From this table, we see that the execution times of the GHO-EKF and SRGHF are almost three to four times those of the EKF. The CO-EKF and UO-EKF take nearly 50% more execution time than the EKF, whereas the computation time of the SRCKF and SRUKF is nearly twice that of the EKF. We also observe that for both BOT and

DBT measurements, the orthogonal polynomial filters have lower computation times than their respective square root deterministic sample filters. As expected, the adaptation of measurement noise covariance further increases the execution time of the filters.

Table 4. Relative execution times of the different filters for BOT and DBT.

| Filter | BOT | | DBT | |
|---------|----------|----------|----------|----------|
| | Ordinary | Adaptive | Ordinary | Adaptive |
| EKF | 1.00 | 2.14 | 1.28 | 2.40 |
| SRCKF | 1.84 | 3.72 | 2.14 | 4.02 |
| CO-EKF | 1.51 | 2.70 | 1.95 | 2.92 |
| SRUKF | 1.88 | 3.80 | 2.28 | 4.09 |
| UO-EKF | 1.53 | 2.78 | 2.03 | 2.99 |
| SRGHF | 6.53 | 10.35 | 10.52 | 14.13 |
| GHO-EKF | 4.16 | 5.97 | 5.64 | 7.75 |

To assess how our proposed algorithms work in the presence of a target maneuver, we consider a target that is maneuvering with the constant turn rate (CT) model, as described in Section 2.1.2, with known $\omega = -0.015^\circ/s$, which is shown in Figure 1a. Initially, we assume that the estimators know the target maneuvering model and check the performance of all the filters. We see that all the filters perform with similar accuracy, as described in Figures 2–13. Further, to study the robustness of these filters, we assume that the target is following a maneuver as described above, but the estimators are unaware of this dynamics and they use the constant velocity model to estimate the position and velocity of the target. To cope with such model mismatch, the process noise intensity of the filtering algorithm is increased to $100\tilde{q}$. The target and all the estimators are initialised as described in Section 4.3. Initially, we experiment with a 500 m track loss threshold, i.e., $e_T = 500$ m, and it has been observed that the final track error in all the filters in this model mismatched case was above this threshold in most of the runs. Further, we calculate the percentage of track loss for all the filters considering $e_T = 1500$ m and list them in Table 5. From this table, we note that DBT shows much less track loss than BOT and the adaptive filters in DBT show less than 1% track loss, while their non-adaptive counterparts exhibit zero track loss. However, for BOT, the percentage of track loss obtained from adaptive filters is almost double that of their non-adaptive counterparts. In Figure 14, we plot the RMSE of position and velocity obtained from the adaptive and ordinary GHO-EKF for both the bearings-only and Doppler-bearing tracking. Comparing Figure 5 with Figure 14, it has been observed that the position RMSE for the model mismatch case is converging to a slightly higher value than that of the constant velocity scenario. Further, the RMSE obtained from DBT is lower compared to BOT, and the same obtained with GHO-EKF is lower than its adaptive counterpart.

