Research Article

Adaptive Weight Update Algorithm for Target Tracking of UUV Based on Improved Gaussian Mixture Cubature Kalman Filter

Hongjian Wang ¹, Ying Wang ^{1,2}, Cun Li ³, Juan Li,¹ Qing Li,¹ and Xicheng Ban ¹

¹Automation College, Harbin Engineering University, Harbin 150001, China

²Economic and Technological Research Institute of State Grid Heilongjiang Electric Power Co., Ltd., Harbin 150036, China

³China Electronics Technology Instruments Co. Ltd., Qingdao 266000, China

Correspondence should be addressed to Ying Wang; bo831129@126.com and Cun Li; licun_1986@163.com

Received 4 December 2019; Accepted 16 April 2020; Published 4 June 2020

Academic Editor: Michele Scarpiniti

Copyright © 2020 Hongjian Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Gaussian mixture filter can solve the non-Gaussian problem of target tracking in complex environment by the multimode approximation method, but the weights of the Gaussian component of the conventional Gaussian mixture filter are only updated with the arrival of the measurement value in the measurement update stage. When the nonlinear degree of the system is high or the measurement value is missing, the weight of the Gauss component remains unchanged, and the probability density function of the system state cannot be accurately approximated. To solve this problem, this paper proposes an algorithm to update adaptive weights for the Gaussian components of a Gaussian mixture cubature Kalman filter (CKF) in the time update stage. The proposed method approximates the non-Gaussian noise by splitting the system state, process noise, and observation noise into several Gaussian components and updates the weight of the Gaussian components in the time update stage. The method contributes to obtaining a better approximation of the posterior probability density function, which is constrained by the substantial uncertainty associated with the measurements or ambiguity in the model. The estimation accuracy of the proposed algorithm was analyzed using a Taylor expansion. A series of extensive trials was performed to assess the estimation precision corresponding to various algorithms. The results based on the data pertaining to the lake trial of an unmanned underwater vehicle (UUV) demonstrated the superiority of the proposed algorithm in terms of its better accuracy and stability compared to those of conventional tracking algorithms, along with the associated reasonable computational time that could satisfy real-time tracking requirements.

1. Introduction

Nonlinear filtering is widely applied in the fields of target tracking, localization, navigation, signal processing, communication, and automatic control. Based on the stochastic state space mode, the core task of nonlinear filtering involves the computation of the probability density function (PDF). Bayesian estimation is usually used to handle nonlinear filtering problems, which can provide an optimal solution for the dynamic state estimation by iteratively estimating the PDF of the states [1]. However, since a multidimensional nonlinear integral has no analytic solution for nonlinear stochastic dynamic systems, most nonlinear filters cannot provide an optimal estimation. To solve the problems of state estimation for nonlinear

stochastic dynamic systems, Gaussian approximation filtering is widely applied because of its high estimation accuracy and computational efficiency.

A classic nonlinear Gaussian mixture filter is the EKF [2], which approximates the nonlinear system model using a first-order Taylor expansion. The Gaussian hypothesis is poor when the system model is highly nonlinear, and the estimation results are divergent. The core task of the Gaussian mixture filter is to compute the Gaussian weight integral, and various algorithms have been proposed to enable nonlinear filtering. Unscented Kalman filters (UKFs) [3, 4] are nonlinear filters based on the unscented transformation. In principle, a UKF is simple and easy to implement as it does not require the calculation of Jacobian matrices at each time step. An improved UKF

algorithm [5] was proposed that the modeling of target state and measurement vectors is carried out, and the UKF is integrated into the model to result in evolution of simulator. A new kind of the nonlinear state estimation algorithm [6] was proposed, which is robust to non-Gaussian noise. Furthermore, a cubature Kalman filter (CKF) [7], whose sampling approximation method is similar to that of the UKF, was proposed; in this approach, the state of the CKF approximation system and the probability density obtained using the spherical-radial cubature rule can be used to determine the propagation system state without linearizing the nonlinear model [8]. The CKF has several advantages such as excellent stability, low calculation complexity, and high estimation accuracy. To improve the performance of the CKF, Jia et al. [9] proposed a new class of CKFs having arbitrary degrees of accuracy to compute the spherical and radial integrals. Sparse-grid quadrature nonlinear filtering [10] was proposed to approximate the multidimensional integrals using weighted sparse-grid quadrature points, and such filters could achieve higher accuracy than conventional filters could. To address the problem of low accuracy of the CKF in several practical applications, a square root embedded CKF [11] was proposed, which employed an embedded cubature rule. The spherical simplex radial CKF [12] was proposed to improve the accuracy and efficiency of the traditional CKF; the simplex radial CKF calculated the spherical and radial integrals using a regular simplex, whereas the conventional CKF employed the moment matching method for the same purpose. The stochastic integration filter [13] was proposed to realize the estimation of the asymptotically exact integral by using the stochastic integration method. A robust PF-based multi-target tracking solution for passive sonar systems [14] was proposed to track an unknown time-varying number of multiple targets while keeping continuous track of such targets. The PF algorithm is unable to avoid the disadvantages of particle degradation and sample dilution; hence, the research on PFs is primarily focused on the resolution of particle degradation and sample dilution [15, 16]. To overcome the problem of particle degradation, several studies focused on the design of the importance density function [17, 18], and the authors proposed that the importance density function be designed using the EKF, UKF, and CDKF. In the phase of measurement updation, more particles are moved to the region with a higher likelihood function, which alleviates the particle degradation problem, and the precision of this approach is better than that attained using the traditional particle filter algorithm.

Most suboptimal nonlinear filtering algorithms implicate an important hypothesis condition that the system state has a single mode. However, in practical situations, several problems are more complex. A typical case is one in which the noise of the system processes or observations do not have ideal Gaussian densities. In addition, the various Gaussian approximation filtering algorithms based on Gaussian noise do not demonstrate ideal performance because of the mismatched model. One approach to resolve this problem is

the multimode approximation method [19], which involves adopting several Gaussian components to represent the Gaussian mixture density to approximate the multimode problems, yielding the so-called Gaussian sum filter. Recently, a modified interacting multiple model filter was proposed [20], which was derived for jump Markov systems with unknown process and measurement noise covariances. Using the inverse Wishart distribution as the conjugate prior of noise covariances, the system state together with the noise parameters for each mode is inferred by the variational Bayesian method.

In the literature [21], the Gaussian-sum CKF has been proposed for bearings-only tracking problems, in which the CKF demonstrates better performance than the particle filter does. Furthermore, for highly nonlinear passive tracking systems, an improved Gaussian mixture filter algorithm has been proposed in the literature [22], and the limited Gaussian mixture model has been used to approximate the posterior density of the state and process noises and measurement noises. However, in all these methods, while propagating the uncertainty through a nonlinear system, the weights of the Gaussian components are not regular and are updated only in the measurement update stage. This assumption is valid if the system is marginally nonlinear or the measurements are precise and frequently available. Terejanu et al. [23, 24] proposed a new Gaussian sum filter by adapting the weights of the Gaussian components for the time and measurement update stages; the proposed approach could achieve a better approximation of the posterior probability density function.

