

## Article

# Bearings-Only Target Tracking with an Unbiased Pseudo-Linear Kalman Filter

Zihao Huang <sup>1,2</sup> , Shijin Chen <sup>1,2</sup>, Chengpeng Hao <sup>1,\*</sup>  and Danilo Orlando <sup>3</sup>

<sup>1</sup> Institute of Acoustics, Chinese Academy of Sciences, Beijing 100190, China; huangzihao19@mails.ucas.ac.cn (Z.H.); chenshijin19@mails.ucas.ac.cn (S.C.)

<sup>2</sup> School of Electronic, Electrical and Communication, University of Chinese Academy of Sciences, Beijing 100864, China

<sup>3</sup> Department of Engineering, University of Niccolò Cusano, via Don Carlo Gnocchi 3, 00166 Rome, Italy; danilo.orlando@unicusano.it

\* Correspondence: haochengp@mail.ioa.ac.cn

**Abstract:** In bearings-only target tracking, the pseudo-linear Kalman filter (PLKF) attracts much attention because of its stability and its low computational burden. However, the PLKF's measurement vector and the pseudo-linear noise are correlated, which makes it suffer from bias problems. Although the bias-compensated PLKF (BC-PLKF) and the instrumental variable-based PLKF (IV-PLKF) can eliminate the bias, they only work well when the target behaves with non-maneuvring movement. To extend the PLKF to the manoeuvring target tracking scenario, an unbiased PLKF (UB-PLKF) algorithm, which splits the noise away from the measurement vector directly, is proposed. Based on the results of the UB-PLKF, we also propose its velocity-constrained version (VC-PLKF) to further improve the performance. Simulations show that the UB-PLKF and VC-PLKF outperform the BC-PLKF and IV-PLKF both in non-maneuvring and manoeuvring scenarios.

**Keywords:** bearings-only tracking; pseudo-linear Kalman filter; norm-constrained Kalman filter



**Citation:** Huang, Z.; Chen, S.; Hao, C.; Orlando, D. Bearings-Only Target Tracking with an Unbiased Pseudo-Linear Kalman Filter. *Remote Sens.* **2021**, *13*, 2915. <https://doi.org/10.3390/rs13152915>

Academic Editor: Andrzej Stateczny

Received: 18 May 2021

Accepted: 10 July 2021

Published: 24 July 2021

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



**Copyright:** © 2021 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

Bearings-only tracking (BOT) comprises estimating a target's state from bearing measurements received by an observer [1]. It plays an important role in both military and civil applications, for example, underwater surveillance [2,3], 3-D passive target tracking [4–7] and UAV (unmanned aerial vehicle) path planning [8–11]. However, the nonlinear relationship between the bearing measurements and the target state vector makes BOT a typical nonlinear filtering problem [12].

There are two common ways to deal with the nonlinearity in the BOT problem. The first way is to choose the nonlinear Bayesian filtering algorithms. Early approaches are based on the extended Kalman filter and its improved versions [1,13,14]. However, the extended Kalman filter uses the first-order Taylor expansion to replace the corresponding nonlinear function, which will certainly lead to truncation errors. Other more sophisticated Kalman filters, such as the unscented Kalman filter [15] and the cubature Kalman filter [16], applied to the BOT problem, can be found in [17,18]. Another important approach is based upon particle filtering (see, for instance, [19,20]). However, the high computational requirements of the particle filter would involve high-performance expensive hardware that discourages its implementation in BOT applications. It is important to also mention here the non-Bayesian-based shadowing filter that, recently, has been applied to the BOT problem [21]. However, such a filter is a kind of gradient descent algorithm, hence, it is highly sensitive to the step size as well as the computation of the Jacobian matrix [22].

Another way to cope with the nonlinearity in BOT is to convert it into linear form, and the most common method used is the pseudo-linear Kalman filter (PLKF) [23]. It has lower computational complexity and is more robust compared with the nonlinear Bayesian

filtering algorithms introduced above [24], and no other *prior* parameters are required compared with the shadowing filter. These advantages make the PLKF an attractive method for engineering. However, unlike the traditional Kalman filter, the PLKF measurement vector contains the bearing noise, which makes it correlated with the pseudo-linear noise. This correlation leads to bias problems for the PLKF. To improve the PLKF performance, the instantaneous bias of PLKF is analysed in [25], and two improved PLKF algorithms, the bias-compensated PLKF (BC-PLKF) and the instrumental variable-based PLKF (IV-PLKF), are proposed. The BC-PLKF subtracts the approximated bias from the estimate of the PLKF to reduce the bias. The IV-PLKF utilizes the results of the BC-PLKF to construct a noise-free measurement vector, and its asymptotic unbiasedness has been proved in [25,26]. Although BC-PLKF and IV-PLKF effectiveness has been verified in [25] when the target moves without manoeuvring, their performance in the manoeuvring target tracking scenario may be unsatisfactory. It is not surprising because the measurement vector and the noise are still correlated in BC-PLKF, and its performance cannot be guaranteed by only subtracting the bias when the target manoeuvres. IV-PLKF also suffers from this problem because it uses the results of the BC-PLKF as its input parameters.

To extend the PLKF algorithm to the case of the BOT problem with a manoeuvring target and at the same time make an excellent compromise between computational load and accuracy, we first propose an unbiased PLKF (UB-PLKF) algorithm which splits the bearing noise away from the measurement vector directly to eliminate their correlation, thus avoiding the bias compensation operation in BC-PLKF. The unbiasedness makes the UB-PLKF more adaptable to the abrupt changes in the target state and ensures the UB-PLKF's convergence when the target manoeuvres. Then, for further improving its performance, the target velocity range information is incorporated into the UB-PLKF to construct the velocity-constrained PLKF (VC-PLKF) based on [27]. Incorporating some *prior* information about the target state into the Kalman filter to construct the state-constrained Kalman filter [28] is common; however, this method is rarely seen in PLKF. Although [29–31] have introduced the constraints into BOT, they are all based on the batch algorithms which can hardly handle the manoeuvring target tracking problem. To the best of the authors' knowledge, this is the first time that a velocity-constrained PLKF has been proposed. Simulations will verify the effectiveness of the UB-PLKF and VC-PLKF both in non-manoeuving and manoeuvring target tracking scenarios.

This paper is organised as follows: In Section 2, we first formulate the BOT problem to be solved and generally review the PLKF, BC-PLKF, and IV-PLKF, then we propose the UB-PLKF and VC-PLKF. Section 3 presents the simulation results to verify the effectiveness of the proposed algorithms.

### Notations

Throughout the paper, vectors and matrices are denoted by boldface lower-case and upper-case letters, respectively.  $\mathbb{E}\{\cdot\}$  and  $\text{tr}\{\cdot\}$  denote the expectation operator and trace operator, respectively;  $I_{n \times n}$  denotes the  $n \times n$  identity matrix;  $\mathbf{0}_{n \times m}$  denotes the  $n \times m$  zero matrix; the superscript "BC", "IV", "UB" and "VC" denote, respectively, the vector or matrix in BC-PLKF, IV-PLKF, UB-PLKF and VC-PLKF, and any vector or matrix with this kind of superscript has the same dimension as its original form in PLKF; the superscript "T" denotes the transpose operation of a vector or matrix; the superscript " $-1$ " denotes the inverse operation of a matrix; the superscript "+" means that the vector or matrix belongs to the *posterior* estimate, and " $-$ " means that the vector or matrix belongs to the *predictive* estimate;  $\mathbf{y} = \mathbf{x}(i : j)$  means that  $\mathbf{y}$  consists of the  $i$ th through the  $j$ th element of  $\mathbf{x}$ , and  $\|\cdot\|$  denotes the  $l$ -2 norm of a vector.

## 2. Materials and Methods

In this section, we first formulate the 2D BOT problem, then we provide brief reviews of the PLKF and its two improved versions—the BC-PLKF and the IV-PLKF. After these, we propose the UB-PLKF to cope with the biased estimate problem in PLKF. Then,

we incorporate the velocity constraint information into the UB-PLKF to construct the VC-PLKF.

### 2.1. Problem Formulation

We focus on the BOT problem in 2D-plane with the bearing measurements received by a moving observer, which is shown in Figure 1. We denote by  $\mathbf{p}_k = [p_{x,k}, p_{y,k}]^T \in \mathbb{R}^{2 \times 1}$  and  $\mathbf{v}_k = [v_{x,k}, v_{y,k}]^T \in \mathbb{R}^{2 \times 1}$  the position and velocity of the target at the discrete time index  $k = 1, \dots, N$ , respectively. We then have target state vector  $\mathbf{x}_k = [\mathbf{p}_k^T, \mathbf{v}_k^T]^T \in \mathbb{R}^{4 \times 1}$ . The observer position is represented by the vector  $\mathbf{s}_k = [s_{x,k}, s_{y,k}]^T \in \mathbb{R}^{2 \times 1}$ , which is assumed to be known.

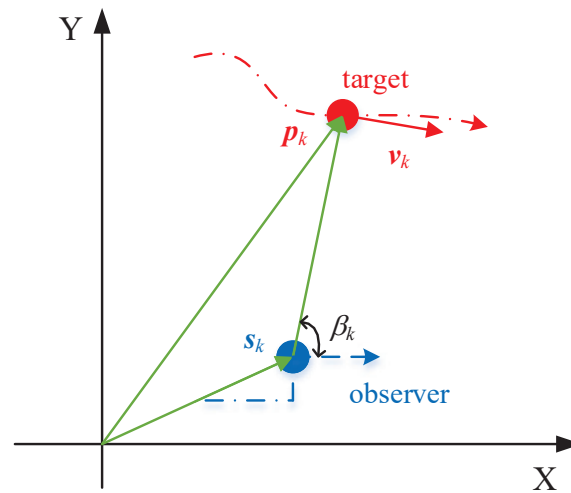


Figure 1. Geometry for 2-D BOT problem.

