Target Tracking Algorithm Using Two-Stage Cubature Kalman Filter

Lu Zhang[1]*, Ashish Bagwari[2], Gang Huang[1]

[1] College of Electrical and Information Engineering, Quzhou University, 324000 Quzhou, China
[2] Department of Electronics and Communication Engg., Women Institute of Technology (State Govt. Institution), Uttarakhand Technical University (State Govt. Technical University), 248007 Dehradun, India

* Correspondence: Lu Zhang (zhanglu@qzc.edu.cn)

Abstract: This study presents the two-stage cubature Kalman filter (TSCKF), which is a sophisticated technique designed to address the issue of variations in system models in real-life scenarios, and utilises nonlinear two-stage transformations to reorganise covariance matrices into a block-diagonal structure, effectively overcoming the limitations of conventional augmented methods. This technique effectively eliminates the need to calculate the cross-covariance between state variables and biases. This leads to a substantial reduction in computational load and facilitates seamless operation of the filter. The TSCKF design is underpinned by a robust theoretical framework, which ensures optimal computational efficiency while also ensuring precise estimations. This work demonstrates the mathematical equivalence between the TSCKF and the standard cubature Kalman filter (CKF) by utilising updated information equivalent transformations, and empirically verifies the equivalence through trajectory tracking experiments conducted on two-wheeled robotic systems subjected to random perturbations, thus affirming the greater accuracy and dependability of the TSCKF in tracking scenarios. Moreover, comparison evaluations offer further proof of the same performance between both methodologies. This study introduces a highly efficient approach in the domain of nonlinear systems and provides a dependable remedy for scenarios where traditional filtering procedures may be inadequate due to deficiencies in the system model.

Keywords: Nonlinear systems; Two-stage cubature Kalman filter algorithm; Nonlinear two-stage transformation; Matrix block diagonalization

1 Introduction

Introduced in 2009 by Arasaratnam and Haykin [1], the CKF is a nonlinear filtering methodology, and accurately estimates the posterior mean and covariance of nonlinear function transmission using the third-order spherical-radial cubature rule. Different from the unscented Kalman filter, this approach entails careful mathematical calculations to estimate the values of the five Gaussian integrals in nonlinear Gaussian filtering. Theoretical resilience is achieved by guaranteeing that the cubature points within the filter have positive weights, thereby successfully inhibiting the occurrence of covariance matrices that are not positive definite. The CKF, considered the most optimal approach for approximating Gaussian integrals [2, 3], quickly received extensive recognition in several disciplines due to its simplicity, superior estimate accuracy, and numerical stability in solving a variety of nonlinear estimating problems [4, 5].

Precise system models and accurate random information are required for the standard CKF. However, system models often contain unknown random biases in practical applications. A common solution for this is to treat the biases as part of the system state, augment the state, and then utilise Kalman filter while estimating both the system state and biases, which is known as the augmented state Kalman filter (ASKF) algorithm. However, it creates a substantial computational demand as system dimensions increase. Thus, this often leads to computational overflow and failures in digital implementations [6–9].

To address the aforementioned challenges, researchers worldwide have made various improvements. The square root CKF algorithm was updated by Zhong [10] using the Huber method for nonlinear statistical regression. In the presence of anomalous noise interference in the observed system values, Huber's robust filtering method was proposed by intersecting the observation values and modifying the affected data. Zhao et al. [11] developed

an integrated algorithm combining factor graphs and CKF, enabling the effective use of historical information for measurement updates to overcome uncertainties in the observation environments. Zhang et al. [12] proposed an adaptive H∞ CKF algorithm, based on a fading factor. This algorithm prevents a reduction in estimation accuracy in the face of uncertainty in system models. Huang et al. [13] introduced a filter algorithm based on variational Bayesian methods to determine state information in complex maritime conditions with erroneous inputs. Zhang et al. [14] proposed an adaptive Kalman filter algorithm based on variational Bayesian methods. This approach is specifically designed to simultaneously estimate the variables that represent the current state of the system, the matrices that describe the uncertainty in future predictions, and the matrices that represent the uncertainty in the measurements, thereby effectively dealing with the complex relationships that naturally exist in state estimation and predictive covariance matrices.

The TSCKF algorithm is introduced in this study. Utilizing nonlinear T-transform, the TSCKF algorithm applies matrix diagonalization to effectively diagonalize the state error covariance matrix. This methodological advancement circumvents the need for additional computation of cross-covariance between the state and biases, significantly reducing the computational burden. Such an approach not only streamlines the filtering process but also assures the uninterrupted and efficient operation of the system.

2 System Model

The primary emphasis is placed on a model of a discrete stochastic system that is nonlinear and includes random biases of unknown kind. The model is described in the following manner:

$$
\begin{aligned}
x_{k+1} &= f\left(x_k\right) + B_k b_k + \omega_k^x \\
b_{k+1} &= b_k + \omega_k^b \\
z_k &= h\left(x_k\right) + F_k b_k + v_k
\end{aligned}
\tag{1}
$$

Let $x_k \in R^n$ represent the state vector, $z_k \in R^m$ represent the observation vector, and $b_k \in R^p$ represent the system bias vector. The nonlinear functions $f(\cdot)$ and $h(\cdot)$, which represent the state transition and observation functions respectively, are continuously differentiable at point x_k. The matrices D_k and F_k are the coefficient matrices for the bias state equation and observation equation, respectively, with the correct dimensions. The noise sequences ω_k^x, ω_k^b and v_k are independent zero-mean Gaussian white noise sequences with variances as indicated:

$$
E\left[\left[\begin{array}{c} \omega_k^x \\ \omega_k^b \\ v_k \end{array}\right]\left[\begin{array}{c} \omega_j^x \\ \omega_j^b \\ v_j \end{array}\right]^T\right] = \left[\begin{array}{ccc} Q_k^x & 0 & 0 \\ 0 & Q_k^b & 0 \\ 0 & 0 & R_k \end{array}\right]\delta_{kj}
\tag{2}
$$

Given that Q_k^x, Q_k^b and R_k are positive definite symmetric matrices, and δ_{kj} represents the Kronecker function.

The starting state values x_0 and b_0 are supposed to be uncorrelated white noise processes, with each being a Gaussian random variable possessing defined features.

$$
E\left[x_0\right] = \hat{x}_0, E\left[\left(x_0 - \hat{x}_0\right)\left(x_0 - \hat{x}_0\right)^T\right] = P_0^x > 0
$$

$$
E\left[b_0\right] = \hat{b}_0, E\left[\left(b_0 - \hat{b}_0\right)\left(b_0 - \hat{b}_0\right)^T\right] = P_0^b > 0
$$

The subsequent parts utilise the augmented state cubature Kalman filter (ASCKF) algorithm to initially estimate the system states and biases. An analysis is conducted on the limits of this augmented technique, which is then followed by the introduction of the TSCKF algorithm and its subsequent algorithmic analysis. Ultimately, experimental validation is performed.