In summary, when the conventional Gaussian mixture filter transfers the Gaussian component by using the state transfer model in the time update phase, the weight of the Gaussian component remains unchanged, and the weight of the Gaussian component is only updated with the arrival of the measurement value in the measurement update phase. When the equivalent measurement value is not enough to satisfy the real-time update, the state estimation accuracy will decrease if the weight remains unchanged. This paper proposes an adaptive weight update scheme of the Gaussian components for the Gaussian mixture filter in the time update stage. This method contributes to obtaining a better approximation of the posterior probability density function, which is constrained by large uncertainty in the measurements or ambiguity in the model. The Gaussian mixture filter is improved through combination with the CKF. The Gaussian components are predicted and updated using a CKF with the results merged and weighted. The estimation accuracy of the proposed algorithm is analyzed using a Taylor expansion. Section 2 describes the principles of the Gaussian mixture CKF algorithm. In Section 3, the method of the Gaussian component weight updating is introduced, and the IGM-CKF algorithm is described in Section 4. The results of a simulation experiment conducted to compare the various filtering algorithms are presented in Section 5. Finally, Section 6 presents the results of the trials performed to verify the feasibility of the proposed UUV IGM-CKF-based navigation algorithm.

2. The Gaussian Mixture Cubature Kalman Filter (GM-CKF)

Consider a nonlinear system model with non-Gaussian noise, as follows:

$$\begin{cases} x_k = f(x_{k-1}) + w_k, \\ z_k = h(x_k) + v_k. \end{cases} \quad (1)$$

The prior and posterior densities can be defined using a Gaussian sum. The GM-CKF was designed by combining the Gaussian mixture density distribution model with the CKF. The algorithm for the GM-CKF can be described as follows.

The probability density function for the initial system state (1) can be defined as

$$p(x_0) = \sum_{i=1}^I \omega_0^s(i) N(x_0(i); \hat{x}_0(i), P_0(i)). \quad (2)$$

The state and covariance can be propagated using the CKF time update stage as follows:

$$\hat{x}_{k|k-1}(r) = \sum_{c=1}^m \omega_c \xi_{c,k|k-1}(i) + \bar{w}_k(j), \quad (3)$$

$$\begin{aligned} P_{k|k-1}(r) &= \sum_{c=1}^m \omega_c \xi_{c,k|k-1}(i) \xi_{c,k|k-1}^T(i) + Q_k(j) \\ &\quad - [\hat{x}_{k|k-1}(r) - \bar{w}_k(j)] [\hat{x}_{k|k-1}(r) - \bar{w}_k(j)]^T, \end{aligned} \quad (4)$$

$$\omega_{k|k-1}^t(r) = \omega_{k|k-1}^s(i) \cdot \omega_k^p(j), \quad (5)$$

$$\omega_{k|k-1}^s(i) = \omega_{k-1|k-1}^s(i), \quad (6)$$

where m is the number of cubature points; $r = (i-1)I + j$; I denotes the number of Gaussian components in the system state decomposition; j denotes the number of Gaussian components decomposed by process noise of the system; $\omega_{k|k-1}^t(r)$ denotes the weight of the Gaussian components after the time update; $\omega_k^p(j)$ is the weight of the j -th Gaussian noise of the process noise; $\omega_{k-1|k-1}^s(i)$ is the weight of the i -th state Gaussian component at time $k-1$; $\omega_{k|k-1}^s(i)$ denotes the weight after the propagation; and $\bar{w}_k$ denotes the covariance error for the covariance error matrix Q_k .

The state and covariance are updated in the CKF measurement update stage. The final outputs of the filter and the weights of the components are updated as follows:

$$\hat{x}_k = \sum_{n=1}^{I \cdot J \cdot L} \omega_k(n) \hat{x}_k(n), \quad (7)$$

$$P_k = \sum_{n=1}^{I \cdot J \cdot L} \omega_k(n) (P_k(n) + [\hat{x}_k(n) - \hat{x}_k] [\hat{x}_k(n) - \hat{x}_k]^T), \quad (8)$$

where $\omega_k(n)$ is the weighted value of the n -th Gaussian component, which can be updated as follows:

$$\omega_k(n) = \frac{\omega_{k|k-1}^t(r) \omega_k^m(l) p(z_k | x_k, n)}{\sum_{r=1}^{I \cdot J} \sum_{l=1}^L \omega_{k|k-1}^t(r) \omega_k^m(l) p(z_k | x_k, n)}, \quad (9)$$

where $\omega_k^m(l)$ is the weight of the l -th Gaussian component of the observation noise. $p(z_k | x_k, n)$ is the observation likelihood distribution of the n -th Gaussian component and can be calculated as follows:

$$p(z_k | x_k, n) = \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp \left[-\frac{1}{2} \left(\frac{z_k - \hat{z}_{k|k-1}(r, l)}{\sigma_n} \right)^2 \right]. \quad (10)$$

Note that the weights $\omega_{k|k-1}^s(i)$ do not change during the propagation from (7), which is acceptable if the system has a precise model. The reason that the weight remains constant is that the covariance is assumed to be sufficiently small. This aspect is a problem particularly when the system model has a large uncertainty and the measurements are not frequently available.

3. Adaptive Weight Update

Consider the nonlinear system (1) with the probability density function of the initial conditions being $p(x_0)$. The Gaussian mixture approximation of the probability density function is defined as

$$\hat{p}(x_k) = \sum_{i=1}^I \omega_k^s(i) N(x_k(i); \hat{x}_k(i), P_k(i)). \quad (11)$$

The true probability density function of the system state can be defined using the Chapman-Kolmogorov formula [25]:

$$p(x_k) = \int_{\mathbb{R}^n} p(x_k | x_{k-1}) p(x_{k-1}) dx_{k-1}. \quad (12)$$

The mean-square optimal new weights can be obtained by minimizing the following integral square difference between the true probability $p(x_k)$ and the approximated one $\hat{p}(x_k)$ by using the least squares algorithm:

$$w_{k|k-1} = \arg \min \frac{1}{2} \int_{\mathbb{R}^n} (p(x_k) - \hat{p}(x_k))^2 dx_k, \quad (13)$$

where $w_{k|k-1} = [w_{k|k-1}(1) \ w_{k|k-1}(2) \ \dots \ w_{k|k-1}(I)]^T$ denotes the vector of the weights of every Gaussian component at time k . To resolve formula (13), the cost function can be defined as follows [26]:

$$J(w_{k|k-1}) = J^{PP}(w_{k|k-1}) - 2J^{P\hat{P}}(w_{k|k-1}) + J^{\hat{P}\hat{P}}(w_{k|k-1}), \quad (14)$$

where the first term denotes the self-likeness of the true probability density function of the system state and $J^{PP}(w_{k|k-1}) = \int_{\mathbb{R}^n} p(x_k) p(x_k) dx_k$. The second term denotes the cross likeness of the true probability and the approximation, and $J^{P\hat{P}}(w_{k|k-1}) = \int_{\mathbb{R}^n} p(x_k) \hat{p}(x_k) dx_k$. The last term represents the self-likeness of the approximation probability of the system state, and $J^{\hat{P}\hat{P}}(w_{k|k-1}) = \int_{\mathbb{R}^n} \hat{p}(x_k) \hat{p}(x_k) dx_k$.