In the following, we assume that the target is moving under the constant velocity model as discussed in [32], then the target dynamic model is:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}, \quad (1)$$

where  $\mathbf{F} \in \mathbb{R}^{4 \times 4}$  is the transition matrix, and  $\mathbf{w}_{k-1} \in \mathbb{R}^{4 \times 1}$  is the process noise, which is assumed to be a Gaussian random vector with mean  $\mathbf{0}_{4 \times 1}$  and covariance matrix  $\mathbf{Q} \in \mathbb{R}^{4 \times 4}$ . The matrices  $\mathbf{F}$  and  $\mathbf{Q}$  are given by [32]

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2)$$

$$\mathbf{Q} = \begin{bmatrix} q_x \frac{T^3}{3} & 0 & q_x \frac{T^2}{2} & 0 \\ 0 & q_y \frac{T^3}{3} & 0 & q_y \frac{T^2}{2} \\ q_x \frac{T^2}{2} & 0 & q_x T & 0 \\ 0 & q_y \frac{T^2}{2} & 0 & q_y T \end{bmatrix} \quad (3)$$

where  $T$  is the sample interval, which is assumed to be equal to 1 s throughout the paper.  $q_x$  and  $q_y$  are respectively the power spectral densities of  $\mathbf{w}_k$  in two coordinates.

We adopt  $\tilde{\beta}_k$  to denote the bearing collected by the observer at  $k$ . It can be written as

$$\tilde{\beta}_k = \beta_k + n_k, \quad (4)$$

$$\beta_k = \tan^{-1} \frac{\Delta y_k}{\Delta x_k}, \quad (5)$$

where  $\Delta x_k = p_{x,k} - s_{x,k}$ ,  $\Delta y_k = p_{y,k} - s_{y,k}$ ,  $\tan^{-1}$  is the four-quadrant inverse tangent function, and  $n_k$  is the bearing noise, which is assumed to be Gaussian with zero mean and variance  $\sigma_k^2$ .  $\sigma_k$  is assumed to be known, and  $n_k$  is independent of the process noise  $w_k$  throughout the paper.

## 2.2. Overview of the PLKF, BC-PLKF, and IV-PLKF

This subsection first discusses the PLKF algorithm and then analyses its bias. Afterwards, the improved PLKF algorithms—BC-PLKF and IV-PLKF in [25] are introduced.

### 2.2.1. PLKF

Obviously, we have the following equation:

$$\begin{aligned}\sin n_k &= \sin(\tilde{\beta}_k - \beta_k) \\ &= \sin \tilde{\beta}_k \cos \beta_k - \cos \tilde{\beta}_k \sin \beta_k,\end{aligned}\quad (6)$$

multiplying  $d_k = \sqrt{\Delta x_k^2 + \Delta y_k^2}$  to both sides of (6), and notice that  $d_k \times \cos \beta_k = \Delta x_k$ ,  $d_k \times \sin \beta_k = \Delta y_k$ , we have

$$\begin{aligned}d_k \sin n_k &= \sin \tilde{\beta}_k \Delta x_k - \cos \tilde{\beta}_k \Delta y_k \\ &= \sin \tilde{\beta}_k (p_{x,k} - s_{x,k}) - \cos \tilde{\beta}_k (p_{y,k} - s_{y,k}) \\ &= [\sin \tilde{\beta}_k, -\cos \tilde{\beta}_k] \mathbf{p}_k - [\sin \tilde{\beta}_k, -\cos \tilde{\beta}_k] \mathbf{s}_k \\ &= [\sin \tilde{\beta}_k, -\cos \tilde{\beta}_k, 0, 0] \mathbf{x}_k - [\sin \tilde{\beta}_k, -\cos \tilde{\beta}_k] \mathbf{s}_k.\end{aligned}\quad (7)$$

Then we have the following pseudo-linear measurement equation:

$$z_k = \mathbf{H}_k \mathbf{x}_k + \eta_k, \quad (8)$$

where  $z_k = [\sin \tilde{\beta}_k, -\cos \tilde{\beta}_k] \mathbf{s}_k$ ,  $\mathbf{H}_k = [\sin \tilde{\beta}_k, -\cos \tilde{\beta}_k, 0, 0]$ , and  $\eta_k = -d_k \sin n_k$  is the pseudo-linear noise. The variance of  $\eta_k$  is denoted by  $R_k = \mathbb{E}\{\eta_k^2\} = d_k^2 \cdot \mathbb{E}\{\sin^2 n_k\} \approx d_k^2 \sigma_k^2$  for small bearing noise  $n_k$ .

Combining (1) and (8), the PLKF is given by

step 1 Predicting the state:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F} \hat{\mathbf{x}}_{k-1|k-1}; \quad (9)$$

step 2 Predicting the covariance matrix:

$$\mathbf{P}_{k|k-1} = \mathbf{F} \mathbf{P}_{k-1|k-1} \mathbf{F}^T + \mathbf{Q}; \quad (10)$$

step 3 Calculating the gain matrix:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{R}_k + \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T)^{-1}; \quad (11)$$

step 4 Updating the state:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (z_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}); \quad (12)$$

step 5 Updating the covariance matrix:

$$\mathbf{P}_{k|k} = (\mathbf{I}_{4 \times 4} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1}, \quad (13)$$

where  $\hat{\mathbf{x}}_{k|k-1} \in \mathbb{R}^{4 \times 1}$  and  $\mathbf{P}_{k|k-1} \in \mathbb{R}^{4 \times 4}$  denote, respectively, the one-step prediction and the corresponding prediction error covariance matrix, and  $\mathbf{K}_k \in \mathbb{R}^{4 \times 1}$  denotes the Kalman gain, and  $\hat{\mathbf{x}}_{k|k} \in \mathbb{R}^{4 \times 1}$  and  $\mathbf{P}_{k|k} \in \mathbb{R}^{4 \times 4}$  are, respectively, the posterior state estimate and corresponding estimation error covariance matrix. Since the true values of  $d_k$  and  $R_k$  are not available, the approximated values are used,

$$\hat{d}_k = \|\hat{\mathbf{x}}_{k|k-1}(1:2) - \mathbf{s}_k\|, \quad (14)$$

$$\hat{R}_k = \hat{d}_k^2 \sigma_k^2. \quad (15)$$

### 2.2.2. BC-PLKF

According to [25], the *posterior* bias  $\mathbf{e}_k = \hat{\mathbf{x}}_{k|k} - \mathbf{x}_k$  of the PLKF can be written as:

$$\mathbf{e}_k = \mathbf{A}_k + \mathbf{B}_k + \mathbf{C}_k \quad (16)$$

where

$$\mathbf{A}_k = \left( \mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_k^T \hat{\mathbf{R}}_k^{-1} \mathbf{H}_k \right)^{-1} \mathbf{P}_{k|k-1}^{-1} \mathbf{F} \left( \hat{\mathbf{x}}_{k-1|k-1} - \mathbf{x}_{k-1} \right), \quad (17)$$

$$\mathbf{B}_k = - \left( \mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_k^T \hat{\mathbf{R}}_k^{-1} \mathbf{H}_k \right)^{-1} \mathbf{P}_{k|k-1}^{-1} \mathbf{w}_{k-1}, \quad (18)$$

$$\mathbf{C}_k = \left( \mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_k^T \hat{\mathbf{R}}_k^{-1} \mathbf{H}_k \right)^{-1} \mathbf{P}_{k|k-1}^{-1} \mathbf{H}_k^T \hat{\mathbf{R}}_k^{-1} \eta_k. \quad (19)$$

The first term  $\mathbf{A}_k$  propagates the error from the last time  $k-1$ , and it is not the cause that leads to the biased estimate in PLKF. The second term  $\mathbf{B}_k$  propagates the bias from the correlation between  $\mathbf{H}_k$  and  $\mathbf{w}_{k-1}$ . If we assume the process noise  $\mathbf{w}_k$  is relatively small, then  $\mathbf{B}_k$  can be ignored. When it comes to the third term  $\mathbf{C}_k$ , it propagates the bias from the correlation between  $\mathbf{H}_k$  and  $\eta_k$ , which cannot be ignored since they all contain the bearing noise  $n_k$ . Thus, we can have the conclusion that it is the correlation between the  $\mathbf{H}_k$  and  $\eta_k$  that leads to the biased estimate, and if we can compensate the bias from  $\mathbf{C}_k$ , the bias in  $\hat{\mathbf{x}}_{k|k}$  can be reduced.

So the bias compensation can be achieved by

$$\hat{\mathbf{x}}_{k|k} - \left( \mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_k^T \hat{\mathbf{R}}_k^{-1} \mathbf{H}_k \right)^{-1} \mathbf{P}_{k|k-1}^{-1} \mathbf{H}_k^T \hat{\mathbf{R}}_k^{-1} \eta_k. \quad (20)$$

However, the true value of  $\mathbf{C}_k$  cannot be obtained; we are supposed to replace it with its conditional expectation based on  $\hat{\mathbf{x}}_{k|k}$ , which can be formulated as: [25]

$$\begin{aligned} \hat{\mathbf{C}}_k &= \left( \mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_k^T \hat{\mathbf{R}}_k^{-1} \mathbf{H}_k \right)^{-1} \mathbf{P}_{k|k-1}^{-1} \mathbb{E} \left\{ \mathbf{H}_k^T \hat{\mathbf{R}}_k^{-1} \eta_k | \hat{\mathbf{x}}_{k|k} \right\} \\ &= -\mathbf{P}_{k|k} \hat{\mathbf{R}}_k^{-1} \sigma_k^2 \mathbf{M}^T (\mathbf{M} \hat{\mathbf{x}}_{k|k} - \mathbf{s}_k), \end{aligned} \quad (21)$$

where  $\mathbf{M} = [\mathbf{I}_{2 \times 2}, \mathbf{0}_{2 \times 2}]$ , and we use the formula  $\mathbf{P}_{k|k} = \left( \mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_k^T \hat{\mathbf{R}}_k^{-1} \mathbf{H}_k \right)^{-1}$ .