3 ASCKF Algorithm

Let

$$
X_{k+1} = \left[\begin{array}{c} x_{k+1} \\ b_{k+1} \end{array}\right], f\left(X_k\right) = \left[\begin{array}{c} f\left(x_k\right) + B_k b_k \\ b_k \end{array}\right], \omega_k = \left[\begin{array}{c} \omega_k^x \\ \omega_k^b \end{array}\right], h\left(X_k\right) = h\left(x_k\right) + F_k b_k
$$

The Eqs. (2)-(36) can be reformulated as:

$$
\begin{aligned}
X_{k+1} &= f\left(X_k\right) + \omega_k \\
Z_k &= h\left(X_k\right) + v_k
\end{aligned}
\tag{3}
$$

where, $E\left(\omega_k, \omega_j^T\right) = \begin{bmatrix} Q_k^x & 0 \\ 0 & Q_k^b \end{bmatrix} \delta_{kj}$

b_k is incorporated into the current state of the system, and both x_k and b_k are considered as additional dimensions of the expanded state. The ASCKF algorithm operates in the following manner:

Step 1: Time update

(a) The Cholesky decomposition for $P_{k-1|k-1}$ is applied to the posterior density function $p\left(x_{k-1} \mid D_{k-1}\right) = N\left(x_{k-1}; \hat{x}_{k-1}, P_{k-1}\right)$ at time $k-1$ to obtain:

$$P_{k-1|k-1} = S_{k-1|k-1}S_{k-1|k-1}^T \tag{4}$$

(b) Determine the cubature points $(i = 1, 2, \ldots, m = 2n_x)$:

$$X_{i,k-1|k-1} = S_{k-1|k-1}\xi_i + \hat{X}_{k-1|k-1} \tag{5}$$

(c) Calculate the propagated cubature points $(i = 1, 2, \ldots, m)$:

$$X_{i,k|k-1}^* = f\left(X_{i,k-1|k-1}, u_{k-1}\right) \tag{6}$$

(d) Calculate the anticipated state at time k:

$$\hat{x}_{k|k-1} = \frac{1}{m}\sum_{i=1}^m X_{i,k|k-1}^* \tag{7}$$

(e) Calculate the estimated covariance of the state error at time k:

$$P_{k|k-1} = \frac{1}{m}\sum_{i=1}^m X_{i,k|k-1}^* X_{i,k|k-1}^{*T} - \hat{x}_{k|k-1}\hat{x}_{k|k-1}^T + Q_{k-1} \tag{8}$$

Step 2: Measurement update

(a) Compute the Cholesky decomposition:

$$P_{k|k-1} = S_{k|k-1}S_{k|k-1}^T \tag{9}$$

(b) Determine the cubature points $(i = 1, 2, \ldots, m)$:

$$X_{i,k|k-1} = S_{k|k-1}\xi_i + \hat{X}_{k|k-1} \tag{10}$$

(c) Calculate the propagated cubature points $(i = 1, 2, \ldots, m)$:

$$Z_{i,k|k-1} = h\left(X_{i,k|k-1}, u_k\right) \tag{11}$$

(d) Calculate the anticipated value of the measurement at time k:

$$\hat{z}_{k|k-1} = \frac{1}{m}\sum_{i=1}^m Z_{i,k|k-1} \tag{12}$$

(e) Calculate the covariance of the measurement error at time k:

$$P_{zz,k|k-1} = \frac{1}{m}\sum_{i=1}^m Z_{i,k|k-1}Z_{i,k|k-1}^T - \hat{z}_{k|k-1}\hat{z}_{k|k-1}^T + R_k \tag{13}$$

(f) Calculate the cross-covariance at time k:

$$P_{xz,k|k-1} = \frac{1}{m} \sum_{i=1}^{m} X_{i,k|k-1} Z_{i,k|k-1}^T - \hat{x}_{k|k-1} \hat{z}_{k|k-1}^T \tag{14}$$

(g) Determine the gain at time k:

$$K_k = P_{xz,k|k-1} P_{zz,k|k-1}^{-1} \tag{15}$$

(h) Calculate the state estimation at time k:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \left(z_k - \hat{z}_{k|k-1} \right) \tag{16}$$

(i) Calculate the estimation of the state error covariance at time k:

$$P_{k|k} = P_{k|k-1} - K_k P_{zz,k|k-1} K_k^T \tag{17}$$

In the algorithm, $\hat{X}(\cdot) = \begin{bmatrix} \hat{x}(\cdot) \\ \hat{b}(\cdot) \end{bmatrix}$, $K_k = \begin{bmatrix} K_k^x \\ K_k^b \end{bmatrix}$, $P(\cdot) = Cov\{\hat{X}(\cdot)\} = \begin{bmatrix} P_{(\cdot)}^x & P_{(\cdot)}^{xb} \\ \left(P_{(\cdot)}^{xb} \right)^T & P_{(\cdot)}^b \end{bmatrix}$.

The dimensions of the augmented matrix are given by the sum of n and p. When the values of p and n are similar, the dimension of the new state vector X_k grows greatly in comparison to the dimension of the original state. The increase in computational burden for the ASCKF filter results in a significant escalation, which might potentially lead to overflow and system crashes during digital calculation. The main factor contributing to this problem is the supplementary calculation of $P_{(\cdot)}^{xb}$. In response to this, the notion of a two-stage filter was established. Based on the literature, there are two approaches to execute a two-stage filter. Friedland proposed an approach that suggests separating the augmented filter into two parallel reduced-order filters: a bias-free filter and a bias filter [15]. The bias-free filter operates under the assumption that there is no inherent bias in the system. On the other hand, the bias filter generates an estimation of the bias vector, which is subsequently employed to rectify the output of the bias-free filter. This process allows for the estimation of both the state and bias vectors. Friedland's two-stage method is considered best within specific algebraic constraints. However, these requirements are seldom attainable in practical scenarios, rendering most conventional two-stage methods less than optimal. Another approach utilises a matrix transformation technique referred to as the two-stage transformation. This transformation, devoid of algebraic limitations, can attain optimal performance [16, 17]. In linear systems, the two-stage transformation is observed as matrix block diagonalization. In nonlinear systems, it is observed as matrix block diagonalization achieved by deriving the covariance matrix.

4 TSCKF Algorithm

The enhanced Kalman filter employs the Jacobian matrix to calculate the error covariance matrix and the gain matrix. However, the Jacobian matrix is unnecessary in the CKF calculation. Although the two-stage augmented Kalman filter methods have been suggested by researchers [18–20], it is not possible for the TSCKF to directly utilise the framework of the augmented Kalman filter. To produce the effects of matrix block diagonalization observed in the augmented Kalman filter, it is essential to determine the relationship of matrix block diagonalization inside the CKF framework. This section presents the matrix block decoupling approach as a means to develop the TSCKF algorithm, building upon the aforementioned concept.