It is assumed that the Gaussian mixture approximation is equal to the true probability density function at time $k-1$; specifically, $\hat{p}(x_{k-1}) \approx p(x_{k-1})$, and the proof procedure is as follows.

Substitute formula (11) into formula (12) and the following relationship can be given:

$$\begin{aligned}
p(x_k) &= \sum_{i=1}^I \omega_{k-1}^s(i) \int_{\mathbb{R}^n} N(x_{k-1}(i); \hat{x}_{k-1}(i), P_{k-1}(i)) N \\
&\quad \cdot (x_k; f(x_{k-1}) + w_{k-1}, Q_{k-1}) dx_{k-1} \\
&= \sum_{i=1}^I \omega_{k-1}^s(i) \int_{\mathbb{R}^n} N(x_{k-1}(i); \hat{x}_{k-1}(i), P_{k-1}(i)) N \\
&\quad \cdot (x_k; (x_{k-1} - \hat{x}_{k-1}(i)) + f(x_{k-1}) + w_k, Q_k) dx_{k-1} \\
&\quad + \sum_{i=1}^I \omega_{k-1}^s(i) \int_{\mathbb{R}^n} \varepsilon_{k-1} dx_{k-1},
\end{aligned} \tag{15}$$

where $\varepsilon_{k-1} = N(x_{k-1}(i); \hat{x}_{k-1}(i), P_{k-1}(i)) [N(x_k; f(x_{k-1}) + w_{k-1}, Q_{k-1}) - N(x_k; (x_{k-1} - \hat{x}_{k-1}(i)) + f(x_{k-1}) + w_k, Q_k)]$. Assuming that all covariance $p_{k-1}(i)$ of the Gaussian components is small enough so that the linearization around a mean is representative for the dynamics in the vicinity of the respective mean and that there are sufficient number of Gaussian components, and then $p_{k-1}(i) \rightarrow 0$ implies $\int_{\mathbb{R}^n} \varepsilon_{k-1} dx_{k-1} \rightarrow 0$. $p(x_{k-1})$ can be rewritten as

$$\begin{aligned}
p(x_{k-1}) &\approx \sum_{i=1}^I \omega_{k-1}^s(i) \int_{\mathbb{R}^n} N(x_{k-1}(i); \hat{x}_{k-1}(i), P_{k-1}(i)) N \\
&\quad \cdot (x_k; (x_{k-1} - \hat{x}_{k-1}(i)) + f(x_{k-1}) + w_k, Q_k) dx_{k-1} \\
&= \sum_{i=1}^I \omega_{k-1}^s(i) N(x_{k-1}; f(x_{k-1}) + w_{k-1}, Q_{k-1}) \\
&= \sum_{i=1}^I \omega_{k-1}^s(i) N(x_k(i); \hat{x}_k(i), P_k(i)) \\
&= \hat{p}(x_{k-1}).
\end{aligned} \tag{16}$$

The derivation of the terms of the cost function can be expressed as formula (16):

$$\begin{aligned}
J^{PP}(w_{k|k-1}) &= \int_{\mathbb{R}^n} p(x_k)^2 dx_k \\
&= \int_{\mathbb{R}^n} \left[\int_{\mathbb{R}^n} p(x_k | x_{k-1}) p(x_{k-1}) dx_{k-1} \right]^2 dx_k \\
&\approx \int_{\mathbb{R}^n} \left[\int_{\mathbb{R}^n} p(x_k | x_{k-1}) \hat{p}(x_{k-1}) dx_{k-1} \right]^2 dx_k \\
&= \int_{\mathbb{R}^n} \left[\int_{\mathbb{R}^n} N(x_k; f(x_{k-1}) + \bar{w}_k, Q_k) \hat{p} \right. \\
&\quad \cdot (x_{k-1}) dx_{k-1} \left. \right]^2 dx_k \\
&= w_{k-1}^T J^{PP} w_{k-1}.
\end{aligned} \tag{17}$$

Similarly,

$$J^{\hat{P}\hat{P}}(w_{k|k-1}) = w_{k|k-1}^T J^{\hat{P}\hat{P}} w_{k-1}, \tag{18}$$

$$J^{\hat{P}\hat{P}}(w_{k|k-1}) = w_{k|k-1}^T J^{\hat{P}\hat{P}} w_{k|k-1}. \tag{19}$$

The elements of the matrices are given as follows:

$$\begin{aligned}
J_{ij}^{PP} &= \int \left[\int N(x_k; f(x_{k-1}) + \bar{w}_k, Q_k) \cdot N(x_{k-1}(i); \hat{x}_{k-1}(i), P_{k-1}(i)) dx_{k-1} \right]^2 dx_k \\
&= \int \left(E_{N(x_{k-1}(i); \hat{x}_{k-1}(i), P_{k-1}(i))} [N(x_k; f(x_{k-1}) + \bar{w}_k, Q_k)] \right)^2 dx_k,
\end{aligned} \tag{20}$$

$$\begin{aligned}
J_{ij}^{\hat{P}\hat{P}} &= \int N(f(x_{k-1}); x_{k|k-1}(i), P_{k|k-1}(i) + Q_{k-1}) \cdot N(x_{k-1}(j); \hat{x}_{k-1}(j), P_{k-1}(j)) dx_{k-1} \\
&= E_{N(x_{k-1}(j); \hat{x}_{k-1}(j), P_{k-1}(j))} [N(f(x_{k-1}); x_{k|k-1}(i), P_{k|k-1}(i) + Q_{k-1})] \\
&= \sum_{c=1}^m \omega_c N(\xi_{c,k|k-1}; x_{k|k-1}(i), P_{k|k-1}(i)),
\end{aligned} \tag{21}$$

where (ω_c, ξ_c) denotes are the cubature points.

$$\begin{aligned}
J_{ij}^{\hat{P}\hat{P}} &= N(x_{k|k-1}(i); x_{k|k-1}(j), P_{k|k-1}(i) + P_{k|k-1}(j)) \\
&= \left[2\pi(P_{k|k-1}(i) + P_{k|k-1}(j)) \right]^{-1/2} \\
&\quad \cdot \exp \left[-\frac{1}{2} (x_{k|k-1}(i) - x_{k|k-1}(j))^T \right. \\
&\quad \cdot (P_{k|k-1}(i) + P_{k|k-1}(j))^{-1} (x_{k|k-1}(i) - x_{k|k-1}(j)) \left. \right].
\end{aligned} \tag{22}$$

The first term in the cost function (13) is not required in the optimization process, and it is only used to represent an overall magnitude of the uncertainty propagation error [25]. According to the aforementioned relations, the final formulation of formula (13) can be expressed as follows:

$$w_{k|k-1} = \arg \min \frac{1}{2} w_{k|k-1}^T J^{\hat{P}\hat{P}} w_{k|k-1} - w_{k|k-1}^T J^{PP} w_{k-1}, \tag{23}$$

where $1^T w_{k|k-1} = 1, w_{k|k-1} \geq 0$.