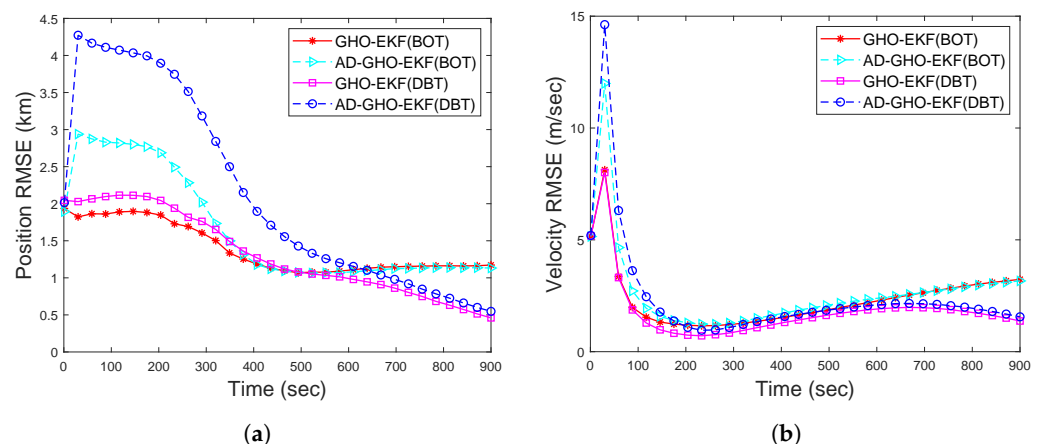


Figure 14. RMSEs of the (a) position; (b) velocity of the BOT and DBT for maneuvering target.

Table 5. Percentage of track loss for BOT and DBT when model mismatch occurs.

| Filter | BOT | | DBT | |
|---------|----------|----------|----------|----------|
| | Ordinary | Adaptive | Ordinary | Adaptive |
| EKF | 13.34 | 29.72 | 0 | 0.35 |
| SRCKF | 12.30 | 26.95 | 0 | 0.25 |
| CO-EKF | 12.40 | 26.94 | 0 | 0.26 |
| SRUKF | 12.62 | 26.80 | 0 | 0.28 |
| UO-EKF | 12.04 | 27.08 | 0 | 0.25 |
| SRGHF | 11.55 | 26.57 | 0 | 0.20 |
| GHO-EKF | 11.53 | 26.62 | 0 | 0.20 |

5. Discussion and Conclusions

In this paper, we have proposed an adaptive orthogonal polynomial-based filter that can estimate the target trajectories along with the measurement noise error covariance. It is based on the linearisation of the measurement function using the first-order orthogonal polynomial approximation, and the coefficients of these polynomials are evaluated using the numerical integration technique. The proposed method is used to estimate a constant velocity target trajectory for two different tracking scenarios, each with bearings-only and Doppler-bearing measurements. The estimation accuracy of the filters is studied in terms of RMSE. We also use the percentage track loss, computation time, average NEES, and bias norm to compare the performance of the filters. The robustness of the proposed algorithm is studied when there is a mismatch between the target and the filter-assumed dynamics. From the simulation results, it can be deduced that the proposed filters consistently outperform the Taylor series-based EKF and provide comparable performance with their square root counterparts of the deterministic sample point filters with a reduced computational cost. Moreover, the proposed filters with CV model target dynamics can estimate the track of a lightly maneuvering target with acceptable accuracy. Target motion analysis with unknown sensor misalignment remains under the scope of future works.

Author Contributions: Conceptualisation, K.K., S.B. and S.A.; methodology, K.K., S.B. and S.A.; software, K.K.; validation, K.K., S.B. and S.A.; writing—original draft preparation, K.K.; writing—review and editing, S.B. and S.A.; supervision, S.B. and S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable as all the results are generated with simulated data.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Arulampalam, S.; Ristic, B. Comparison of the particle filter with range-parameterized and modified polar EKFs for angle-only tracking. In Proceedings of the AeroSense 2000 Signal and Data Processing of Small Targets 2000, Orlando, FL, USA, 24–28 April 2000; Volume 4048, pp. 288–299.
- Ristic, B.; Arulampalam, S.; Gordon, N. *Beyond the Kalman Filter: Particle Filters For Tracking Applications*; Artech House: London, UK, 2003.
- Ho, K.; Chan, Y.T. An asymptotically unbiased estimator for bearings-only and Doppler-bearing target motion analysis. *IEEE Trans. Signal Process.* **2006**, *54*, 809–822. [[CrossRef](#)]
- Guo, Y.; Tharmarasa, R.; Rajan, S.; Song, T.L.; Kirubarajan, T. Passive tracking in heavy clutter with sensor location uncertainty. *IEEE Trans. Aerosp. Electron. Syst.* **2016**, *52*, 1536–1554. [[CrossRef](#)]
- Radhakrishnan, R.; Bhaumik, S.; Tomar, N.K. Gaussian sum shifted Rayleigh filter for underwater bearings-only target tracking problems. *IEEE J. Ocean. Eng.* **2018**, *44*, 492–501. [[CrossRef](#)]
- Luo, J.; Han, Y.; Fan, L. Underwater acoustic target tracking: A review. *Sensors* **2018**, *18*, 112. [[CrossRef](#)] [[PubMed](#)]
- Leong, P.H.; Arulampalam, S.; Lamaheewa, T.A.; Abhayapala, T.D. A Gaussian-sum based cubature Kalman filter for bearings-only tracking. *IEEE Trans. Aerosp. Electron. Syst.* **2013**, *49*, 1161–1176. [[CrossRef](#)]
- Yang, L.; Sun, M.; Ho, K. Doppler-bearing tracking in the presence of observer location error. *IEEE Trans. Signal Process.* **2008**, *56*, 4082–4087. [[CrossRef](#)]