4.1 TSCKF

The T-transform in linear systems is defined as:

$$T(G) = \begin{pmatrix} I_{n-p} & G \\ 0 & I_P \end{pmatrix} \tag{18}$$

Therefore, the linear two-stage transformation can be represented by the attributes of the T-transform as follows:

$$\hat{X}_{k|k-1} = T(U_k) \bar{X}_{k|k-1}$$
$$\hat{X}_{k|k} = T(V_k) \bar{X}_{k|k}$$
$$P_{k|k-1} = T(U_k) \bar{P}_{k|k-1} T^T(U_k) \tag{19}$$
$$P_{k|k} = T(V_k) \bar{P}_{k|k} T^T(V_k)$$
$$K_k = T(V_k) \bar{K}_k$$

where, $\bar{P} = diag\left\{\bar{P}^1, \bar{P}^2\right\}$.

The nonlinear T-transform and two-stage transformation are obtained for the given Eq. (1). This represents a nonlinear discrete system with unknown biases. This derivation is based on the linear T-transform and two-stage transformation from Eq. (18) and Eq. (19).

Lemma 1: By applying techniques from studies [8, 9], the linear system T-transform may be extended to nonlinear systems, resulting in the following formula for the nonlinear T-transform:

$$T(F, X) = \begin{bmatrix} X_1 + F(X_2) \\ X_2 \end{bmatrix} \tag{20}$$

where, $X = \left\{(X^1)^T, (X^2)^T\right\}^T, X^1 \in R^{n-p}, X^2 \in R^p, F(X^2)$ are non-linear functions of the sub-state X^2. The nonlinear T-transform properties are used to obtain formulas for a two-stage transformation:

$$\hat{X}_{k|k-1} = T\left(\Phi, \bar{X}_{k|k-1}\right) \tag{21}$$

$$\hat{X}_{k|k} = T\left(\Psi, \bar{X}_{k|k}\right) \tag{22}$$

$$P_{k|k-1} = \frac{\partial T\left(\Phi, \bar{X}_{k|k-1}\right)}{\partial \bar{X}_{k|k-1}} \bar{P}_{k|k-1} \left(\frac{\partial T\left(\Phi, \bar{X}_{k|k-1}\right)}{\partial \bar{X}_{k|k-1}}\right)^T \tag{23}$$

$$P_{k|k} = \frac{\partial T\left(\Psi, \bar{X}_{k|k}\right)}{\partial \bar{X}_{k|k}} \bar{P}_{k|k} \left(\frac{\partial T\left(\Psi, \bar{X}_{k|k}\right)}{\partial \bar{X}_{k|k}}\right)^T \tag{24}$$

$$K_k = \frac{\partial T\left(\Psi, \bar{X}_{k|k}\right)}{\partial \bar{X}_{k|k}} \bar{K}_k \tag{25}$$

where, Φ and Ψ represent two specific nonlinear functions.

The lemma concludes.

The lemma has the subsequent characteristics:

$$\frac{\partial T\left(\Phi, \bar{X}_{k|k-1}\right)}{\partial \bar{X}_{k|k-1}} = \begin{bmatrix} I_{n-p} & U_k \\ 0 & I_p \end{bmatrix} \equiv T(U_k) \tag{26}$$

$$\frac{\partial T\left(\Psi, \bar{X}_{k|k}\right)}{\partial \bar{X}_{k|k}} = \begin{bmatrix} I_{n-p} & V_k \\ 0 & I_p \end{bmatrix} \equiv T(V_k) \tag{27}$$

Therefore, it can be inferred that:

$$U_k = \frac{\partial \Phi\left(\bar{X}_{k|k-1}^2\right)}{\partial \bar{X}_{k|k-1}^2}, V_k = \frac{\partial \Psi\left(\bar{X}_{k|k}^2\right)}{\partial \bar{X}_{k|k}^2} \tag{28}$$

The TSCKF operates as follows for the nonlinear discrete stochastic system with unknown random biases, as depicted in Eq. (1):

(a) Unbiased filter:

$$\bar{X}^1_{k|k-1} = \frac{1}{m}\sum_{i=1}^{m} f^1\left(S_{k-1|k-1}\xi_i + T\left(\Psi, \bar{X}_{k-1|k-1}\right), u_{k-1}\right) - \Phi\left(\bar{X}^2_{k|k-1}\right) \tag{29}$$

$$\begin{aligned} \bar{X}^1_{k|k} &= \bar{X}^1_{k|k-1} + \Phi\left(\bar{X}^2_{k|k-1}\right) + V_k\left(\bar{X}^2_{k|k} - \bar{X}^2_{k|k-1}\right) - \Psi\left(\bar{X}^2_{k|k}\right) \\ &\quad + \bar{K}^1_k\left(Z_k - \frac{1}{m}\sum_{i=1}^{m} h\left(S_{k|k-1}\xi_i + T\left(\Phi, \bar{X}_{k|k-1}\right), u_k\right)\right) \end{aligned} \tag{30}$$

$$\bar{P}^1_{k|k-1} = M^{11}_{k-1} + Q^{11}_{k-1} - U_k\left(M^{22}_{k-1} + Q^{22}_{k-1}\right)U^T_k \tag{31}$$

$$\begin{aligned} \bar{P}^1_{k|k} &= \bar{P}^1_{k|k-1} + U_k\bar{P}^2_{k|k-1}U^T_k - V_k\bar{P}^2_{k|k-1}V^T_k - \bar{K}^1_k P_{zz,k|k-1}\left(\bar{K}^1_k\right)^T \\ &\quad - \bar{K}^1_k P_{zz,k|k-1}\left(\bar{K}^2_k\right)^T V^T_k - \left(\bar{K}^1_k P_{zz,k|k-1}\left(\bar{K}^2_k\right)^T V^T_k\right)^T \end{aligned} \tag{32}$$

$$\bar{K}^1_k = N^1_k - V_k N^2_k \tag{33}$$

(b) Bias filter:

$$\bar{X}^2_{k|k-1} = \frac{1}{m}\sum_{i=1}^{m} f^2\left(S_{k-1|k-1}\xi_i + T\left(\Psi, \bar{X}_{k-1|k-1}\right), u_{k-1}\right) \tag{34}$$

$$\bar{X}^2_{k|k} = \bar{X}^2_{k|k-1} + \bar{K}^2_k\left(Z_k - \frac{1}{m}\sum_{i=1}^{m} h\left(S_{k|k-1}\xi_i + T\left(\Phi, \bar{X}_{k|k-1}\right), u_k\right)\right) \tag{35}$$

$$\bar{P}^2_{k|k-1} = M^{22}_{k-1} + Q^{22}_{k-1} \tag{36}$$

$$\bar{P}^2_{k|k} = \bar{P}^2_{k|k-1} - \bar{K}^2_k P_{zz,k|k-1}\left(\bar{K}^2_k\right)^T \tag{37}$$

$$\bar{K}^2_k = N^2_k \tag{38}$$

The interdependence between the two filters:

$$U_k = \left(M^{12}_{k-1} + Q^{12}_{k-1}\right)\left(M^{22}_{k-1} + Q^{22}_{k-1}\right)^{-1} \tag{39}$$

$$V_k = U_k - \bar{K}^1_k P_{zz,k|k-1}\left(\bar{K}^2_k\right)^T\left(\bar{P}^2_{k|k-1}\right)^{-1} \tag{40}$$

where, $P_{k|k-1} = M_{k-1} + Q_{k-1} = \begin{bmatrix} M^{11}_{k-1} + Q^{11}_{k-1} & M^{12}_{k-1} + Q^{12}_{k-1} \\ \left(M^{12}_{k-1} + Q^{12}_{k-1}\right)^T & M^{22}_{k-1} + Q^{22}_{k-1} \end{bmatrix}$, $K_k = \begin{bmatrix} K^1_k \\ K^2_k \end{bmatrix}$.