Based on the above derivation, the summary of the IGM-CKF with the adaptive weight update can be presented as in Table 1.

4. Analysis of the Estimation Accuracy of IGM-CKF

The estimation accuracy of the UKF [27] and CKF [28] has been discussed using the Taylor expansion. Using a similar approach, the performance and estimation accuracy of the proposed algorithm were further examined and evaluated mathematically using the Taylor expansion, as described in this section.

4.1. Time Update. To evaluate the proposed estimation accuracy, the IGM-CKF is expanded into the Taylor terms. Considering the Gaussian component $x_{k-1|k-1}(r) \sim N(\bar{x}_{k-1|k-1}(r), P_{k-1|k-1}(r))$ and defining $(\tilde{x}_{k-1|k-1}(r) = x_{k-1|k-1}(r) - \bar{x}_{k-1|k-1}(r)) \sim N(0, P_{k-1|k-1}(r))$, the state transition function $f(x_{k-1|k-1}(r))$ can be expanded into a Taylor series as follows:

$$\begin{aligned} f(x_{k-1|k-1}(r)) &= f(\bar{x}_{k-1|k-1}(r) + \tilde{x}_{k-1|k-1}(r)) \\ &= f(\bar{x}_{k-1|k-1}(r)) + T_{\tilde{x}_{k-1|k-1}(r)} f \\ &\quad + \frac{T_{\tilde{x}_{k-1|k-1}(r)}^2 f}{2!} + \frac{T_{\tilde{x}_{k-1|k-1}(r)}^3 f}{3!} + \frac{T_{\tilde{x}_{k-1|k-1}(r)}^4 f}{4!} + \dots, \end{aligned} \quad (24)$$

Where $T_{\tilde{x}_{k-1|k-1}(r)} f = [(\tilde{x}_{k-1|k-1}^T(r) \nabla) f(x)^T]^T$ and ∇ denotes the differential of $f(x)$.

The estimated value of the cubature points is

$$\begin{aligned} \xi_{c,k|k-1}(i) &= f(\xi_{c,k-1}(i)) = f(\bar{x}_{k-1|k-1}(r) + \tilde{x}_{k-1|k-1}(r)) \\ &= f(\bar{x}_{k-1|k-1}(r)) + T_{\tilde{x}_{k-1|k-1}(r)} f + \frac{T_{\tilde{x}_{k-1|k-1}(r)}^2 f}{2!} \\ &\quad + \frac{T_{\tilde{x}_{k-1|k-1}(r)}^3 f}{3!} + \frac{T_{\tilde{x}_{k-1|k-1}(r)}^4 f}{4!} + \dots \end{aligned} \quad (25)$$

The predicted value of the Gaussian component is

$$\begin{aligned} x_{k|k-1}(r) &= \sum_{c=1}^m \omega_c \xi_{c,k|k-1}(i) = \sum_{c=1}^m \omega_c f(\bar{x}_{k-1|k-1}(r) + \tilde{x}_{k-1|k-1}(r)) \\ &= \sum_{c=1}^m \omega_c \left(f(\bar{x}_{k-1|k-1}(r)) + T_{\tilde{x}_{k-1|k-1}(r)} f \right. \\ &\quad \left. + \frac{T_{\tilde{x}_{k-1|k-1}(r)}^2 f}{2!} + \frac{T_{\tilde{x}_{k-1|k-1}(r)}^3 f}{3!} + \frac{T_{\tilde{x}_{k-1|k-1}(r)}^4 f}{4!} + \dots \right). \end{aligned} \quad (26)$$

All of the odd moments in the formula sum to 0 because of the symmetrisation of $\tilde{x}_{k|k-1}(r)$. Therefore, the formula can be rewritten as

$$\begin{aligned} x_{k|k-1}(r) &= f(\bar{x}_{k-1|k-1}(r)) + \sum_{c=1}^m \omega_c \\ &\quad \cdot \left(\frac{T_{\tilde{x}_{k-1|k-1}(r)}^2 f}{2!} + \frac{T_{\tilde{x}_{k-1|k-1}(r)}^4 f}{4!} + \dots \right). \end{aligned} \quad (27)$$

To simplify, we retain the second-order term of the formula by neglecting the higher-order terms:

$$x_{k|k-1}(r) = f(\bar{x}_{k-1|k-1}(r)) + E \left[\frac{T_{\tilde{x}_{k-1|k-1}(r)}^2 f}{2!} \right]. \quad (28)$$

According to the definition of the covariance matrix, the second-order term of the formula is

$$E \left[\frac{T_{\tilde{x}_{k-1|k-1}(r)}^2 f}{2!} \right] = \frac{\nabla^T P_{k-1|k-1}(r) \nabla f}{2!}. \quad (29)$$

Combining formulas (25) and (28), the covariance matrix can be determined as

$$\begin{aligned} P_{k|k-1}(r) &= E \left[T_{\tilde{x}_{k|k-1}(r)} f \left(T_{\tilde{x}_{k|k-1}(r)} f \right)^T \right. \\ &\quad + \frac{T_{\tilde{x}_{k|k-1}(r)} f \left(T_{\tilde{x}_{k|k-1}(r)}^3 f \right)^T}{3!} \\ &\quad + \frac{T_{\tilde{x}_{k|k-1}(r)}^2 f \left(T_{\tilde{x}_{k|k-1}(r)}^2 f \right)^T}{2 \times 2!} \\ &\quad \left. + \frac{T_{\tilde{x}_{k|k-1}(r)}^3 f \left(T_{\tilde{x}_{k|k-1}(r)} f \right)^T}{3!} \right] \\ &\quad - E \left[\frac{T_{\tilde{x}_{k|k-1}(r)}^2 f}{2!} \right] E \left[\frac{T_{\tilde{x}_{k|k-1}(r)}^2 f}{2!} \right]^T + Q_k(j). \end{aligned} \quad (30)$$

Similar to the case of the EKF, we define $T_{\tilde{x}} f = \Phi_k \tilde{x}$, where Φ_k is the Jacobian matrix of $f(x)$, and the formula can be rewritten as

TABLE 1: Computation steps of the IGM-CKF.