9. Rao, S.K. Doppler-bearing passive target tracking using a parameterized unscented Kalman filter. *IETE J. Res.* **2010**, *56*, 69–75. [[CrossRef](#)]
10. Li, X.; Zhao, C.; Yu, J.; Wei, W. Underwater bearing-only and bearing-Doppler target tracking based on square root unscented Kalman filter. *Entropy* **2019**, *21*, 740. [[CrossRef](#)]
11. Borisov, A.; Bosov, A.; Miller, B.; Miller, G. Passive underwater target tracking: Conditionally minimax nonlinear filtering with bearing-Doppler observations. *Sensors* **2020**, *20*, 2257. [[CrossRef](#)]
12. Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*; John Wiley & Sons: Hoboken, NJ, USA, 2004.
13. Anderson, B.D.; Moore, J.B. *Optimal Filtering*; Courier Corporation: North Chelmsford, MA, USA, 2012.
14. Karaman, S. Fixed point smoothing algorithm to the torpedo tracking problem. Master's Thesis, Naval Postgraduate School Monterey, Monterey, CA, USA, 1986.
15. Karlsson, R. *Various Topics on Angle-Only Tracking Using Particle Filters*; Linköping University: Linköping, Sweden, 2002.
16. Karlsson, R. Simulation based methods for target tracking. Ph.D. Thesis, Linköping University, Linköping, Sweden, 2002.
17. Kumar, D.R.; Rao, S.K.; Raju, K.P. A novel estimation algorithm for torpedo tracking in undersea environment. *J. Cent. South Univ.* **2019**, *26*, 673–683. [[CrossRef](#)]
18. Reif, K.; Gunther, S.; Yaz, E.; Unbehauen, R. Stochastic stability of the discrete-time extended Kalman filter. *IEEE Trans. Autom. Control* **1999**, *44*, 714–728. [[CrossRef](#)]
19. Julier, S.J.; Uhlmann, J.K. New extension of the Kalman filter to nonlinear systems. In Proceedings of the AeroSense '97, Signal Processing, Sensor Fusion, and Target Recognition VI, Orlando, FL, USA, 21–25 April 1991; Volume 3068, pp. 182–193.
20. Julier, S.; Uhlmann, J.; Durrant-Whyte, H.F. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Trans. Autom. Control* **2000**, *45*, 477–482. [[CrossRef](#)]
21. Ito, K.; Xiong, K. Gaussian filters for nonlinear filtering problems. *IEEE Trans. Autom. Control* **2000**, *45*, 910–927. [[CrossRef](#)]
22. Radhakrishnan, R.; Singh, A.K.; Bhaumik, S.; Tomar, N.K. Multiple sparse-grid Gauss–Hermite filtering. *Appl. Math. Model.* **2016**, *40*, 4441–4450. [[CrossRef](#)]
23. Arasaratnam, I.; Haykin, S. Cubature Kalman filters. *IEEE Trans. Autom. Control* **2009**, *54*, 1254–1269. [[CrossRef](#)]
24. Bhaumik, S.; Swati. Cubature quadrature Kalman filter. *IET Signal Process.* **2013**, *7*, 533–541. [[CrossRef](#)]
25. Van Der Merwe, R.; Wan, E.A. The square-root unscented Kalman filter for state and parameter-estimation. In Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 01CH37221), Salt Lake City, UT, USA, 7–11 May 2001; Volume 6, pp. 3461–3464.