The nonlinear functions Φ and Ψ can be calculated by utilising Eq. (28) and the backward difference equation.

$$\Phi\left(\bar{X}^2_{k|k-1}\right) = \Phi\left(\bar{X}^2_{k-1|k-2}\right) + U_k\left(\bar{X}^2_{k|k-1} - \bar{X}^2_{k-1|k-2}\right)$$
$$\Psi\left(\bar{X}^2_{k|k}\right) = \Psi\left(\bar{X}^2_{k-1|k-1}\right) + V_k\left(\bar{X}^2_{k|k} - \bar{X}^2_{k-1|k-1}\right)$$

4.2 Demonstration of Equivalence for the TSCKF Algorithm

The TSCKF algorithm is obtained by applying a non-singular two-stage transformation to the CKF method, which establishes their mathematical equivalence.

Theorem 1: The TSCKF method is mathematically equivalent to the CKF algorithm, as proven in Eqs. (4)-(14).

Proof: Employing inductive reasoning, let us assume that at time k:

$$\hat{X}_{k|k} = X_{k|k} \tag{41}$$

By applying Eqs. (5), (6), (7), (20), (29), (34), and (41), it can be deduced that:

$$
T\left(\phi, \bar{X}_{k+1|k}\right) = \begin{pmatrix} \bar{X}^1_{k+1|k} + \phi\left(\bar{X}^2_{k+1|k}\right) \\ \bar{X}^2_{k+1|k} \end{pmatrix} = \frac{1}{m}\sum_{i=1}^{m} f\left(S_{k|k}\xi_i + T\left(\psi, \bar{X}_{k|k}\right), u_k\right)
$$

$$
= \frac{1}{m}\sum_{i=1}^{m} f\left(S_{k|k}\xi_i + \hat{X}_{k|k}, u_k\right) = X_{k+1|k} \tag{42}
$$

Subsequently, by utilising Eqs. (16), (22), (25), (30), (35), and (42), it can be deduced that:

$$
\hat{X}_{k+1|k+1} =
$$
$$
\begin{bmatrix} \bar{X}^1_{k+1|k} + \Phi\left(\bar{X}^2_{k+1|k}\right) + V_k\left(\bar{X}^2_{k+1|k+1} - \bar{X}^2_{k+1|k}\right) \\ +\bar{K}^1_{k+1}\left(Z_{k+1} - \frac{1}{m}\sum_{i=1}^{m} h\left(S_{k+1|k}\xi_i + T\left(\Phi, \bar{X}_{k+1|k}\right), u_{k+1}\right)\right) \\ \bar{X}^2_{k+1|k} + \bar{K}^2_{k+1}\left(Z_{k+1} - \frac{1}{m}\sum_{i=1}^{m} h\left(S_{k+1|k}\xi_i + T\left(\Phi, \bar{X}_{k+1|k}\right), u_{k+1}\right)\right) \end{bmatrix}
$$
$$
= T\left(\phi, \bar{X}_{k+1|k}\right) + \begin{bmatrix} \bar{K}^1_{k+1} \\ \bar{K}^2_{k+1} \end{bmatrix}\left(Z_{k+1} - \hat{Z}_{k+1|k}\right) + \begin{bmatrix} V_{k+1}\left(\bar{X}^2_{k+1|k+1} - \bar{X}^2_{k+1|k}\right) \\ 0 \end{bmatrix} \tag{43}
$$
$$
= T\left(\phi, \bar{X}_{k+1|k}\right) + \begin{bmatrix} \bar{K}^1_{k+1} + V_{k+1}\bar{K}^2_{k+1} \\ \bar{K}^2_{k+1} \end{bmatrix}\left(Z_{k+1} - \hat{Z}_{k+1|k}\right)
$$
$$
= X_{k+1|k} + K_{k+1}\left(Z_{k+1} - \hat{Z}_{k+1|k}\right) = X_{k+1|k+1}
$$

Eq. (43) implies that at time $k + 1$, Eq. (41) remains valid.

Evidence has been established.

5 Experiment and Analysis

This section first provides details about the experimental setup and the simulation system model to assess the effectiveness of the TSCKF method, and then utilises the TSCKF algorithm for target tracking in this system, validating the algorithm's efficacy. Afterwards, a comparison is made between the CKF and TSCKF algorithms in the identical experimental setting. The purpose is to confirm the equality of both methods.

5.1 Simulation System Model

The TSCKF algorithm is utilised to estimate the posture, velocity, and oscillation of a two-wheeled robot, which is operated by the use of an inertial sensor and a real-time Kinematic global positioning system (GPS), aiming to assess the effectiveness of the algorithm. The system model is depicted in the following manner:

$$x_{k+1} = f\left(x_k\right) + B_k b_k + \omega^x_k$$
$$b_{k+1} = b_k + \omega^b_k$$
$$z_k = h\left(x_k\right) + b_k + v_k$$

where,

$$
f\left(x_k\right) = \begin{bmatrix} X_k + \Delta t V_{l,k}\cos\left(\theta_k\right) - \Delta t V_{y,k}\sin\left(\theta_k\right) \\ Y_k + \Delta t V_{l,k}\sin\left(\theta_k\right) - \Delta t V_{y,k}\cos\left(\theta_k\right) \\ V_{l,k} + \Delta t V_{y,k}\gamma_k + \Delta t \alpha_{x,k} \\ V_{y,k} - \Delta t V_{l,k}\gamma_k + \Delta t \alpha_{y,k} \\ \theta_k + \Delta t \gamma_{m,k} \end{bmatrix}
$$

$$h\left(x_{k}\right)=\left[\begin{array}{c} X_{k} \\ Y_{k} \\ \cos\left(\theta_{k}\right)V_{l,k}-\sin\left(\theta_{k}\right)V_{y,k} \\ \sin\left(\theta_{k}\right)V_{l,k}+\cos\left(\theta_{k}\right)V_{y,k} \\ \theta_{k} \end{array}\right]$$

The state vector $x_{k}=\left[\begin{array}{ccccc} X_{k} & Y_{k} & V_{l,k} & V_{y,k} & \theta_{k} \end{array}\right]^{T}$ is unaffected by the bias vector b_{k} and initially takes on the value of $x_{0}\sim N\left(2,0.05^{2}\right)$. The measurement vector $z_{k}=\left[\begin{array}{ccccc} z_{X,k} & z_{Y,k} & z_{X',k} & z_{Y',k} & z_{\theta,k} \end{array}\right]^{T}$ consists of the precise values for position, velocity, and angle. The state and measurement noises are both zero-mean Gaussian white noises that are independent of each other, with the state noise vector being denoted as $\omega_{k}^{x}\sim N\left(0,0.5^{2}\right)$, and the measurement noise vector as $v_{k}\sim N\left(0,0.05^{2}\right)$.