Time update
$\hat{x}_{k k-1}(r) = \sum_{c=1}^m \omega_c \xi_{c,k k-1}(i) + \bar{w}_k(j)$
$P_{k k-1}(r) = \sum_{c=1}^m \omega_c \xi_{c,k k-1}(i) \xi_{c,k k-1}^T(i) - [\hat{x}_{k k-1}(r) - \bar{w}_k(j)][\hat{x}_{k k-1}(r) - \bar{w}_k(j)]^T + Q_k(j)$
$w_{k k-1} = \arg \min (1/2) w_{k k-1}^T J^{PP} w_{k k-1} - w_{k k-1}^T J^{P\hat{P}} w_{k-1}$
Measurement update
$P_{zz,k k-1}(r, l) = \sum_{c=1}^m \omega_c \xi_{c,k k-1}(r) \xi_{c,k k-1}^T(r) - [\hat{z}_{k k-1}(r) - \bar{v}_k(l)][\hat{z}_{k k-1}(r) - \bar{v}_k(l)]^T + R_k(l)$
$P_{xz,k k-1}(r, l) = \sum_{c=1}^m \omega_c \xi_{c,k k-1}(r) \xi_{c,k k-1}^T(r) - \hat{x}_{k k-1}(r)[\hat{z}_{k k-1}(r) - \bar{v}_k(l)]^T$
$K_k(r, l) = P_{xz,k k-1}(r, l) P_{zz,k k-1}^{-1}(r, l)$
$\hat{x}_k(n) = \hat{x}_{k k-1}(r) + K_k(r, l)[\hat{z}_k - \bar{z}_{k k-1}(r, l)]$
$P_k(n) = P_{k k-1}(r) - K_k(r, l) P_{zz,k k-1}(r, l) K_k^T(r, l)$
Filter output
Formula (3)~(8)

$$\begin{aligned}
P_{k|k-1}(r) &= \Phi_k P_{k-1|k-1}(r) \Phi_k^T \\
&+ E \left[\frac{T_{\tilde{x}_{k-1|k-1}}(r) f \left(T_{\tilde{x}_{k-1|k-1}}^3(r) f \right)^T}{3!} \right. \\
&+ \frac{T_{\tilde{x}_{k-1|k-1}}^2(r) f \left(T_{\tilde{x}_{k-1|k-1}}^2(r) f \right)^T}{2 \times 2!} \\
&+ \left. \frac{T_{\tilde{x}_{k-1|k-1}}^3(r) f \left(T_{\tilde{x}_{k-1|k-1}}(r) f \right)^T}{3!} \right] \\
&- \left[\frac{\nabla^T P_{k-1|k-1}(r) \nabla f}{2!} \right] \left[\frac{\nabla^T P_{k-1|k-1}(r) \nabla f}{2!} \right]^T + Q_k(j).
\end{aligned} \tag{31}$$

The aforementioned formulas indicate that the proposed algorithm can propagate the second-order and higher-order terms of the nonlinear system. The higher-order terms of each Gaussian component are regarded as 0 when the system is linear, which is the same as in the EKF.

4.2. Measurement Update. Similar to the time update stage described in Section 4.1, we define $(\tilde{x}_{k|k-1}(r) = x_{k|k-1}(r) -$

$\bar{x}_{k|k-1}(r)) \sim N(0, P_{k|k-1}(r))$ and $x_{k|k-1}(r) \sim N(\bar{x}_{k|k-1}(r), P_{k|k-1}(r))$. The measurement function $h(x_{k|k-1}(r))$ can be expanded into a Taylor series as follows:

$$\begin{aligned}
h(x_{k|k-1}(r)) &= h(\bar{x}_{k|k-1}(r) + \tilde{x}_{k|k-1}(r)) \\
&= h(\bar{x}_{k|k-1}(r)) + T_{\tilde{x}_{k|k-1}}(r) h + \frac{T_{\tilde{x}_{k|k-1}}^2(r) h}{2!} \\
&+ \frac{T_{\tilde{x}_{k|k-1}}^3(r) h}{3!} + \frac{T_{\tilde{x}_{k|k-1}}^4(r) h}{4!} + \dots
\end{aligned} \tag{32}$$

The estimated value of the cubature points is

$$\begin{aligned}
\zeta_{c,k|k-1}(r) &= h(\xi_{c,k|k-1}(r)) = h(\bar{x}_{k|k-1}(r) + \tilde{x}_{k|k-1}(r)) \\
&= h(\bar{x}_{k|k-1}(r)) + T_{\tilde{x}_{k|k-1}}(r) h + \frac{T_{\tilde{x}_{k|k-1}}^2(r) h}{2!} \\
&+ \frac{T_{\tilde{x}_{k|k-1}}^3(r) h}{3!} + \frac{T_{\tilde{x}_{k|k-1}}^4(r) h}{4!} + \dots
\end{aligned} \tag{33}$$

Analogous to the calculation in the time update stage, the self-correlation covariance matrix is calculated as

$$\begin{aligned}
P_{zz,k|k-1}(r, l) &= E \left[T_{\tilde{x}_{k|k-1}}(r) h \left(T_{\tilde{x}_{k|k-1}}(r) h \right)^T + \frac{T_{\tilde{x}_{k|k-1}}^2(r) h \left(T_{\tilde{x}_{k|k-1}}^3(r) h \right)^T}{3!} \right. \\
&+ \frac{T_{\tilde{x}_{k|k-1}}^2(r) h \left(T_{\tilde{x}_{k|k-1}}^2(r) h \right)^T}{2!} + \left. \frac{T_{\tilde{x}_{k|k-1}}^3(r) h \left(T_{\tilde{x}_{k|k-1}}(r) h \right)^T}{3!} \right] \\
&- E \left[\frac{T_{\tilde{x}_{k|k-1}}^2(r) h}{2!} \right] E \left[\frac{T_{\tilde{x}_{k|k-1}}^2(r) h}{2!} \right]^T + R_k(l).
\end{aligned} \tag{34}$$

Considering that H_k is the Jacobian matrix of $h(\cdot)$, the formula can be rewritten as

$$P_{zz,k|k-1}(r, l) = H_k P_{k|k-1}(r) H_k^T + E \left[\frac{T_{\tilde{x}_{k|k-1}(r)} h \left(T_{\tilde{x}_{k|k-1}(r)}^3 h \right)^T}{3!} + \frac{T_{\tilde{x}_{k|k-1}(r)}^2 h \left(T_{\tilde{x}_{k|k-1}(r)}^2 h \right)^T}{2!} + \frac{T_{\tilde{x}_{k|k-1}(r)}^3 h \left(T_{\tilde{x}_{k|k-1}(r)} h \right)^T}{3!} \right] - E \left[\frac{\nabla^T P_{k|k-1}(r) \nabla}{2!} h \right] E \left[\frac{\nabla^T P_{k|k-1}(r) \nabla}{2!} h \right]^T + R_k(l). \quad (35)$$