Finally, the BC-PLKF is summarized below [25]:

step 1 Predicting the state:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F} \hat{\mathbf{x}}_{k-1|k-1}^{\text{BC}}; \quad (22)$$

step 2 Predicting the covariance matrix:

$$\mathbf{P}_{k|k-1} = \mathbf{F} \mathbf{P}_{k-1|k-1}^{\text{BC}} \mathbf{F}^T + \mathbf{Q}; \quad (23)$$

step 3 Calculating the gain matrix:

$$\mathbf{K}_k^{\text{BC}} = \mathbf{P}_{k|k-1} \mathbf{H}_k^T \left( \hat{\mathbf{R}}_k + \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T \right)^{-1}; \quad (24)$$

step 4 Updating the state:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k^{\text{BC}} (z_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}); \quad (25)$$

step 5 Updating the covariance matrix:

$$\mathbf{P}_{k|k}^{\text{BC}} = (\mathbf{I}_{4 \times 4} - \mathbf{K}_k^{\text{BC}} \mathbf{H}_k) \mathbf{P}_{k|k-1}; \quad (26)$$

step 6 Bias compensation:

$$\hat{\mathbf{x}}_{k|k}^{\text{BC}} = \hat{\mathbf{x}}_{k|k} + \mathbf{P}_{k|k}^{\text{BC}} \hat{\mathbf{R}}_k^{-1} \sigma_k^2 \mathbf{M}^T (\mathbf{M} \hat{\mathbf{x}}_{k|k} - \mathbf{s}_k). \quad (27)$$

### 2.2.3. IV-PLKF

Unlike the BC-PLKF, the IV-PLKF tries to construct a vector  $\mathbf{G}_k$ , which is chosen to be independent of  $\eta_k$ , and at the same time is strongly correlated with  $\mathbf{H}_k$ . The optimal form of  $\mathbf{G}_k$  is the noise-free version of  $\mathbf{H}_k$  [25]:

$$\mathbf{G}_k^{\text{opt}} = [\sin \beta_k, -\cos \beta_k] \mathbf{M}. \quad (28)$$

Since the true bearing measurement  $\beta_k$  is unavailable, a suboptimal vector based upon the estimate  $\hat{\beta}_k^{\text{BC}}$  from BC-PLKF is chosen:

$$\hat{\beta}_k^{\text{BC}} = \tan^{-1} \frac{\hat{\mathbf{x}}_{k|k}^{\text{BC}}(2) - s_{y,k}}{\hat{\mathbf{x}}_{k|k}^{\text{BC}}(1) - s_{x,k}}, \quad (29)$$

$$\mathbf{G}_k = [\sin \hat{\beta}_k^{\text{BC}}, -\cos \hat{\beta}_k^{\text{BC}}] \mathbf{M}. \quad (30)$$

The complete IV-PLKF algorithm is [25]:

step 1 Predicting the state:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F} \hat{\mathbf{x}}_{k-1|k-1}^{\text{IV}}; \quad (31)$$

step 2 Predicting the covariance matrix:

$$\mathbf{P}_{k|k-1} = \mathbf{F} \mathbf{P}_{k-1|k-1}^{\text{IV}} \mathbf{F}^T + \mathbf{Q}; \quad (32)$$

step 3 Calculating the gain matrix:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\hat{\mathbf{R}}_k + \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T)^{-1}; \quad (33)$$

step 4 Updating the state:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (z_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}); \quad (34)$$

step 5 Updating the covariance matrix:

$$\mathbf{P}_{k|k} = (\mathbf{I}_{4 \times 4} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1}; \quad (35)$$

step 6 Bias compensation:

$$\hat{\mathbf{x}}_{k|k}^{\text{BC}} = \hat{\mathbf{x}}_{k|k} + \mathbf{P}_{k|k} \hat{\mathbf{R}}_k^{-1} \sigma_k^2 \mathbf{M}^T (\mathbf{M} \hat{\mathbf{x}}_{k|k} - \mathbf{s}_k); \quad (36)$$

step 7 IV estimation:

$$\mathbf{G}_k = [\sin \hat{\beta}_k^{\text{BC}}, -\cos \hat{\beta}_k^{\text{BC}}] \mathbf{M} \quad (37)$$

$$\mathbf{K}_k^{\text{IV}} = \mathbf{P}_{k|k-1} \mathbf{G}_k^T (\hat{\mathbf{R}}_k + \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{G}_k^T)^{-1} \quad (38)$$

$$\hat{\mathbf{x}}_{k|k}^{\text{IV}} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k^{\text{IV}} (z_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}) \quad (39)$$

$$\mathbf{P}_{k|k}^{\text{IV}} = \mathbf{P}_{k|k-1} - \mathbf{K}_k^{\text{IV}} \mathbf{H}_k \mathbf{P}_{k|k-1}. \quad (40)$$

We shall notice that, although the asymptotic convergence of the IV based method is verified in [26], the performance of the IV–PLKF is still limited by the BC–PLKF due to the usage of  $\hat{x}_{k|k}^{BC}$  in calculating  $\hat{\beta}_k^{BC}$ . When the BC–PLKF performance deteriorates in the manoeuvring scenario, the IV–PLKF will also suffer.

### 2.3. The Proposed UB–PLKF and VC–PLKF

We first propose the UB–PLKF to cope with the biased estimate problem in PLKF. Then we incorporate the velocity constraint information into the UB–PLKF to construct the VC–PLKF.

#### 2.3.1. UB–PLKF

Revisiting the matrix  $H_k$  in (8), we can rewrite it as follows:

$$\begin{aligned} H_k &= [\sin \tilde{\beta}_k, -\cos \tilde{\beta}_k, 0, 0] \\ &= [\sin \beta_k \cos n_k, -\cos \beta_k \cos n_k, 0, 0] \\ &\quad + [\cos \beta_k \sin n_k, \sin \beta_k \sin n_k, 0, 0] \\ &\approx [\sin \beta_k, -\cos \beta_k, 0, 0] + n_k [\cos \beta_k, \sin \beta_k, 0, 0], \end{aligned} \quad (41)$$

where we assume  $\cos n_k \approx 1$  and  $\sin n_k \approx n_k$  for small bearing noise.

Substituting (41) into (8), we get:

$$z_k \approx [\sin \beta_k, -\cos \beta_k, 0, 0]x_k + n_k ([\cos \beta_k, \sin \beta_k, 0, 0]x_k - d_k); \quad (42)$$

here we also assume that  $\eta_k = -d_k \sin n_k \approx -d_k n_k$ . Let  $m_k = [\cos \beta_k, \sin \beta_k, 0, 0]x_k - d_k$ , and multiplying  $1/m_k$  to both sides of (42), we have:

$$\bar{z}_k = \bar{H}_k x_k + n_k, \quad (43)$$

where  $\bar{z}_k = z_k/m_k$ ,  $\bar{H}_k = [\sin \beta_k, -\cos \beta_k, 0, 0]/m_k$ . Notice that in (43), the measurement vector is independent of the noise, and the noise here is the bearing noise  $n_k$ , not the pseudo-linear noise  $\eta_k$ .

Based on (43), the expectation of  $C_k$  is

$$\mathbb{E}\{C_k\} = \mathbb{E}\left\{\left(P_{k|k-1}^{-1} + \bar{H}_k^T \sigma_k^{-2} \bar{H}_k\right)^{-1}\right\} \mathbb{E}\{\bar{H}_k^T \sigma_k^{-2} n_k\}. \quad (44)$$

It is clear that (44) equals 0 because  $\bar{H}_k$  is no longer correlated with  $n_k$ . Then, an unbiased PLKF algorithm is obtained as follows:

step 1 Predicting the state:

$$\hat{x}_{k|k-1} = F \hat{x}_{k-1|k-1}^{UB}; \quad (45)$$

step 2 Predicting the covariance matrix:

$$P_{k|k-1} = F P_{k-1|k-1}^{UB} F^T + Q; \quad (46)$$

step 3 Calculating the gain matrix:

$$K_k^{UB} = P_{k|k-1} \bar{H}_k^T \left( \sigma_k^2 + \bar{H}_k P_{k|k-1} \bar{H}_k^T \right)^{-1}; \quad (47)$$

step 4 Updating the state:

$$\hat{x}_{k|k}^{UB} = \hat{x}_{k|k-1} + K_k^{UB} (\bar{z}_k - \bar{H}_k \hat{x}_{k|k-1}); \quad (48)$$

step 5 Updating the covariance matrix:

$$\mathbf{P}_{k|k}^{\text{UB}} = (\mathbf{I}_{4 \times 4} - \mathbf{K}_k^{\text{UB}} \bar{\mathbf{H}}_k) \mathbf{P}_{k|k-1}. \quad (49)$$

Obviously, the true values of the bearing  $\beta_k$  and  $m_k$  in (43) cannot be obtained, we replace them with corresponding approximated versions  $\hat{\beta}_k^{\text{UB}}$  and  $\hat{m}_k$ , which are defined as follows:

$$\hat{\beta}_k^{\text{UB}} = \tan^{-1} \frac{\hat{\mathbf{x}}_{k|k-1}(2) - s_{y,k}}{\hat{\mathbf{x}}_{k|k-1}(1) - s_{x,k}}, \quad (50)$$

$$\hat{m}_k = [\cos \hat{\beta}_k^{\text{UB}}, \sin \hat{\beta}_k^{\text{UB}}, 0, 0] \hat{\mathbf{x}}_{k|k-1} - \hat{d}_k^{\text{UB}}, \quad (51)$$

$$\hat{d}_k^{\text{UB}} = \|\hat{\mathbf{x}}_{k|k-1}(1:2) - \mathbf{s}_k\|. \quad (52)$$

We emphasise here that the main idea behind the IV-PLKF and the UB-PLKF is similar. They both try to construct a noise-free measurement vector. However, the IV-PLKF performance is limited by the BC-PLKF, while the UB-PLKF does not suffer from this. The advantages of the UB-PLKF will be verified in the simulation part.