26. Arasaratnam, I.; Haykin, S. Square-root quadrature Kalman filtering. *IEEE Trans. Signal Process.* **2008**, *56*, 2589–2593. [[CrossRef](#)]
27. Bhaumik, S.; Swati. Square-root cubature-quadrature Kalman filter. *Asian J. Control* **2014**, *16*, 617–622. [[CrossRef](#)]
28. Kumar, K.; Bhaumik, S.; Date, P. Extended Kalman filter using orthogonal polynomials. *IEEE Access* **2021**, *9*, 59675–59691. [[CrossRef](#)]
29. Li, X.R.; Jilkov, V.P. Survey of maneuvering target tracking. Part I: dynamic models. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1333–1364.
30. Hou, X.; Yang, L.; Yang, Y.; Zhou, J.; Qiao, G. Bearing-only underwater uncooperative target tracking for non-Gaussian environment using fast particle filter. *IET Radar Sonar Navig.* **2022**, *16*, 501–514. [[CrossRef](#)]
31. Abramowitz, M.; Stegun, I.A. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*; US Government Printing Office: Washington, DC, USA, 1964; Volume 55.
32. Holmquist, B. The d-variate vector Hermite polynomial of order k. *Linear Algebra Its Appl.* **1996**, *237*, 155–190. [[CrossRef](#)]
33. Buet-Golfouse, F. A multinomial theorem for Hermite polynomials and financial applications. *Appl. Math.* **2015**, *6*, 1017. [[CrossRef](#)]
34. Wu, Y.; Hu, D.; Wu, M.; Hu, X. A numerical-integration perspective on Gaussian filters. *IEEE Trans. Signal Process.* **2006**, *54*, 2910–2921. [[CrossRef](#)]
35. Jia, B.; Xin, M.; Cheng, Y. High-degree cubature Kalman filter. *Automatica* **2013**, *49*, 510–518. [[CrossRef](#)]
36. Singh, A.K.; Bhaumik, S. Higher degree cubature quadrature Kalman filter. *Int. J. Control Autom. Syst.* **2015**, *13*, 1097–1105. [[CrossRef](#)]
37. Arasaratnam, I.; Haykin, S.; Elliott, R.J. Discrete-time nonlinear filtering algorithms using Gauss-Hermite quadrature. *Proc. IEEE* **2007**, *95*, 953–977. [[CrossRef](#)]
38. Chalasani, G.; Bhaumik, S. Bearing only tracking using Gauss-Hermite filter. In Proceedings of the 2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA), Singapore, 18–20 July 2012; pp. 1549–1554.
39. Zhang, X.; Ying, W.; Yang, P.; Sun, M. Parameter estimation of underwater impulsive noise with the class B model. *IET Radar Sonar Navig.* **2020**, *14*, 1055–1060. [[CrossRef](#)]
40. Mahmood, A.; Chitre, M. Modeling colored impulsive noise by Markov chains and alpha-stable processes. In Proceedings of the OCEANS 2015-Genova, Genova, Italy, 18–21 May 2015; pp. 1–7.
41. Zhou, H.; Xu, L.; Chen, W.; Guo, K.; Shen, F.; Guo, Z. A novel robust filtering strategy for systems with non-Gaussian noises. *AEU Int. J. Electron. Commun.* **2018**, *97*, 154–164. [[CrossRef](#)]
42. Mohamed, A.; Schwarz, K. Adaptive Kalman filtering for INS/GPS. *J. Geod.* **1999**, *73*, 193–203. [[CrossRef](#)]
43. Wang, J. Stochastic modeling for real-time kinematic GPS/GLONASS positioning. *Navigation* **1999**, *46*, 297–305. [[CrossRef](#)]