Similarly, the cross correlation covariance matrix can be calculated as

$$P_{xz,k|k-1}(r, l) = P_{k|k-1}(r) H_k^T + E \left[\hat{x}_{k|k-1}(r) \frac{D_{\tilde{x}_{k|k-1}(r)}^3 h}{3!} \right]^T. \quad (36)$$

The Kalman gain can be calculated using the following expression:

$$K_k(r, l) = P_{xz,k|k-1}(r, l) P_{zz,k|k-1}^{-1}(r, l). \quad (37)$$

The posterior state and covariance matrix can be calculated as

$$\begin{cases} \hat{x}_k(n) = \hat{x}_{k|k-1}(r) + K_k(r, l) [\hat{z}_k - \hat{z}_{k|k-1}(r, l)], \\ P_k(n) = P_{k|k-1}(r) - K_k(r, l) P_{zz,k|k-1}(r, l) K_k^T(r, l). \end{cases} \quad (38)$$

The above deduction indicates that the IGM-CKF can approximate the state that is related not only to the first-order terms but also to the higher-order terms. This finding demonstrates that the IGM-CKF can capture the higher-order terms of the nonlinear function.

5. IGM-CKF Simulation

Next, we compared the performance of the GM-EKF, GM-UKF, GM-CKF, and IGM-CKF with the naïve model:

$$\begin{cases} x_k = 0.01x_{k-1}^2 + 0.2x_{k-1} + 8 \cos(1.2(k-1)) + w_{k-1}, \\ z_k = x_k^2 + v_k, \end{cases} \quad (39)$$

where w_{k-1} is the zero-mean Gaussian process noise with variance Q . v_k is the glint noise, which can be modeled as the mixing form of the Gaussian and Laplace distributions or a combination of several Gaussian distributions with different covariances; in the simulation performed in this study, the glint noise was modeled as the additive sum of two Gaussian distributions with different covariances.

$$p(v) = (1 - \varepsilon)N(v; u_1, R_1) + \varepsilon N(v; u_2, R_2), \quad (40)$$

where ε is the glint frequency factor and $\varepsilon = 0.25$.

To implement a fair and reasonable performance comparison of different filter algorithms, an important requirement was that all the filters undergo the same initialization with an initial state of $x_0 = 1$, $Q = 10$, $R_1 = 0.1$, and $R_2 = 1$, along with an initial state covariance of $P_0 = 10$. In addition, the parameters α , β , and κ of the GM-UKF were selected as 1, 2, and 0, respectively, and five sigma points were present in the unscented transformation. The number of cubature points in the GM-CKF was selected to be 2, resulting in a dimension of state of 1. The number of simulation steps was set as 100.

The system state was split into 4 Gaussian components in the IGM-CKF algorithm, and each component was generated around the state value along with a random P_0 . The initial weight value of each component was determined using

$$p(x_0) = \sum_{i=0}^4 \omega_0(i) N(x_0; \mu_0(i), P_0(i)). \quad (41)$$

The process noise was selected as a single component, with a value given of $Q = 10$, and the observation noise was split into 2 Gaussian components. Under the given conditions, the number of state Gaussian components increased from 4 to 8 when one recursion was completed. The redundant components were merged, and thus the total number of components remained 4. Figures 1 and 2 show the estimation results and a comparison of the errors associated with the various filtering algorithms.

The accuracy achieved using the filtering methods was analyzed in terms of the following metrics.

The root-mean-square error (RMSE) is defined as

$$\text{RMSE} = \sqrt{\frac{1}{t_{\text{step}}} \sum_{k=1}^{t_{\text{step}}} (x_k - \hat{x}_k)^2}, \quad (42)$$

where t_{step} is the simulation step time and x_k and $\hat{x}_k$ denote the state and estimate values at time k , respectively.

The average noncredibility index (NCI) [29, 30] is defined as

FIGURE 1: Various filtering algorithm results.

FIGURE 2: The error of various filtering algorithms.

$$\text{NCI} = \frac{1}{t_{\text{step}}} \sum_{k=1}^{t_{\text{step}}} \left[10 \ln \left(\frac{(x_k - \hat{x}_k)^2}{P_{k|k}} \right) - 10 \ln \left(\frac{(x_k - \hat{x}_k)^2}{\text{RMSE}} \right) \right], \quad (43)$$

where $P_{k|k}$ is the covariance of the estimation provided by the filter and RMSE denotes the mean-square error of the estimation, as calculated using (42).

The RMSE and NCI values for various filtering algorithm calculations are presented in Table 2.

From Figures 1 and 2 and Table 2, it can be observed that the estimation performance of the GM-EKF is the worst and prone to fluctuations. The simulation results indicate that the GM-CKF and IGM-CKF exhibit the same estimation precision. The RMSE values indicate that the precision of the

TABLE 2: RMSE of various filtering algorithms.

Algorithm	GM-EKF	GM-UKF	GM-CKF	IGM-CKF
RMSE	7.7449	5.2376	3.8102	2.9515
Average NCI	1.1251	0.7412	0.5687	0.5422

IGM-CKF is better, exhibiting a decrease of approximately 22.5% in the RMSE. The average NCI provides an evaluation of the relative estimation error and also of the self-assessment provided by each filter in the form of the covariance matrix of the estimated error. The average NCI values of the GM-UKF, GM-CKF, and IGM-CKF are listed in Table 1, and they indicate that the filters are credible.

If it is assumed that the dimension of state is N , the dimension of observation is M , and the number of Gaussian components is IJL , the overall naïve computational complexity of the filter due to the time and measurement updates is $O(N^3 + \underline{M}^3 + N^2M + NM^2)$. The complexity to obtain the matrix J^{PP} , as defined in (23), is $O((IJL)^2N^3)$, and the complexity of J^{PP} can be obtained using the $O(IJLN^3 + (IJL)^2m)$, where m is the number of cubature points. The additional computational cost for the optimizing the weights can be solved in polynomial time using $(S(IJL)^5)$, where S is the length of the weight matrix.

6. Experimental Verification of the Navigation Algorithm Based on IGM-CKF

6.1. System Model. The method uses simple 3-degree-of-freedom (DOF) constant velocity kinematics to predict the evolution of the state from time:

$$\begin{bmatrix} x \\ y \\ \psi \\ u \\ v \\ r \end{bmatrix}_{(k)} = \begin{bmatrix} x + uT \cos \psi - vT \sin \psi \\ y + uT \sin \psi + vT \cos \psi \\ \psi + rT \\ u \\ v \\ r \end{bmatrix}_{(k-1)} + w_{k-1}, \quad (44)$$

where $[x, y, \psi]$ denotes the position and heading of the UUV in global coordinates; $[u, v, r]$ represents the linear and angular velocities of the UUV in vessel coordinates; u and v , respectively, denote the velocities of the UUV in the directions of surge and sway; and r is the yaw angle velocity.