The parameter $\left\{\begin{array}{ccc} \alpha_{x,k} & \alpha_{y,k} & \gamma_{m,k} \end{array}\right\}$ varies over time, reflects the acceleration on the x and y axes and the orientation at time k, and is obtained via the accelerometer and gyroscope. Δt represents the sample interval of the discrete system. The gyroscope is believed to measure the yaw rate, denoted as γ_{k}, which is equivalent to γ_{m}.

The TSCKF algorithm represents the bias equation as:

$$b_{k+1}=b_{k}+\omega_{k}^{b},\omega_{k}^{b}\sim N\left(0,0.05^{2}\right)$$

The computational hardware utilised in this simulation experiment consists of a quad-core Intel Core i5 4258U 2.4 GHz CPU, 4GB RAM, working on a Windows 7 64-bit Professional Edition operating system. The simulation programme employed is Matlab2013a.

5.2 Evaluation of the Efficacy of the TSCKF Algorithm

The simulation experiment initialised the system with mean values of 2 and a variance of 0.05 for five random integers, which takes $x_{0}=[2+0.05*\mathrm{rand}(5,1)]$ as the initial estimated value. The matrix with a variance of $P_{0}=\mathrm{diag}([0.05,0.05,0.05,0.05,0.05])$ was a 5×5 matrix with diagonal elements of 0.05 . The bias was initially assigned a value of $b_{0}=[5+0.01*\mathrm{rand}(1,1)]$, with a variance of 0.05 . The simulation lasted for a duration of 200 seconds, and the TSCKF method underwent 1,000 Monte Carlo simulations.
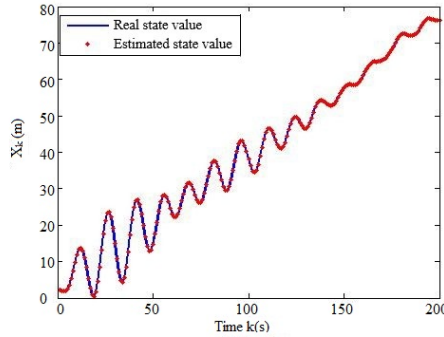
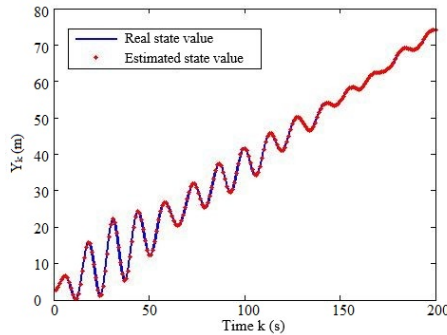**Figure 1.** Comparison of real and estimated state values of X_{k}

Figure 2. Comparison of real and estimated state values of Y_{k}
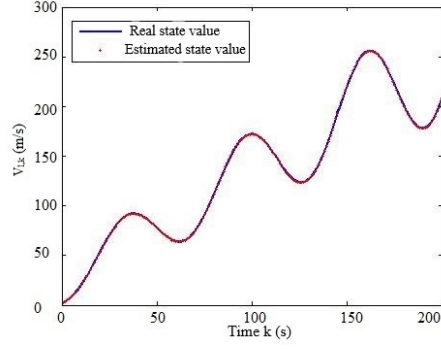
Figure 3. Comparison of real and estimated state values of $V_{l,k}$
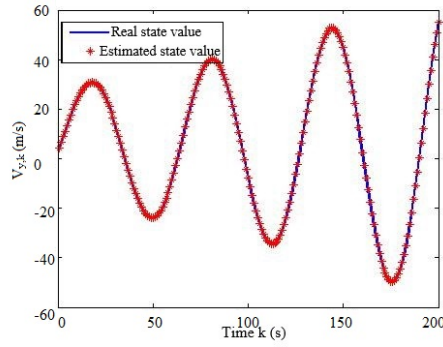

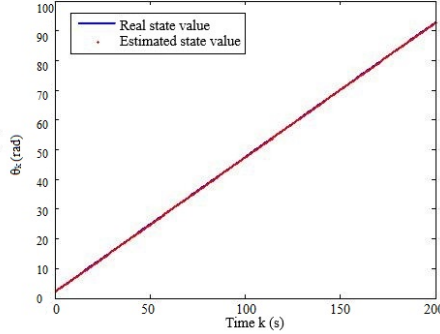**Figure 4.** Comparison of real and estimated state values of $V_{y,k}$

Figure 5. Comparison of real and estimated state values of θ_k

The estimation error for each component of the state vector is defined by Eq. (44), which quantifies the level of agreement between the estimated and real state values.

$$ERROR(k) = \left| x_{*,k} - \hat{x}_{*,k|k} \right| \tag{44}$$

The Root Mean Square Error (RMSE) is a metric used to evaluate and compare the effectiveness of different filtering algorithms.

$$RMSE(k) = \sqrt{\frac{1}{M} \sum_{i=1}^{M} \left(x_{*,k}^{(i)} - \hat{x}_{*,k|k}^{(i)} \right)^2} \tag{45}$$