### 2.3.2. VC-PLKF

Based on the UB-PLKF, we develop its velocity constrained version in this part. First, we shall handle an optimisation problem with an equality constraint. Let  $\mathbf{v}_{k|k} = \hat{\mathbf{x}}_{k|k}^{\text{UB}}$  (3:4) and  $\mathbf{e}_{v,k} = \mathbf{v}_k - \mathbf{v}_{k|k}$ , we have:

$$\begin{aligned} \min_{\mathbf{K}_{v,k}} \quad & \mathbb{E}\{\mathbf{e}_{v,k}^T \mathbf{e}_{v,k}\} \\ \text{s.t.} \quad & \|\mathbf{v}_{k|k}\| = v, \end{aligned} \quad (53)$$

where  $\mathbf{K}_k^{\text{UB}} = [\mathbf{K}_{p,k}^T, \mathbf{K}_{v,k}^T]^T$ , and  $\mathbf{K}_{v,k} \in \mathbb{R}^{2 \times 1}$ .  $v$  is the velocity constraint, which is a positive constant. Note that  $\mathbb{E}\{\mathbf{e}_{v,k}^T \mathbf{e}_{v,k}\} = \text{tr}\{\mathbf{P}_{vv,k}^+\}$ , and  $\mathbf{P}_{k|k}^{\text{UB}} = \begin{bmatrix} \mathbf{P}_{pp,k}^+ & \mathbf{P}_{pv,k}^+ \\ \mathbf{P}_{vp,k}^+ & \mathbf{P}_{vv,k}^+ \end{bmatrix}$ . Then (53) can be rewritten as

$$\begin{aligned} \min_{\mathbf{K}_{v,k}} \quad & \text{tr}\{\mathbf{P}_{vv,k}^+\} \\ \text{s.t.} \quad & \|\mathbf{v}_{k|k}\| = v. \end{aligned} \quad (54)$$

Let  $\mathbf{v}_{k|k-1} = \hat{\mathbf{x}}_{k|k-1}$  (3:4) in (45), and according to the Kalman formula, we have:

$$\mathbf{v}_{k|k} = \mathbf{v}_{k|k-1} + \mathbf{K}_{v,k} \epsilon_k, \quad (55)$$

where  $\epsilon_k = \bar{\mathbf{z}}_k - \bar{\mathbf{H}}_k \hat{\mathbf{x}}_{k|k-1}^{\text{UB}}$ . Substituting (55) into the velocity constraint in (54), we have:

$$\epsilon_k^2 \mathbf{K}_{v,k}^T \mathbf{K}_{v,k} + 2 \mathbf{v}_{k|k-1}^T \mathbf{K}_{v,k} \epsilon_k + \mathbf{v}_{k|k-1}^T \mathbf{v}_{k|k-1} - v^2 = 0. \quad (56)$$

According to the Joseph formula [33], the update formula of the partitioned *posterior* covariance matrix can be written as:

$$\mathbf{P}_{vv,k}^+ = \mathbf{P}_{vv,k}^- - \mathbf{K}_{v,k} \bar{\mathbf{H}}_k \mathbf{P}_{2,k} - \mathbf{P}_{2,k}^T \bar{\mathbf{H}}_k^T \mathbf{K}_{v,k}^T + \mathbf{K}_{v,k} \mathbf{W}_k \mathbf{K}_{v,k}^T, \quad (57)$$



where

$$\begin{aligned} \mathbf{P}_{k|k-1} &= [\mathbf{P}_{1,k}, \mathbf{P}_{2,k}] \\ &= \begin{bmatrix} \mathbf{P}_{pp,k}^- & \mathbf{P}_{pv,k}^- \\ \mathbf{P}_{vp,k}^- & \mathbf{P}_{vv,k}^- \end{bmatrix} \end{aligned} \quad (58)$$

and

$$\mathbf{W}_k = \bar{\mathbf{H}}_k \mathbf{P}_{k|k-1} \bar{\mathbf{H}}_k^T + \sigma_k^2. \quad (59)$$

Utilising (56) and (57) and the *Lagrange* multiplier method, we can construct the performance index  $J_k$  as follows:

$$\begin{aligned} J_k = & \text{tr} \left\{ \mathbf{P}_{vv,k}^- - \mathbf{K}_{v,k} \bar{\mathbf{H}}_k \mathbf{P}_{2,k} - \mathbf{P}_{2,k}^T \bar{\mathbf{H}}_k^T \mathbf{K}_{v,k}^T + \mathbf{K}_{v,k} \mathbf{W}_k \mathbf{K}_{v,k}^T \right\} \\ & + \lambda_k \left[ \epsilon_k^2 \mathbf{K}_{v,k}^T \mathbf{K}_{v,k} + 2 \mathbf{v}_{k|k-1}^T \mathbf{K}_{v,k} \epsilon_k + \mathbf{v}_{k|k-1}^T \mathbf{v}_{k|k-1} - v^2 \right]. \end{aligned} \quad (60)$$

Taking the first derivative of  $J_k$  with respect to  $\mathbf{K}_{v,k}$  and setting it to zero yields:

$$- \mathbf{P}_{2,k}^T \bar{\mathbf{H}}_k^T + \mathbf{K}_{v,k} \mathbf{W}_k + \lambda_k (\mathbf{v}_{k|k-1} \epsilon_k + \mathbf{K}_{v,k} \epsilon_k^2) = 0. \quad (61)$$

Then we have the first-order condition:

$$\mathbf{K}_{v,k} = \frac{\mathbf{P}_{2,k}^T \bar{\mathbf{H}}_k^T - \lambda_k \mathbf{v}_{k|k-1} \epsilon_k}{\mathbf{W}_k + \lambda_k \epsilon_k^2}. \quad (62)$$

Substituting (62) into (56), and according to [27], the scalar equation with  $\lambda_k$  can be obtained as:

$$- \lambda_k^2 \tilde{\epsilon}_k^2 v^2 - 2 \lambda_k \tilde{\epsilon}_k v^2 + \left( \frac{\epsilon_k^2 \bar{\mathbf{H}}_k \mathbf{P}_{2,k} \mathbf{P}_{2,k}^T \bar{\mathbf{H}}_k^T}{\mathbf{W}_k^2} + \frac{2 \mathbf{v}_{k|k-1}^T \mathbf{P}_{2,k}^T \bar{\mathbf{H}}_k^T \epsilon_k}{\mathbf{W}_k} + \mathbf{v}_{k|k-1}^T \mathbf{v}_{k|k-1} - v^2 \right) = 0, \quad (63)$$

where  $\tilde{\epsilon}_k = \epsilon_k^2 / \mathbf{W}_k$ .

Then,  $\lambda_k$  can be written as

$$\lambda_k = \frac{-b - \sqrt{b^2 - 4ac}}{2a}, \quad (64)$$

where

$$a = -v^2 \tilde{\epsilon}_k^2, \quad (65)$$

$$b = -2 \tilde{\epsilon}_k v^2, \quad (66)$$

$$c = \epsilon_k^2 \bar{\mathbf{H}}_k \mathbf{P}_{2,k} \mathbf{P}_{2,k}^T \bar{\mathbf{H}}_k^T / \mathbf{W}_k^2 + 2 \mathbf{v}_{k|k-1}^T \mathbf{P}_{2,k}^T \bar{\mathbf{H}}_k^T \epsilon_k / \mathbf{W}_k + \mathbf{v}_{k|k-1}^T \mathbf{v}_{k|k-1} - v^2. \quad (67)$$

Notice that we use  $-b - \sqrt{b^2 - 4ac}$ , not  $-b \pm \sqrt{b^2 - 4ac}$ , because the minimum performance index  $J_k$  occurs when the minus sign is chosen according to [27].

Substituting (64)–(67) into (62) we get:

$$\mathbf{K}_{v,k}^* = \mathbf{K}_{v,k} + \left( \frac{v}{\|\mathbf{v}_{k|k}\|} - 1 \right) \frac{\mathbf{v}_{k|k}}{\epsilon_k}, \quad (68)$$

where the asterisk was added to distinguish from the unconstrained form  $\mathbf{K}_{v,k}$ . (see Appendix A).

Now we have the VC–PLKF with an equality constraint as follows:

step 1 Predicting the state:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1}^{\text{VC}}; \quad (69)$$

step 2 Predicting the covariance matrix:

$$\mathbf{P}_{k|k-1} = \mathbf{F}\mathbf{P}_{k-1|k-1}^{\text{VC}}\mathbf{F}^T + \mathbf{Q}; \quad (70)$$

step 3 Calculating the gain matrix for UB–PLKF:

$$\mathbf{K}_k^{\text{UB}} = \mathbf{P}_{k|k-1}\bar{\mathbf{H}}_k^T / (\sigma_k^2 + \bar{\mathbf{H}}_k\mathbf{P}_{k|k-1}\bar{\mathbf{H}}_k^T); \quad (71)$$

step 4 Calculating the innovation:

$$\epsilon_k = \bar{z}_k - \bar{\mathbf{H}}_k\hat{\mathbf{x}}_{k|k-1}; \quad (72)$$

step 5 Updating the state:

$$\hat{\mathbf{x}}_{k|k}^{\text{UB}} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k^{\text{UB}}\epsilon_k; \quad (73)$$

step 6 Updating the covariance for UB–PLKF:

$$\mathbf{P}_{k|k}^{\text{UB}} = (\mathbf{I}_{4 \times 4} - \mathbf{K}_k^{\text{UB}}\bar{\mathbf{H}}_k)\mathbf{P}_{k|k-1}; \quad (74)$$

step 7 Constructing the VC–PLKF:

$$\mathbf{v}_{k|k} = \hat{\mathbf{x}}_{k|k}^{\text{UB}}(3:4) \quad (75)$$

$$\mathbf{K}_{v,k}^* = \mathbf{K}_k^{\text{UB}}(3:4) + \left( \frac{v}{\|\mathbf{v}_{k|k}\|} - 1 \right) \frac{\mathbf{v}_{k|k}}{\epsilon_k} \quad (76)$$

$$\mathbf{K}_k^{\text{VC}} = [\mathbf{K}_k^{\text{UB}}(1:2); \mathbf{K}_{v,k}^*] \quad (77)$$

$$\hat{\mathbf{x}}_{k|k}^{\text{VC}} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k^{\text{VC}}\epsilon_k \quad (78)$$

$$\mathbf{P}_{k|k}^{\text{VC}} = (\mathbf{I}_{4 \times 4} - \mathbf{K}_k^{\text{VC}}\bar{\mathbf{H}}_k)\mathbf{P}_{k|k-1}. \quad (79)$$

In most cases, however, what we know is the target velocity range, and this means we should handle a problem with inequality constraints. To translate the inequality constraint into the equality constraint, we adopt the rule in [29], that is,

$$\hat{\mathbf{x}}_{k|k}^{\text{VC}} = \begin{cases} \hat{\mathbf{x}}_{k|k}^{\text{VC}}|_{v=v_{\max}} & \text{if } \|\mathbf{v}_{k|k}\| > v_{\max}, \\ \hat{\mathbf{x}}_{k|k}^{\text{VC}}|_{v=v_{\min}} & \text{if } \|\mathbf{v}_{k|k}\| < v_{\min}, \\ \hat{\mathbf{x}}_{k|k}^{\text{VC}} = \hat{\mathbf{x}}_{k|k}^{\text{UB}} & \text{if } v_{\min} \leq \|\mathbf{v}_{k|k}\| \leq v_{\max}. \end{cases} \quad (80)$$

Here,  $v_{\min}$  and  $v_{\max}$  denote the lower and upper bounds of the velocity, respectively. Equation (80) means that if  $\|\mathbf{v}_{k|k}\| \in [v_{\min}, v_{\max}]$ , the steps after (75) are avoided, and we just output  $\hat{\mathbf{x}}_{k|k}^{\text{VC}} = \hat{\mathbf{x}}_{k|k}^{\text{UB}}$  and  $\mathbf{P}_{k|k}^{\text{VC}} = \mathbf{P}_{k|k}^{\text{UB}}$ . Otherwise, if  $\|\mathbf{v}_{k|k}\|$  is less than  $v_{\min}$  or greater than  $v_{\max}$ , we do the VC–PLKF with the equality constraint with  $v = v_{\min}$  or  $v = v_{\max}$ , respectively.