The input includes the location (x, y) , the heading ψ , and the velocities u and v . w_{k-1} is the process noise with covariance Q .

6.2. Observation Model. The UUV uses a Doppler velocimeter (DVL) and OCTANS [31] to record direct measurements of the velocity and heading of the vehicle, respectively. Thus, the observation model can be defined as follows:

$$z_k = Hx_k + v_k, \quad (45)$$

where z is the observation vector with 3 components, including the heading ψ and the line velocities u and v in the

direction of surge, sway, and heave, respectively; H varies with changes in the measurement; and v_k is the observation noise. The observation data of the UUV have a non-Gaussian density because of the underwater environment and the characteristic of the sensors. The observation noise could be approximated as glint noise, which is composed of the Gaussian noise and Laplace noise. For simplicity, in this study, we modeled the observation noise as 2 Gaussian components with different covariances.

6.3. Simulation and Analysis. The data used in the test conducted in this work were obtained from a UUV lake trial completed in March 2016 at $119^\circ E$, $29^\circ N$ in a region with a size of $350 \text{ m} \times 350 \text{ m}$ region and a trial performed in September 2018 at $127^\circ E$, $45^\circ N$ in a region with a size of approximately $600 \text{ m} \times 600 \text{ m}$. The position, velocity, and heading data of the UUV were measured using a GPS, DVL, and OCTANS, respectively. The longitude and latitude data were transformed into geodetic coordinates. A total of 1,450 and 39,000 data points were used. The experimental results pertaining to the GM-EKF, GM-UKF, GM-CKF, cubature Kalman particle filter (CKPF), and IGM-CKF were compared.

The system and observation models, respectively, defined in formulas (43) and (44) were employed. The observation noise was modeled using formula (39), where $\varepsilon = 0.25$, $u_1 = u_2 = 0$, $R_1 = R$, and $R_2 = 100R$. The number of particles of the CKPF was 100. The system process noise covariance and observation noise covariance values are listed in Table 3.

The system state was split into 12 Gaussian components in the IGM-CKF algorithm, and the initial weight values of all the components were equal. The process noise was selected to have a single component, and the observation noise was split into 2 Gaussian components. Under the given conditions, the number of state Gaussian components increased from 12 to 24 while finishing one recursion and the redundant components were merged so that the total number of components remained constant. Figure 3 shows the results of the trajectory estimations from the different algorithms versus the GPS trajectory.

In Figure 3, the starting point is $(0, 0)$, and the end point is $(4.526, 4.764)$.

Figures 4 and 5 show a comparison of the position errors of different algorithms, in the eastern and northern directions, respectively, under 1,450 data points.

Figures 4 and 5 indicate that the GM-EKF exhibits the worst stability, and an error of up to 10 m exists in the east and north directions. The improvement in performance of the GM-UKF compared to that of the GM-EKF can be clearly observed, and the error is limited to 2 m. The position error of the IGM-CKF is smaller than that of the GM-CKF; the error in the east direction is limited to 1.5 m, and the maximum error in the north direction is 1.5 m. The position error of the CKPF is similar to that of the IGM-CKF; the errors in the east and north directions are approximately 1.4 m and 1.3 m, respectively.

To verify the performance of the IGM-CKF-based navigation algorithm under long-endurance conditions, 39,000 data points from the lake trial were selected. The various filters had the same initialization conditions as

TABLE 3: Values of the noise.

Noise covariance	Values
R	$\text{diag}[0.04^2 \ 0.04^2 \ 0.04^2]$
Q	$\text{diag}[9e-4 \ 1e-3 \ 3e-1 \ 9e-2 \ 8e-4 \ 3e-4]$
P_0	$\text{diag}[1 \ 1 \ 1 \ 1 \ 1 \ 1]$

FIGURE 3: Comparison of the trajectory estimations based on 1,450 data points.

FIGURE 4: Comparison of the eastern direction errors.

described previously for the test. The results of the trajectory estimations pertaining to different algorithms vs. the GPS trajectory are shown in Figure 6.

FIGURE 5: Comparison of the northern direction errors.

FIGURE 6: Comparison of the trajectory estimations based on 39,000 data points. In Figure 6, the starting point is (0, 0), and the end point is (383.5, -115.4).

Figures 7 and 8 show a comparison of the position errors in the eastern and northern directions, pertaining to the different algorithms under 39,000 data points.

From Figure 8, it can be observed that the stability of the GM-EKF is the worst in the east direction; the GM-UKF and IGM-CKF exhibit similar accuracies when 30,000 data points are used, whereas the accuracy and stability of the GM-CKF are worse. The GM-CKF and IGM-CKF exhibit similar accuracies under 30,000–39,000 data points, whereas the GM-UKF exhibits poor navigational accuracy. The IGM-CKF exhibits the best navigation accuracy and

FIGURE 7: Comparison of the errors in the eastern direction using 39,000 data points.

FIGURE 8: Comparison of the errors in the northern direction using 39,000 data points.

stability in the east direction. From Figure 8, it can be observed that the GM-UKF exhibits an error accumulation trend with the worst accuracy in the north direction; the largest navigation error is 60 m. The maximum errors of the GM-EKF and GM-CKF are 20 m, and the stability of the GM-EKF is worse. The IGM-CKF exhibits the best navigation accuracy and stability in the north direction, with the error limited to 10 m.

Similar to formulas (42) and (43), we define the position RMSE and NCI of different navigation algorithms using the following expressions:

TABLE 4: RMSE, NCI, and time of difference algorithms.

	1,450 data points			39,000 data points		
	NCI (m)	RMSE (m)	Time (ms)	NCI (m)	RMSE (m)	Time (ms)
GM-EKF	2.32	9.94	1.72	2.61	28.02	5.83
GM-UKF	0.95	1.46	2.42	1.60	18.04	6.58
GM-CKF	0.95	0.88	2.54	0.99	14.6	7.48
IGM-CKF	0.82	0.73	3.66	0.89	9.24	9.02
CKPF	0.88	0.71	20.65	—	—	—

$$\text{RMSE} = \sqrt{\frac{1}{t_{\text{step}}} \sum_{k=1}^{t_{\text{step}}} \left((x_{vx,k} - \hat{x}_{vx,k})^2 + (x_{vy,k} - \hat{x}_{vy,k})^2 \right)}, \quad (46)$$

$$\text{NCI} = \frac{1}{t_{\text{step}}} \sum_{k=1}^{t_{\text{step}}} \left[10 \ln \left(\frac{\left((x_{vx,k} - \hat{x}_{vx,k})^2 + (x_{vy,k} - \hat{x}_{vy,k})^2 \right)}{\left(\text{sqrt}(P_{vx,k}^2 + P_{vy,k}^2) \right)} \right) - 10 \ln \left(\frac{\left((x_{vx,k} - \hat{x}_{vx,k})^2 + (x_{vy,k} - \hat{x}_{vy,k})^2 \right)}{E} \right) \right], \quad (47)$$

where t_{step} is the run time steps; $(x_{vx,k}, x_{vy,k})$ and $(\hat{x}_{vx,k}, \hat{x}_{vy,k})$ are, respectively, the true and estimated positions of the UUV at time k ; and $P_{vx,k}^2$ and $P_{vy,k}^2$, respectively, denote the diagonal elements in the covariance matrix P of the position information x and y .