where, M indicates the number of Monte Carlo simulations, whereas $x_{*,k}^{(i)}$ and $\hat{x}_{*,k|k}^{(i)}$ refer to the actual state values and estimated values of x_* in the n-th Monte Carlo simulation, respectively.
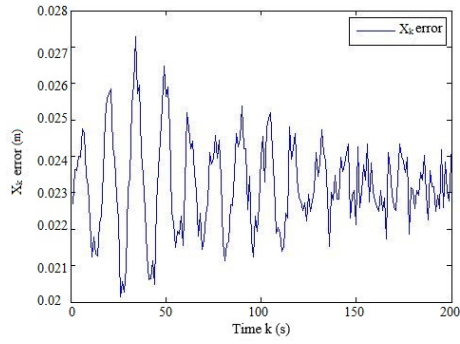
Figure 6. Error graph of X_k
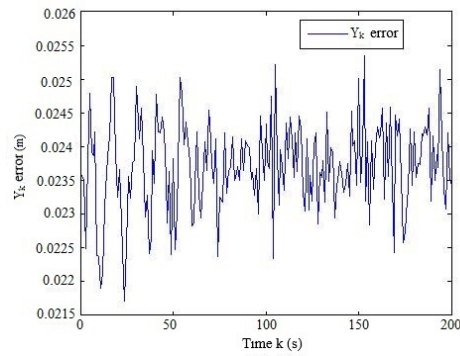
Figure 7. Error graph of Y_k
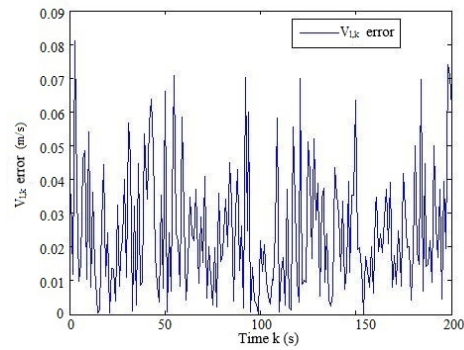
Figure 8. Error graph of $V_{y,k}$
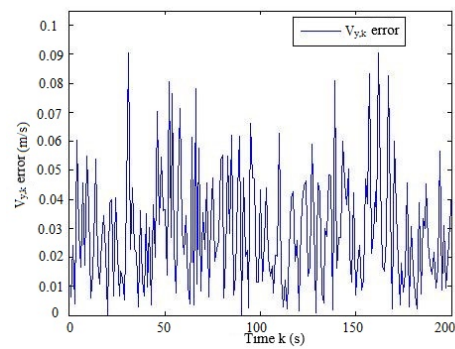
Figure 9. Error graph of $V_{l,k}$
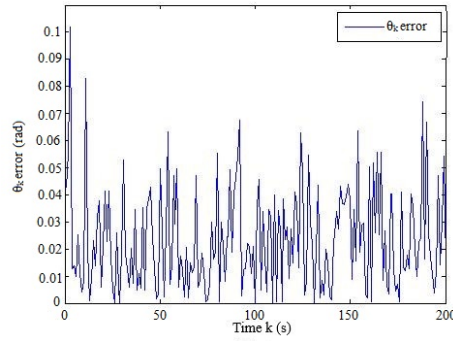
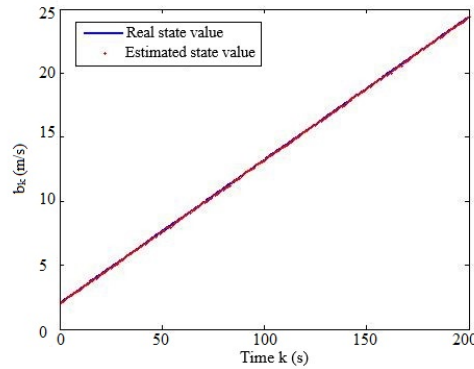**Figure 10.** Error graph of θ_k

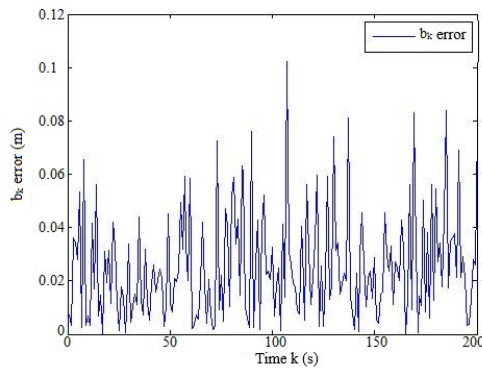**Figure 11.** Comparison of real and estimated state values of b_k

Figure 12. Error graph of b_k

Figure 1, Figure 2, Figure 3, Figure 4, Figure 5 illustrate the comparison between the actual and estimated values of the states in relation to position, velocity, and angle on the X and Y axes. These comparison charts show that the actual values are found within the area of estimated values that are most likely to be correct. This suggests that the algorithm has effective noise reduction abilities and accurate estimation.

Figure 6, Figure 7, Figure 8, Figure 9, Figure 10 depict the error graphs for each constituent of the state vector, obtained from Eq. (44). These graphs demonstrate that the error in the state vector x_k remains within an insignificant range for practical applications. This inaccuracy is actually caused by the inherent computational imperfections in digital computing. Therefore, the estimation findings are considered acceptable.

The graphs in Figure 11 and Figure 12 illustrate the comparison and error between the real and estimated values of the bias vector in the b_k state.

Figure 12 demonstrates that the error in b_k closely resembles that of the state vector x_k, staying within a small range that is insignificant in practical scenarios, thus making the results satisfactory.

To summarise, the simulation studies have confirmed that the TSCKF technique is suitable for state estimation.

5.3 Comparative Experiment between CKF and TSCKF Algorithms

The simulation system model used for this comparative experiment is the same as the one described in Section 5.1, where the state vector x_k is not influenced by the bias vector b_k, with $x_0 \sim N\left(5, 0.01^2\right)$. The state and measurement noises are zero-mean Gaussian white noises that are independent of each other. The state noise vector is denoted as $\omega_k^x \sim N\left(0, 0.02^2\right)$, while the measurement noise vector is denoted as $v_k \sim N\left(0, 0.1^2\right)$. The bias noise vector is represented by the symbol $\omega_k^b \sim N\left(0, 0.01^2\right)$.

In the simulation experiment, the initial estimated value of the system was designated as $x_0 = 5 + 0.01*$ rand $(5, 1)$, using mean values of 5 and a variance of 0.01 for five randomly generated integers. Additionally, the matrix with a variance of $P_0 = \mathrm{diag}([0.02, 0.02, 0.02, 0.02, 0.02])$ was defined as a 5×5 matrix with diagonal elements of 0.02 . The bias was initially set at a value of $b_0 = 2 + 0.02*$ rand $(1, 1)$, with a variance of 0.01 . The simulation lasted for a duration of 200 seconds, and the TSCKF method underwent 1,000 Monte Carlo simulations.
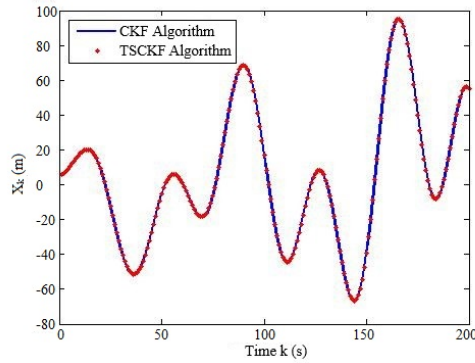

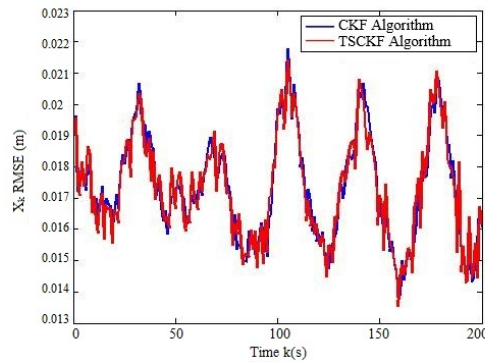**Figure 13.** Comparison of estimated values of X_k between two algorithms