We shall emphasise that, although the VC–PLKF and the algorithms in [29] all make use of the velocity constraints, the VC–PLKF constructs a norm-constrained Kalman filter, which is an iterative algorithm; however, algorithms in [29] are all based on the batch algorithms, which are computationally expensive and can hardly handle the manoeuvring target tracking problem.

### 3. Results

In this section, the proposed UB-PLKF and VC-PLKF are compared with the BC-PLKF and the IV-PLKF via Monte Carlo (MC) simulations. We adopt the root mean square error (RMSE) and the time-averaged RMSE as the performance metrics.

Assuming that  $\hat{\mathbf{x}}_{k|k}^i$  is the state vector that we estimate at  $k$  via  $i$ th MC simulation. We can define the RMSEs obtained from  $M = 300$  MC simulations as follows:

$$\text{RMSE}_k^{\text{pos}} = \left( \frac{1}{M} \sum_{i=1}^M \|\hat{\mathbf{p}}_{k|k}^i - \mathbf{p}_k\|^2 \right)^{1/2} \quad (81)$$

$$\text{RMSE}_k^{\text{vel}} = \left( \frac{1}{M} \sum_{i=1}^M \|\hat{\mathbf{v}}_{k|k}^i - \mathbf{v}_k\|^2 \right)^{1/2} \quad (82)$$

where  $\hat{\mathbf{p}}_{k|k}^i = \hat{\mathbf{x}}_{k|k}^i(1:2)$ ,  $\hat{\mathbf{v}}_{k|k}^i = \hat{\mathbf{x}}_{k|k}^i(3:4)$ .

The time-averaged RMSEs are also adopted and are computed as [25]:

$$\text{RMSE}_{\text{avg}}^{\text{pos}} = \left( \frac{1}{MU} \sum_{i=1}^M \sum_{k=P}^N \|\hat{\mathbf{p}}_{k|k}^i - \mathbf{p}_k\|^2 \right)^{1/2} \quad (83)$$

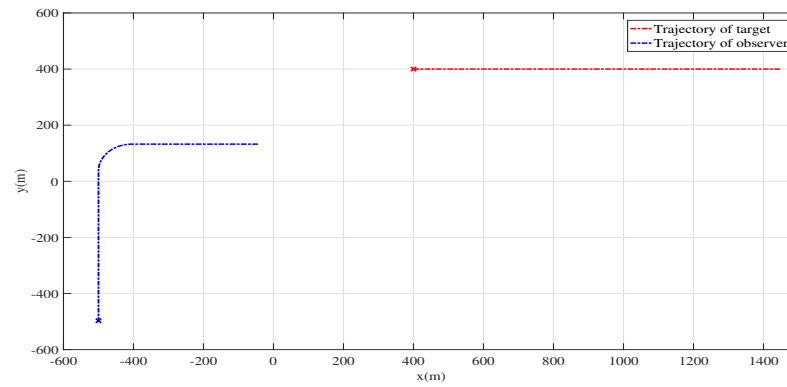
$$\text{RMSE}_{\text{avg}}^{\text{vel}} = \left( \frac{1}{MU} \sum_{i=1}^M \sum_{k=P}^N \|\hat{\mathbf{v}}_{k|k}^i - \mathbf{v}_k\|^2 \right)^{1/2}, \quad (84)$$

where  $U = N - P + 1$  is the effective time that takes into account the computation of time-averaged RMSEs,  $N = 350$  is the total number of time index  $k$ , and  $P = 60$  is chosen to eliminate the influence of the initial error according to [25].

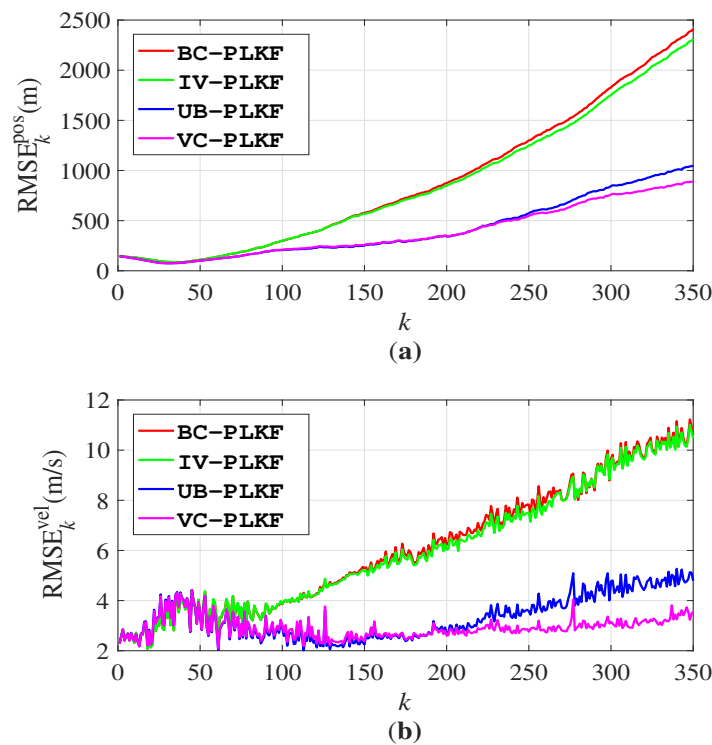
#### 3.1. Scenario 1: Non-Manoeuvring Target Tracking

We first test the filters' performance against poor initialisation and poor target sensor geometry with a non-maneuvring scenario, which is given in Figure 2. The initial state vector of the target is  $\mathbf{x}_0 = [400, 400, 3, 0]^T$ , and to test the performance against poor initialisation, we set the initial state of the estimate  $\hat{\mathbf{x}}_{0|-1}$  Gaussian, and its mean is  $[500, 300, 2, -1]^T$  and the covariance matrix is  $\mathbf{P}_{0|-1} = \text{diag}(625, 625, 4, 4)$ . The observer starts at  $[-500, -500]^T$  m and follows a linear path with a constant speed of  $[0, 3]^T$  m/s for the first 180 scans. Then, it executes a coordinated turn for 50 scans with a turn rate of  $1.8^\circ/\text{s}$  followed by 120 scans, where it moves in a straight line with velocity  $[3, 0]^T$  m/s. Notice that at the last 120 scans, the bearing rate between the target and the observer is 0, which constructs the case of poor target sensor geometry.  $q_x$  and  $q_y$  in (3) are  $q_x = q_y = 0.2 \text{ m}^2/\text{s}^3$ . The standard deviation  $\sigma_k = 5^\circ$  is assumed to be time invariant. The upper bound of the velocity is  $v_{\text{max}} = 6 \text{ m/s}$ , while the lower bound is  $v_{\text{min}} = 1 \text{ m/s}$ .

Figure 3 shows the RMSEs for different algorithms. The IV-PLKF and BC-PLKF show similar performances in RMSEs of both the position and velocity, and so do the UB-PLKF and VC-PLKF. However, the biases of the UB-PLKF and VC-PLKF are much smaller than the BC-PLKF and IV-PLKFs, and this suggests that splitting the noise  $n_k$  away from the  $\mathbf{H}_k$  is more effective than subtracting the bias from PLKF. Divergence problems have not been present in RMSEs of UB-PLKF and VC-PLKF, and this suggests that the proposed UB-PLKF and VC-PLKF can work well in conditions of poor initialisation and poor target sensor geometry.



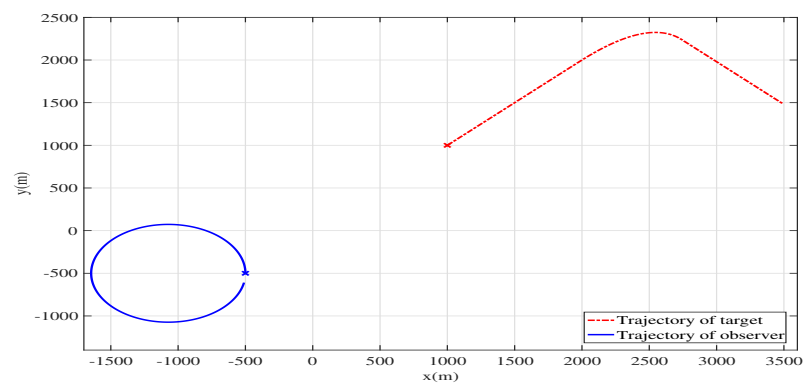
**Figure 2.** Geometry for scenario 1. The red curve represents the target trajectory while the observer trajectory is given by the blue curve.



**Figure 3.** RMSEs for different algorithms in non-maneuvring target tracking: (a) RMSEs for position; (b) RMSEs for velocity.