44. Almagbile, A.; Wang, J.; Ding, W. Evaluating the performances of adaptive Kalman filter methods in GPS/INS integration. *J. Glob. Position. Syst.* **2010**, *9*, 33–40. [[CrossRef](#)]
45. Soken, H.E.; Sakai, S.I. Adaptive tuning of the unscented Kalman filter for satellite attitude estimation. *J. Aerosp. Eng.* **2015**, *28*, 04014088. [[CrossRef](#)]
46. Dey, A.; Sadhu, S.; Ghoshal, T.K. Adaptive Gauss–Hermite filter for non-linear systems with unknown measurement noise covariance. *IET Sci. Meas. Technol.* **2015**, *9*, 1007–1015. [[CrossRef](#)]
47. Huang, Y.; Zhang, Y.; Wu, Z.; Li, N.; Chambers, J. A novel adaptive Kalman filter with inaccurate process and measurement noise covariance matrices. *IEEE Trans. Autom. Control* **2017**, *63*, 594–601. [[CrossRef](#)]
48. Zhang, L.; Sidoti, D.; Bienkowski, A.; Pattipati, K.R.; Bar-Shalom, Y.; Kleinman, D.L. On the identification of noise covariances and adaptive Kalman filtering: a new look at a 50 year-old problem. *IEEE Access* **2020**, *8*, 59362–59388. [[CrossRef](#)]
49. Zhou, Y.; Zhang, C.; Zhang, Y.; Zhang, J. A new adaptive square-root unscented Kalman filter for nonlinear systems with additive noise. *Int. J. Aerosp. Eng.* **2015**, *2015*, 381478. [[CrossRef](#)]
50. Zhang, Z.; Zheng, L.; Li, Y.; Wu, H.; Liang, Y.; Qiao, X. Correction adaptive square-root cubature Kalman filter with application to autonomous vehicle target tracking. *Meas. Sci. Technol.* **2021**, *32*, 115101. [[CrossRef](#)]
51. Ristic, B.; Wang, X.; Arulampalam, S. Target motion analysis with unknown measurement noise variance. In Proceedings of the 2017 20th International Conference on Information Fusion (Fusion), Xi'an, China, 10–13 July 2017; pp. 1–8.
52. Ristic, B.; Arulampalam, S.; Wang, X. Measurement variance ignorant target motion analysis. *Inf. Fusion* **2018**, *43*, 27–32. [[CrossRef](#)]
53. Karlsson, R.; Schon, T.; Gustafsson, F. Complexity analysis of the marginalized particle filter. *IEEE Trans. Signal Process.* **2005**, *53*, 4408–4411. [[CrossRef](#)]
54. Li, X.R.; Zhao, Z.; Jilkov, V.P. Estimator's credibility and its measures. In Proceedings of the IFAC 15th World Congress, Barcelona, Spain, 21–26 July 2002.
55. Li, X.R.; Zhao, Z.; Jilkov, V.P. Practical measures and test for credibility of an estimator. In *Proceedings Workshop on Estimation, Tracking, and Fusion: A Tribute to Yaakov Bar-Shalom*; Citeseer: Princeton, NJ, USA, 2001; pp. 481–495.
56. Arulampalam, S.; Ristic, B.; Kirubarajan, T. Analysis of propagation delay effects on bearings-only fusion of heterogeneous sensors. *IEEE Trans. Signal Process.* **2021**, *69*, 6488–6503. [[CrossRef](#)]