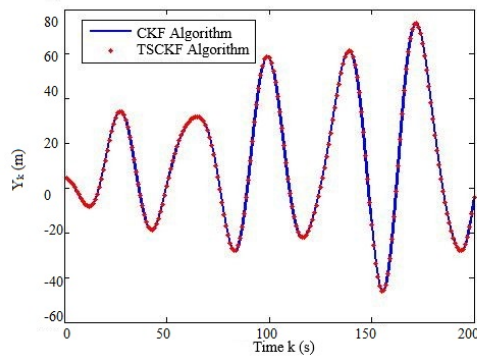**Figure 14.** Comparison of X_k RMSE between two algorithms

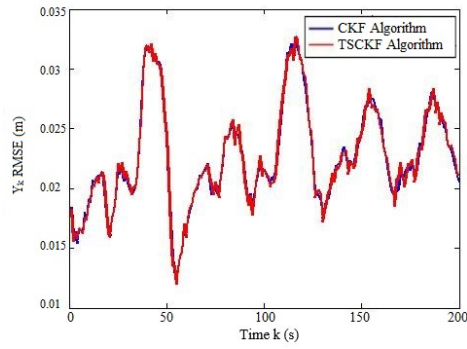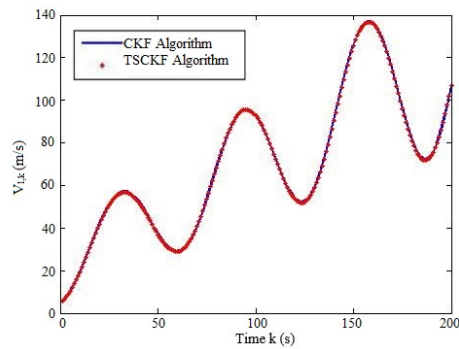**Figure 15.** Comparison of estimated values of Y_k between two algorithms

241

Figure 16. Comparison of Y_k RMSE between two algorithms

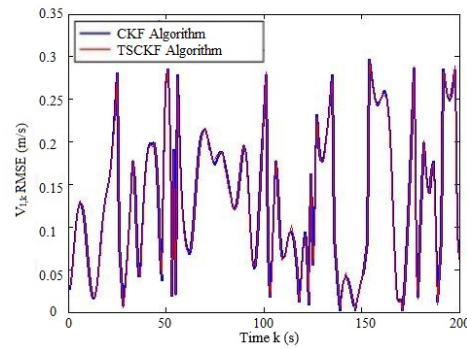**Figure 17.** Comparison of estimated values of $V_{l,k}$ between two algorithms

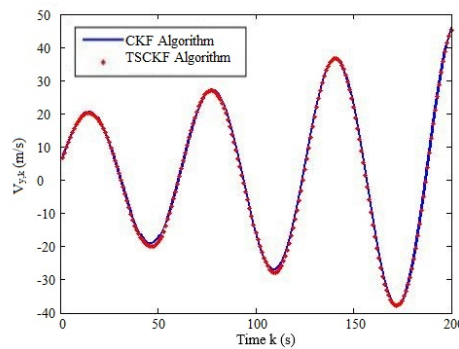**Figure 18.** Comparison of $V_{l,k}$ RMSE between two algorithms

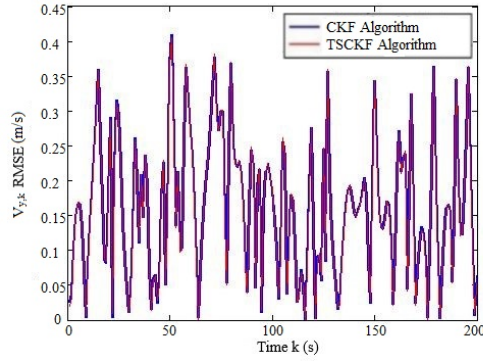**Figure 19.** Comparison of estimated values of $V_{y,k}$ between two algorithms

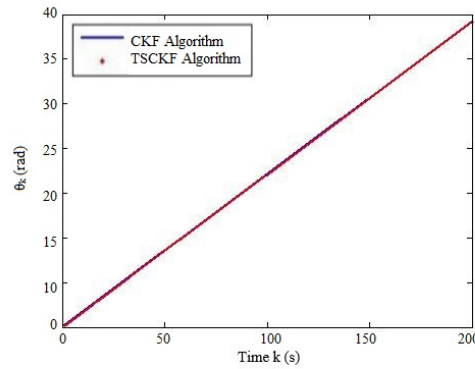**Figure 20.** Comparison of $V_{y,k}$ RMSE between two algorithms

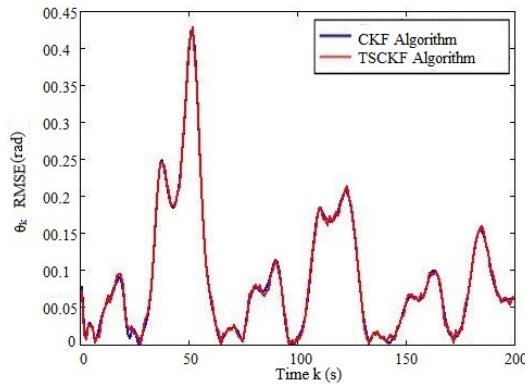**Figure 21.** Comparison of estimated values of θ_k between two algorithms

Figure 22. Comparison of θ_k RMSE between two algorithms

Figure 13, Figure 14, Figure 15, Figure 16, Figure 17, Figure 18, Figure 19, Figure 20, Figure 21, Figure 22 illustrate the comparison of estimated values and RMSE for the state components using both the CKF and TSCKF algorithms. By examining Figure 13, Figure 15, Figure 17, Figure 19, and Figure 21, it can be noted that the estimated value curves of the state vectors for both algorithms are nearly identical, thereby verifying the suitability of the CKF approach. Figure 14, Figure 16, Figure 18, Figure 20, and Figure 22 indicate that the estimation accuracy of both techniques is similar, thereby establishing their equivalency.

Figure 23 and Figure 24 display the estimation curves and RMSE comparison for the bias vector b_k of both techniques. The RMSE accuracy of both algorithms is similar, which further supports their equivalency.