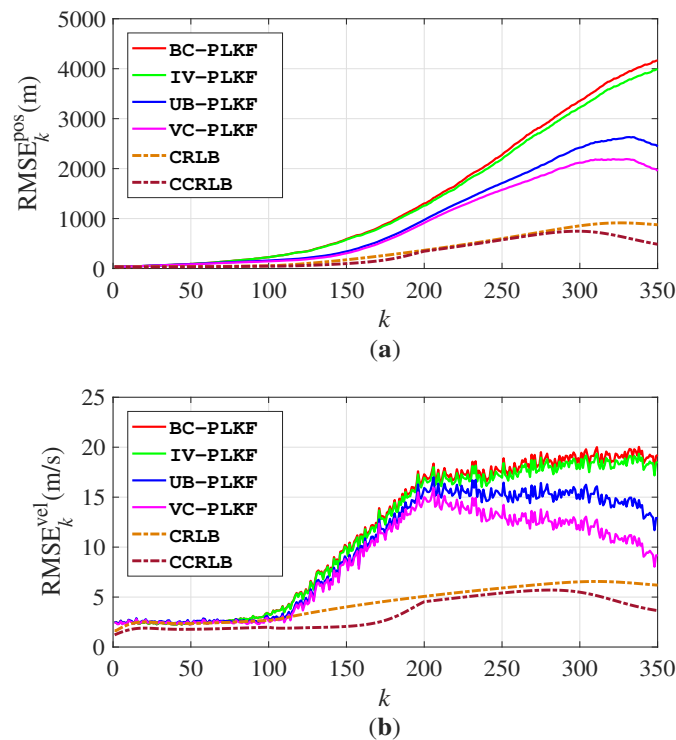
### 3.2. Scenario 2: Manoeuvring Target Tracking

Figure 4 represents the geometry of scenario 2. The initial state of the target is  $x_0 = [1000, 1000, 10, 10]^T$ , and it moves in a straight line for the first 100 scans. Then it performs a constant acceleration movement [32] with acceleration  $[-0.05, -0.15]^T \text{ m/s}^2$  in the following 100 scans. In the final 150 scans, it moves along a straight line at a constant speed  $[5, -5]^T \text{ m/s}$ . The initial state of the estimate  $\hat{x}_{0|-1}$  is a Gaussian random vector with mean  $[1000, 1000, 10, 10]^T$ , and the rest of the settings are the same as those of scenario 1, except that  $v_{\min} = 5 \text{ m/s}$  and  $v_{\max} = 16 \text{ m/s}$ . The observer begins at  $[-500, -500]^T \text{ m}$  with velocity  $[0, 9.45]^T \text{ m/s}$ , and executes a coordinated turn for the whole 350 scans with a turn rate of  $-1.8^\circ/\text{s}$ , where “ $-$ ” means the anticlockwise direction. We also adopt the Cramer–Rao lower bound (CRLB) [34] and the constrained Cramer–Rao lower bound (CCRLB) [35] in this scenario to evaluate the performance of the four algorithms (the derivation of CRLB and CCRLB can be found in Appendix B).

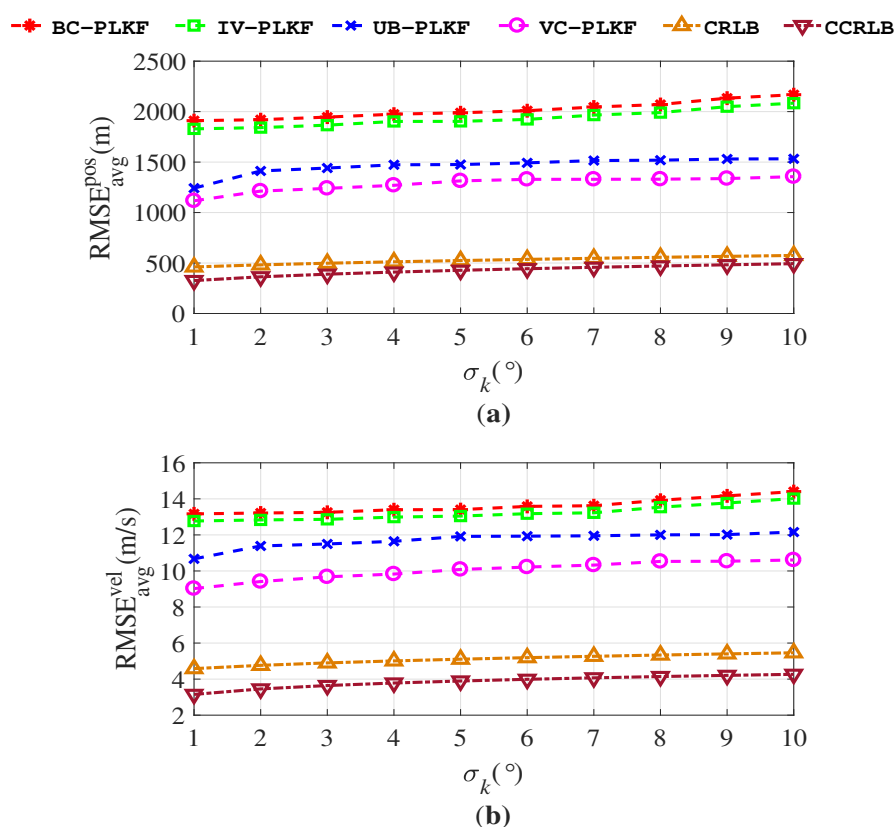


**Figure 4.** Geometry for scenario 2. The red curve represents the target trajectory while the observer trajectory is given by the blue curve.

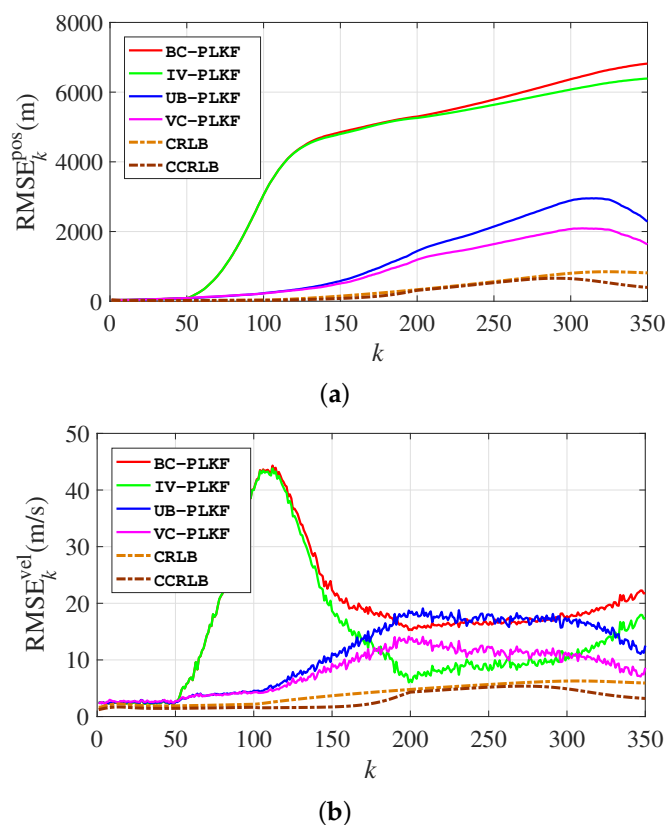
Figure 5 plots the RMSEs and CRLB for the four algorithms. The RMSEs of the VC-PLKF are closest to the CRLB and CCRLB, followed by UB-PLKF. We also find that the velocity RMSEs of UB-PLKF and VC-PLKF decrease after the manoeuvring, which means that the proposed algorithms are more adaptable to abrupt changes in the velocity. Figure 6 shows the time-averaged RMSEs for  $\sigma_k \in \{1^\circ, \dots, 10^\circ\}$ ; it also verifies that the VC-PLKF and UB-PLKF outperform the BC-PLKF and IV-PLKF. The time-averaged RMSEs of the four algorithms are also flat, which proves that the four algorithms all have good robustness to the different levels of the bearing noise.



**Figure 5.** RMSEs for different algorithms in manoeuvring target tracking: (a) RMSEs for position; (b) RMSEs for velocity.

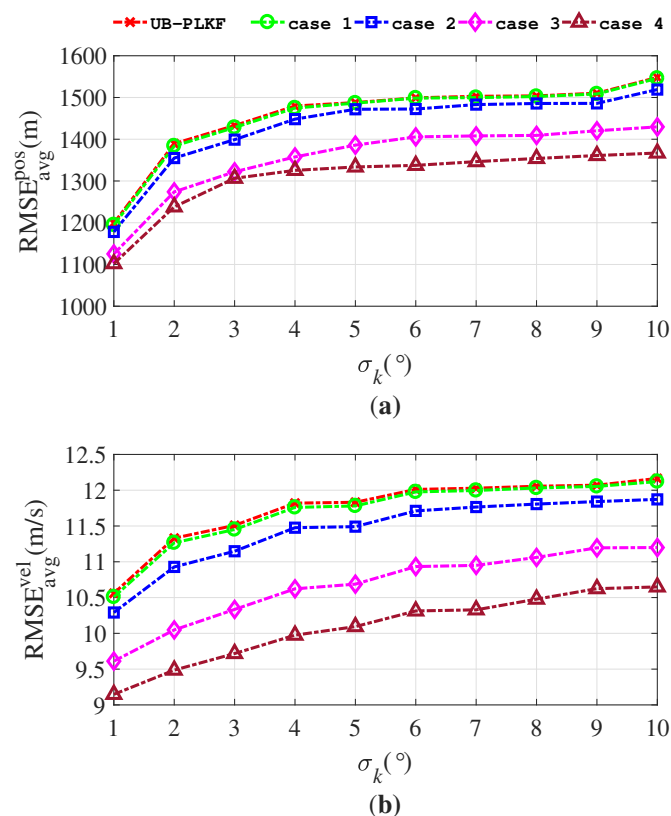


**Figure 6.** Time-averaged RMSEs for different algorithms in manoeuvring target tracking: (a) Time-averaged RMSEs for position; (b) Time-averaged RMSEs for velocity.



**Figure 7.** RMSEs for different algorithms in manoeuvring target tracking when there presents the noise statistics mismatch: (a) RMSEs for position; (b) RMSEs for velocity.