The RMSE of both techniques can be determined using Eq. (45), and the results are shown in Table 1 for comparison. The data presented in this table provides quantitative evidence that both algorithms have comparable accuracy, thereby establishing their equality.
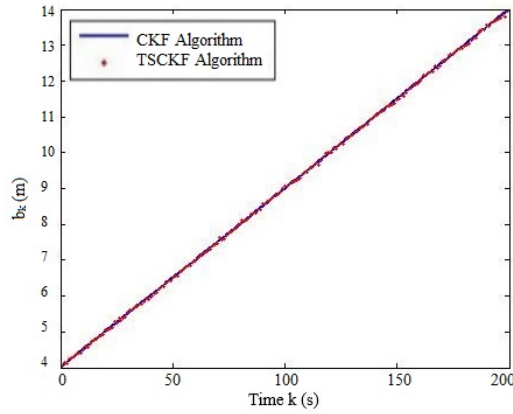
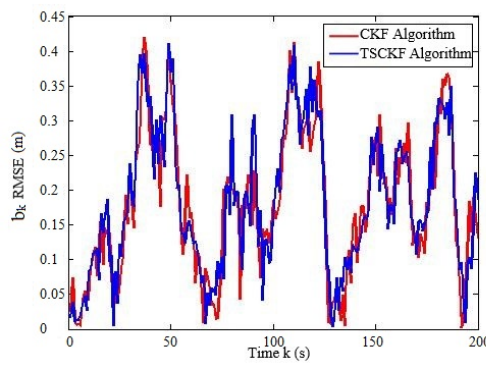**Figure 23.** Comparison of estimated values of b_k between two algorithms

Figure 24. Comparison of b_k RMSE between two algorithms

Table 1. RMSE comparison for state components

	CKF Algorithm	**TSCKF Algorithm**
X_k	0.0177	0.0185
Y_k	0.0224	0.0265
$V_{l,k}$	0.3459	0.3496
$V_{y,k}$	0.4746	0.4721
θ_k	0.0077	0.0091
b_k	0.1805	0.1790

6 Conclusion

Confronting the ubiquitous biases present in system models within practical settings, the ASCKF algorithm was initially developed. An in-depth analysis of this augmented algorithm highlighted certain limitations, leading to the proposition of the TSCKF. Central to the TSCKF algorithm is its innovative approach of block-diagonalizing the covariance matrix via matrix state transformations. This pivotal strategy effectively obviates the need for calculating cross-covariance between the state and biases, thereby substantially diminishing the computational load. As a result, this approach facilitates a smoother and more efficient filtering process. The mathematical equivalence of the TSCKF algorithm with the established CKF algorithm was rigorously demonstrated, underscoring the theoretical robustness of the proposed method. Subsequently, a series of simulation experiments were conducted to assess the practical applicability of the TSCKF algorithm. These experiments not only verified the usability of the TSCKF algorithm but also, through comparative analysis, established its equivalence with the CKF algorithm.

Data Availability

The data used to support the research findings are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Trans. Autom. Control*, vol. 54, no. 6, pp. 1254–1269, 2009. https://doi.org/10.1109/TAC.2009.2019800

[2] W. Li and Y. Jia, "Location of mobile station with maneuvers using an IMM-based cubature Kalman filter," *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4338–4348, 2011. https://doi.org/10.1109/TIE.2011.2180270

[3] C. Sheng, J. Zhao, Y. Liu, and W. Wang, "Prediction for noisy nonlinear time series by echo state network based on dual estimation," *Neurocomputing*, vol. 82, pp. 186–195, 2012. https://doi.org/10.1016/j.neucom.2011.11.021

[4] L. Zhang, M. Lv, Z. Niu, and W. Rao, "Two-stage cubature kalman filter for nonlinear system with random bias," in *2014 International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI), Beijing, China*, 2014, pp. 1–4. https://doi.org/10.1109/MFI.2014.6997720

[5] L. Zhang, W. Rao, and H. Wang, "An advance TSCKF for nonlinear system with random bias," *J. Resid. Sci. Technol.*, vol. 7, no. 13, pp. 1–8, 2016. https://doi.org/10.14257/JRST.2016.7.13.022

[6] Y. Huang, "Research on high-accuracy state estimation method and their applications to target tracking and cooperative localization," pp. 85–238, 2018.

[7] Z. Zhang, "AUV cooperative navigation method based on nonlinear filtering," pp. 19–70, 2018.

[8] B. Xu, S. Li, K. Jin, and L. Wang, "A cooperative localization method for AUVs based on RBF neural network-assisted CKF," *Acta Armamentarii*, vol. 40, no. 10, pp. 2119–2128, 2019. https://doi.org/10.3969/j.issn.1000-1093.2019.10.018

[9] Z. P. Cai, Z. X. Yu, and J. Yang, "Target tracking algorithm based on adaptive embedded CKF," *Electron. Opt. Control*, vol. 25, no. 10, pp. 1–5, 2018. https://doi.org/10.3969/j.issn.1671-637x.2018.10.001

[10] W. Zhong, "Research on optimization method of multi-AUV collaborative navigation and positioning performance in complex environment," pp. 27–82, 2020.

[11] W. Zhao, H. Zhao, D. Zou, and L. Liu, "A novel factor graph and cubature Kalman filter integrated algorithm for single-transponder-aided cooperative localization," *Entropy*, vol. 23, no. 10, p. 1244, 2021. https://doi.org/10.3390/e23101244

[12] A. Zhang, Y. Wu, C. Zhi, and R. Yang, "A novel adaptive factor-based cubature kalman filter for autonomous underwater vehicle," *J. Mar. Sci. Eng.*, vol. 10, no. 3, p. 326, 2022. https://doi.org/10.3390/jmse10030326

[13] H. Huang, J. Tang, C. Liu, B. Zhang, and B. Wang, "Variational bayesian-based filter for inaccurate input in underwater navigation," *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 8441–8452, 2021. https://doi.org/10.1109/TVT.2021.3099126

[14] X. Zhang, B. He, S. Gao, P. Mu, J. Xu, and N. Zhai, "Multiple model AUV navigation methodology with adaptivity and robustness," *Ocean Eng.*, vol. 254, p. 111258, 2022. https://doi.org/10.1016/j.oceaneng.2022.111258

[15] B. Friedland, "Treatment of bias in recursive filtering," *IEEE Trans. Automat. Contr.*, vol. 14, no. 4, pp. 359–367, 1969. https://doi.org/10.1109/TAC.1969.1099223

[16] A. T. Alouani, P. Xia, T. R. Rice, and W. D. Blair, "On the optimality of two-stage state estimation in the presence of random bias," *IEEE Trans. Automat. Contr.*, vol. 38, no. 8, pp. 1279–1283, 1993. https://doi.org/10.1109/9.233168

[17] C. S. Hsieh and F. C. Chen, "General two-stage kalman filters," *IEEE Trans. Automat. Contr.*, vol. 45, no. 4, pp. 819–824, 2000. https://doi.org/10.1109/9.847129

[18] C. S. Hsieh, "General two-stage extended kalman filters," *IEEE Trans. Automat. Contr.*, vol. 48, no. 2, pp. 289–293, 2003. https://doi.org/10.1109/TAC.2002.808483

[19] J. Xu, Y. Jing, G. M. Dimirovski, and Y. Ban, "Two-stage unscented kalman filter for nonlinear systems in the presence of unknown random bias," in *2008 American Control Conference, Seattle, WA, USA*, 2008, pp. 3530–3535. https://doi.org/10.1109/ACC.2008.4587040

[20] D. Zhou and Y. Ye, *Modern fault diagnosis and fault tolerant control*. Beijing, China: Tsing Hua University Publishing House, 2000.