To test the performance of the proposed algorithms in the case in which there exists a noise statistics mismatch, we again assume that the deviation  $\sigma_k$  is time-varying. When  $k < 51$  or  $k > 201$ ,  $\sigma_k = 3^\circ$ , whereas  $\sigma_k = 7^\circ$  at the other times. The other settings remain unchanged. Figure 7 shows the RMSEs of different algorithms. It is clear that the position RMSEs of the BC-PLKF and IV-PLKF get worse when the mismatch begins, and retain high levels the rest of the time. However, the mismatch has little influence on the proposed UB-PLKF and VC-PLKF. The shapes of their positions and velocity RMSE curves are basically the same as those in Figure 5. This verifies that the UB-PLKF and VC-PLKF are more robust to the noise statistics mismatch. When it comes to the velocity RMSEs, some differences may occur. We can find that, although the velocity RMSEs of the IV-PLKF gets worse after the mismatch begins, it becomes smaller after the 100th scan. The possible reason is that at the 100th scan, the target starts to manoeuvre, and at this time, large measurement errors occur. So the large variance may more adaptable to this case. However, after the 201st scan, the velocity RMSEs of the BC-PLKF and IV-PLKF get worse again and finally exceed the RMSEs of UB-PLKF and VC-PLKF.



**Figure 8.** Time-averaged RMSEs for 4 cases of VC-PLKF in Table 1: (a) Time-averaged RMSEs for position; (b) Time-averaged RMSEs for velocity.

Table 1 lists four different cases of velocity constraints to compare the influence of different constraints in VC-PLKF. The time-averaged RMSEs are given by Figure 8. The closer the constraint is to the true speed, the smaller the RMSE of the VC-PLKF is. However, this improvement is at the cost of reducing the application scope of the algorithm. Thus, the choices of the upper and lower bounds depend on our prior knowledge of the target, otherwise it is suggested to choose a large range of constraints.

**Table 1.** Settings for 4 different cases in VC-PLKF.

	Case 1	Case 2	Case 3	Case 4
$v_{\max}(\text{m/s})$	21	19	17	16
$v_{\min}(\text{m/s})$	1	3	5	7

#### 4. Discussion

Section 3 compares the algorithms' performances in terms of RMSE and time-averaged RMSE. Figures 3 and 5 show that the proposed algorithms outperform the BC-PLKF and IV-PLKF in [25] both in non-maneuvring and manoeuvring scenarios. This is not surprising since the BC-PLKF just subtracts the approximated bias and does not split the noise away from the  $H_k$ . When the target manoeuvres or the target sensor geometry becomes poor, the approximated bias may not be accurate, which leads to huge biases in BC-PLKF. The IV-PLKF also suffers from this, since it utilizes the results of the BC-PLKF to calculate the approximated bearings. However, the proposed algorithms are unbiased and can be rapidly adapted to the changes in the scenarios.

Table 2 records the runtimes of the four algorithms, which are normalised by the UB-PLKF runtime. It shows that the UB-PLKF costs the least time among the four improved algorithms because it avoids the bias compensation step in BC-PLKF. The VC-PLKF and BC-PLKF have similar complexity, while the IV-PLKF takes the longest time to run. Considering the RMSEs, the time-averaged RMSEs and the runtimes, we can draw the conclusion that the proposed UB-PLKF and VC-PLKF outperform the BC-PLKF and IV-PLKF, both in terms of tracking accuracy and real-time application.

**Table 2.** Normalised runtimes for the 4 algorithms.

Algorithm	BC-PLKF	IV-PLKF	UB-PLKF	VC-PLKF
Runtime	1.35	2.12	1	1.21

In this paper, we assume that the statistical characteristics of noise are known a priori; however, this is not true in practical application. So, future research may contain the construction of the UB-PLKF in terms of the unknown statistical characteristics of noise by variational inference [36].

#### 5. Conclusions

To handle the bias problem in the traditional PLKF algorithm, we propose the UB-PLKF algorithm. Unlike the existing BC-PLKF and IV-PLKF, the UB-PLKF splits the noise away from the measurement vector directly, which can lead to an unbiased estimate. We then develop the VC-PLKF by solving a partial norm-constrained optimisation problem with an inequality relation to improve the performance of UB-PLKF. Simulation results show that the UB-PLKF and VC-PLKF outperform the BC-PLKF and IV-PLKF according to the RMSE, the time-averaged RMSE, and the computational complexity. The results also verify that the proposed algorithms can adapt to the abrupt changes of the target velocity, and thus are more suitable for the manoeuvring target tracking scenario.

**Author Contributions:** Conceptualization, Z.H.; methodology, Z.H.; software, Z.H. and S.C.; validation, C.H. and D.O.; formal analysis, Z.H.; investigation, Z.H. and S.C.; writing—original draft preparation, Z.H.; funding acquisition, C.H.; writing—review and editing, C.H. and D.O. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research is funded by the National Natural Science Foundation of China under Grant No. 61971412.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.



**Data Availability Statement:** Not applicable.

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

## Appendix A

Substituting (65)–(67) into (64), the polynomial  $b^2 - 4ac$  can be written as

$$\begin{aligned} b^2 - 4ac &= 4\tilde{\epsilon}_k^2 v^4 + 4v^2 \tilde{\epsilon}_k^2 \epsilon_k^2 \mathbf{H}_k \mathbf{P}_{2,k} \mathbf{P}_{2,k}^T \mathbf{H}_k^T / W_k^2 - 4v^4 \tilde{\epsilon}_k^2 \\ &\quad + 8v^2 \tilde{\epsilon}_k^2 \mathbf{v}_{k|k-1}^T \mathbf{P}_{2,k}^T \mathbf{H}_k^T \epsilon_k / W_k + 4v^2 \tilde{\epsilon}_k^2 \mathbf{v}_{k|k-1}^T \mathbf{v}_{k|k-1} \\ &= 4v^2 \tilde{\epsilon}_k^2 \|\epsilon_k \mathbf{P}_{2,k}^T \mathbf{H}_k^T / W_k + \mathbf{v}_{k|k-1}\|^2 \\ &= 4v^2 \tilde{\epsilon}_k^2 \|\mathbf{v}_{k|k}\|^2. \end{aligned} \quad (\text{A1})$$

Thus  $\lambda_k$  in (64) can be written as

$$\lambda_k = \frac{-1}{\tilde{\epsilon}_k} + \frac{\|\mathbf{v}_{k|k}\|}{\tilde{\epsilon}_k v}. \quad (\text{A2})$$

Substituting (A2) into (62), we have

$$\begin{aligned} \mathbf{K}_{v,k}^* &= \left[ \mathbf{P}_{2,k}^T \mathbf{H}_k^T + \frac{v - \|\mathbf{v}_{k|k}\|}{v \tilde{\epsilon}_k} \mathbf{v}_{k|k-1} \epsilon_k \right] (W_k + \lambda_k \epsilon_k^2)^{-1} \\ &= \left[ \mathbf{P}_{2,k}^T \mathbf{H}_k^T + \frac{v - \|\mathbf{v}_{k|k}\|}{v \tilde{\epsilon}_k} \mathbf{v}_{k|k-1} \epsilon_k \right] \frac{v}{\|\mathbf{v}_{k|k}\| W_k} \\ &= \underbrace{\frac{v \mathbf{P}_{2,k}^T \mathbf{H}_k^T}{W_k \|\mathbf{v}_{k|k}\|}}_{\text{I}} + \underbrace{\frac{v \epsilon_k \mathbf{v}_{k|k-1}}{\tilde{\epsilon}_k \|\mathbf{v}_{k|k}\| W_k}}_{\text{II}} - \underbrace{\frac{\epsilon_k \mathbf{v}_{k|k-1}}{\tilde{\epsilon}_k W_k}}_{\text{III}}. \end{aligned} \quad (\text{A3})$$

Note that in part I:  $\mathbf{P}_{2,k}^T \mathbf{H}_k^T / W_k = \mathbf{K}_{v,k}$ , and in part II and III:  $\epsilon_k / (\tilde{\epsilon}_k W_k) = 1 / \epsilon_k$ , (A3) can be further written as

$$\begin{aligned} \mathbf{K}_{v,k}^* &= \frac{v}{\|\mathbf{v}_{k|k}\|} \mathbf{K}_{v,k} + \frac{\mathbf{v}_{k|k-1} v}{\epsilon_k \|\mathbf{v}_{k|k}\|} - \frac{\mathbf{v}_{k|k-1}}{\epsilon_k} \\ &= \frac{v}{\|\mathbf{v}_{k|k}\|} \mathbf{K}_{v,k} + \left( \frac{v}{\|\mathbf{v}_{k|k}\|} - 1 \right) \frac{\mathbf{v}_{k|k-1}}{\epsilon_k} \\ &= \frac{v}{\|\mathbf{v}_{k|k}\|} \mathbf{K}_{v,k} + \left( \frac{v}{\|\mathbf{v}_{k|k}\|} - 1 \right) \left( \frac{\mathbf{v}_{k|k} - \mathbf{K}_{v,k} \epsilon_k}{\epsilon_k} \right) \\ &= \mathbf{K}_{v,k} + \left( \frac{v}{\|\mathbf{v}_{k|k}\|} - 1 \right) \frac{\mathbf{v}_{k|k}}{\epsilon_k}. \end{aligned} \quad (\text{A4})$$

## Appendix B

According to [34], an elegant way to calculate the Fisher information matrix (FIM)  $J_k$  recursively can be written as

$$J_{k+1} = D_k^{22} - D_k^{21} \left( J_k + D_k^{11} \right)^{-1} D_k^{12} \quad (\text{A5})$$

where

$$D_k^{11} = \mathbb{E} \left\{ -\Delta_{\mathbf{x}_k}^{\mathbf{x}_k} \lg p(\mathbf{x}_{k+1} | \mathbf{x}_k) \right\} \quad (\text{A6})$$

$$D_k^{12} = \mathbb{E} \left\{ -\Delta_{\mathbf{x}_k}^{\mathbf{x}_{k+1}} \lg p(\mathbf{x}_{k+1} | \mathbf{x}_k) \right\} \quad (\text{A7})$$

$$D_k^{21} = [D_k^{12}]^T, \quad (\text{A8})$$

and

$$D_k^{22} = \mathbb{E} \left\{ -\Delta_{x_{k+1}}^{x_{k+1}} \lg p(x_{k+1}|x_k) \right\} + \mathbb{E} \left\{ -\Delta_{x_{k+1}}^{x_{k+1}} \lg p(\bar{z}_{k+1}|x_{k+1}) \right\}. \quad (A9)$$

Here  $p(x_{k+1}|x_k)$  and  $p(\bar{z}_{k+1}|x_{k+1})$  are the transition and measurement probability density functions, and  $\Delta_{\Phi}^{\Psi}$  is the second-order partial derivative operators.

Based on (1) and (43), we have  $D_k^{11} = F^T Q^{-1} F$ ,  $D_k^{12} = -F^T Q^{-1}$ ,  $D_k^{21} = -Q^{-1} F$ , and  $D_k^{22} = Q^{-1} + \bar{H}_{k+1}^T \bar{H}_{k+1} / \sigma_{k+1}^2$ . The initial state of  $J_0$  is chosen to be  $P_{0|-1}$ .

When it comes to the constrained Cramer-Rao lower bound, its FIM  $J_{C,k}$  can be calculated as [35]

$$J_{C,k}^{-1} = J_k^{-1} - J_k^{-1} U_k^T (U_k J_k^{-1} U_k^T)^{-1} U_k J_k^{-1} \quad (A10)$$

where  $U_k$  is the gradient matrix which is defined by

$$U_k = \frac{\partial \|Ax_k\|^2}{\partial x_k^T} = 2x_k^T A^T A \quad (A11)$$

where  $A = [0_{2 \times 2}, I_{2 \times 2}]$ .

Finally, according to the relationship between the FIM and Cramer-Rao lower bound  $J_k^{-1} \triangleq \text{CRLB}$ , we can calculate the CRLB and CCRLB recursively.

## References

1. Lingren, A.G.; Gong, K.F. Position and Velocity Estimation Via Bearing Observations. *IEEE Trans. Aerosp. Electron. Syst.* **1978**, *AES-14*, 564–577. [\[CrossRef\]](#)
2. Nardone, S.; Lindgren, A.; Gong, K. Fundamental properties and performance of conventional bearings-only target motion analysis. *IEEE Trans. Autom. Control* **1984**, *29*, 775–787. [\[CrossRef\]](#)
3. Jagan, B.O.L.; Rao, S.K. Underwater surveillance in non-Gaussian noisy environment. *Meas. Control* **2020**, *53*, 250–261. [\[CrossRef\]](#)
4. Xu, S.; Doğançay, K.; Hmam, H. 3D AOA target tracking using distributed sensors with multi-hop information sharing. *Signal Process.* **2018**, *144*, 192–200. [\[CrossRef\]](#)
5. Su, J.; Li, Y.A.; Ali, W. Underwater angle-only tracking with propagation delay and time-offset between observers. *Signal Process.* **2020**, *176*, 107581. [\[CrossRef\]](#)
6. Modalavalasa, N.; Rao, G.S.B.; Prasad, K.S.; Ganesh, L.; Kumar, M.N.V.S.S. A new method of target tracking by EKF using bearing and elevation measurements for underwater environment. *Robot. Auton. Syst.* **2015**, *74*, 221–228. [\[CrossRef\]](#)
7. Liu, B.; Tang, X.; Tharmarasa, R.; Kirubarajan, T.; Jassemi, R.; Halle, S. Underwater Target Tracking in Uncertain Multipath Ocean Environments. *IEEE Trans. Aerosp. Electron. Syst.* **2020**, *56*, 4899–4915. [\[CrossRef\]](#)
8. Xu, S.; Doğançay, K.; Hmam, H. Distributed pseudolinear estimation and UAV path optimization for 3D AOA target tracking. *Signal Process.* **2017**, *133*, 64–78. [\[CrossRef\]](#)
9. He, L.; Gong, P.; Zhang, X.; Wang, Z. The Bearing-Only Target localization via the Single UAV: Asymptotically Unbiased Closed-Form Solution and Path Planning. *IEEE Access* **2019**, *7*, 153592–153604. [\[CrossRef\]](#)
10. Rutkowski, A.; Kawalec, A. Some of Problems of Direction Finding of Ground-Based Radars Using Monopulse Location System Installed on Unmanned Aerial Vehicle. *Sensors* **2018**, *20*, 5186. [\[CrossRef\]](#)
11. Liao, S.L.; Zhu, R.M.; Wu, N.Q.; Shaikh, T.A.; Sharaf, M.; Mostafa, A.M. Path planning for moving target tracking by fixed-wing UAV. *Def. Technol.* **2019**, *16*, 811–824. [\[CrossRef\]](#)
12. Pham, D.T. Some quick and efficient methods for bearing-only target motion analysis. *IEEE Trans. Signal Process.* **1993**, *41*, 2737–2751. [\[CrossRef\]](#)
13. Aidala, V.; Hammel, S. Utilization of modified polar coordinates for bearings-only tracking. *IEEE Trans. Autom. Control* **1983**, *28*, 283–294. [\[CrossRef\]](#)
14. Kim, J.; Suh, T.; Ryu, J. Bearings-only target motion analysis of a highly manoeuvring target. *IET Radar Sonar Navig.* **2017**, *11*, 1011–1019. [\[CrossRef\]](#)
15. 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. [\[CrossRef\]](#)
16. Arasaratnam, I.; Haykin, S. Cubature Kalman Filters. *IEEE Trans. Autom. Control* **2009**, *54*, 1254–1269. [\[CrossRef\]](#)
17. Toloei, A.; Niazi, S. State Estimation for Target Tracking Problems with Nonlinear Kalman Filter Algorithms. *Int. J. Comput. Appl.* **2014**, *98*, 30–36. [\[CrossRef\]](#)

18. Wu, H.; Chen, S.; Yang, B.; Chen, K. Robust Derivative-Free Cubature Kalman Filter for Bearings-Only Tracking. *J. Guid. Control Dyn.* **2016**, *39*, 1865–1870. [\[CrossRef\]](#)
19. Ristic, B.; Houssineau, J.; Arulampalam, S. Robust target motion analysis using the possibility particle filter. *IET Radar Sonar Navig.* **2018**, *13*, 18–22. [\[CrossRef\]](#)
20. Zhang, H.W.; Xie, W.X. Constrained auxiliary particle filtering for bearings-only maneuvering target tracking. *J. Syst. Eng. Electron.* **2019**, *30*, 684–695. [\[CrossRef\]](#)
21. Ayham, Z.; Thomas, S.; Shannon, D.A. Optimal Shadowing Filter for a Positioning and Tracking Methodology with Limited Information. *Sensors* **2019**, *19*, 931. [\[CrossRef\]](#)
22. Thomas, S.; Kevin, J. A guide to using shadowing filters for forecasting and state estimation. *Physica D* **2009**, *238*, 1260–1273. [\[CrossRef\]](#)
23. Aidala, V.J. Kalman Filter Behavior in Bearings-Only Tracking Applications. *IEEE Trans. Aerosp. Electron. Syst.* **1979**, *AES-15*, 29–39. [\[CrossRef\]](#)
24. Lin, X.; Kirubarajan, T.; Bar-Shalom, Y.; Maskell, S. Comparison of EKF, pseudomeasurement, and particle filters for a bearing-only target tracking problem. In Proceedings of the Signal and Data Processing of Small Targets 2002, Orlando, FL, USA, 2–4 April 2002; International Society for Optics and Photonics: Bellingham, WA, USA, 2002; Volume 4728, pp. 240–250. [\[CrossRef\]](#)
25. Nguyen, N.H.; Doğançay, K. Improved Pseudolinear Kalman Filter Algorithms for Bearings-Only Target Tracking. *IEEE Trans. Signal Process.* **2017**, *65*, 6119–6134. [\[CrossRef\]](#)
26. Zhang, Y.J.; Xu, G.Z. Bearings-Only Target Motion Analysis via Instrumental Variable Estimation. *IEEE Trans. Signal Process.* **2010**, *58*, 5523–5533. [\[CrossRef\]](#)
27. Zanetti, R.; Majji, M.; Bishop, R.H.; Mortari, D. Norm-Constrained Kalman Filtering. *J. Guid. Control Dyn.* **2009**, *32*, 1458–1465. [\[CrossRef\]](#)
28. Simon, D. Kalman filtering with state constraints: A survey of linear and nonlinear algorithms. *IET Control Theory Appl.* **2010**, *4*, 1303–1318. [\[CrossRef\]](#)
29. Badriasl, L.; Arulampalam, S.; van der Hoek, J.; Finn, A. Bayesian WIV Estimators for 3-D Bearings-Only TMA With Speed Constraints. *IEEE Trans. Signal Process.* **2019**, *67*, 3576–3591. [\[CrossRef\]](#)
30. He, S.M.; Shin, H.S.; Tsourdos, A. Trajectory Optimization for Target Localization With Bearing-Only Measurement. *IEEE Trans. Robot.* **2019**, *35*, 653–668. [\[CrossRef\]](#)
31. Kim, H.G.; Lee, J.Y.; Kim, H.J. Look Angle Constrained Impact Angle Control Guidance Law for Homing Missiles With Bearings-Only Measurements. *IEEE Trans. Aerosp. Electron. Syst.* **2018**, *54*, 3096–3107. [\[CrossRef\]](#)
32. Rong Li, X.; Jilkov, V.P. Survey of maneuvering target tracking. Part I. Dynamic models. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1333–1364. [\[CrossRef\]](#)
33. Zanetti, R.; DeMars, K.J. Joseph Formulation of Unscented and Quadrature Filters with Application to Consider States. *J. Guid. Control Dyn.* **2013**, *36*, 1860–1864. [\[CrossRef\]](#)
34. Schmitt, L.; Fichter, W. Globally Valid Posterior Cramer-Rao Bound for Three-Dimensional Bearings-Only Filtering. *IEEE Trans. Aerosp. Electron. Syst.* **2019**, *55*, 2036–2044. [\[CrossRef\]](#)
35. Nitzan, E.; Routtenberg, T.; Tabrikian, J. Cramer-Rao Bound for Constrained Parameter Estimation Using Lehmann-Unbiasedness. *IEEE Trans. Signal Process.* **2019**, *67*, 753–768. [\[CrossRef\]](#)
36. Sarkka, S.; Nummenmaa, A. Recursive Noise Adaptive Kalman Filtering by Variational Bayesian Approximations. *IEEE Trans. Autom. Control* **2009**, *3*, 596–600. [\[CrossRef\]](#)