Table 4 lists the values of the RMSE, NCI, and computation cost for one iteration of the different tested algorithms. The values indicate that the RMSE and run times of the various algorithms increase with increase in the amount of test data. The run time cost for the GM-EKF is the lowest, although it exhibits the worst navigation accuracy. The run time of the GM-UKF is higher because of the calculation of the sigma points, but the navigation accuracy is better than that of GM-EKF. The time costs for the GM-CKF and IGM-CKF are considerably greater because they both need to estimate 12 cubature points. The navigation accuracy in these cases with 39,000 data points processed is considerably better, with an increase of approximately 36.8%. The navigation accuracy of the CKPF is similar to that of the IGM-CKF; however, the run time of the CKPF is the largest among those of all the algorithms. The NCI values of the IGM-CKF and GM-CKF are less than 1, which suggests that the filters have a high credibility. The NCI of the GM-EKF is significantly larger than 1, which suggests that the credibility of the filter is poor, likely due to the high nonlinearity and non-Gaussian distribution. In addition, it was determined that the IGM-CKF algorithm can ensure a high navigation accuracy in long-endurance navigation of the UUV. Although the run time of IGM-CKF is the longest, it can satisfy the requirements for the real-time navigation of UUVs.

7. Conclusions

This paper proposes an alternative efficient nonlinear filtering algorithm, which is a Gaussian mixture cubature

Kalman filter involving weight updation of the Gaussian components. The proposed algorithm approximates the non-Gaussian noise by splitting the system state, process noise, and observation noise into several Gaussian components, and it updates the weight of the Gaussian components in the time update stage.

In this paper, an IGM-CKF algorithm was proposed to adaptively update the Gaussian component weights in the time update phase by using the Chapman-Kolmogorov equation, which is used to solve the problem that the Gaussian component weights in the conventional Gaussian mixture filter remain unchanged in the time update phase when the system is highly nonlinear. The feasibility of the algorithm was proved using a set of simulation trials. The navigation algorithm was compared with conventional navigation algorithms and combined with a motion model and observation model based on lake trial data. The comparison indicated that the IGM-CKF algorithm represents a novel approach for UUV navigation and location. Future work on this topic will include examining the filtering performance of the IGM-CKF with highly nonlinear noise and the calculation of the optimal number of split components.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported in part by the National Natural Science Foundation of China (nos. 61633008 and 51609046) and the Research Fund from Science and Technology on Underwater Vehicle Technology (grant no. 614221502061701).

References

- [1] Y. Zhang, Y. Huang, and L. Zhao, "A general framework solution to Gaussian filter with multiple-step randomly-delayed measurements," *Acta Automatica Sinica*, vol. 41, no. 1, pp. 122–125, 2015.
- [2] Y. Bar Shalom and X. R. Li, *Estimation with Applications to Tracking and Navigation*, pp. 381–395, Wiley, New York, NY, USA, 2001.
- [3] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and

- covariances in filters and estimators," *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 472–482, 2004.
- [4] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.
 - [5] B. O. L. Jagan, S. K. Rao, A. Jawahar et al., "Unscented Kalman filter with application to bearings-only passive target tracking," *Indian Journal of Science & Technology*, vol. 9, no. 19, pp. 63–67, 2016.
 - [6] G. Wang, Y. Zhang, and X. Wang, "Iterated maximum correntropy unscented Kalman filters for non-Gaussian systems," *Signal Processing*, vol. 163, pp. 87–94, 2019.
 - [7] X. Zhang, X. Ma, Z. Liu et al., "A target tracking algorithm based on improved cubature Kalman filters," *Journal of Chengdu University of Information Technology*, vol. 30, no. 6, pp. 559–562, 2015.
 - [8] L. Ning, R. Zhu, and Y. Zhang, "A strong tracking square root CKF algorithm based on multiple fading factors for target tracking," in *Proceedings of the 2014 Seventh International Joint Conference on Computational Sciences and Optimization (CSO)*, July 2014.
 - [9] B. Jia, M. Xin, and Y. Cheng, "High-degree cubature Kalman filter," *Automatica*, vol. 49, no. 2, pp. 510–518, 2013.
 - [10] B. Jia, M. Xin, and Y. Cheng, "Sparse-grid quadrature nonlinear filtering," *Automatica*, vol. 48, no. 2, pp. 327–341, 2012.
 - [11] S. Jia, D. Zeng, T. Xu, H. Zhang, and X. Li, "Improved target tracking based on spatio-temporal learning," in *Proceedings of the IEEE International Conference on Information & Automation*, August 2016.
 - [12] H. Liu and W. Wu, "Interacting multiple model (IMM) fifth-degree spherical simplex-radial cubature Kalman filter for maneuvering target tracking," *Sensors*, vol. 17, no. 6, p. 1374, 2017.
 - [13] J. Dunik, O. Straka, and M. Simandl, "Stochastic integration filter," *IEEE Transactions on Automatic Control*, vol. 58, no. 6, pp. 1561–1566, 2013.
 - [14] J. Georgy and A. Noureldin, "Unconstrained underwater multi-target tracking in passive sonar systems using two-stage PF-based technique," *International Journal of Systems Science*, vol. 45, no. 1–3, pp. 439–455, 2014.
 - [15] X. Fu and Y. Jia, "An improvement on resampling algorithm of particle filters," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5414–5420, 2010.
 - [16] Z. Ma, T. Pei, and H. Zhang, "An improved target tracking algorithm combined with particle filter," *Journal of Liaoning Technical University*, vol. 35, 2016.
 - [17] N. I. Chun-Guang, "A new modified unscented particle filter," *Ship Science & Technology*, vol. 35, no. 1, pp. 35–37, 2013.
 - [18] B. Di, "Adaptive double-resampling particle filter algorithm for target tracking," *Computer Science*, vol. 40, no. 3, pp. 248–250, 2016.
 - [19] R. Zhan and J. Zhang, *Nonlinear Filtering Theory with Target Tracking Application*, pp. 90–96, National Defense Industry Press, Beijing, China, 2013.
 - [20] G. Wang, X. Wang, and Y. Zhang, "Variational Bayesian IMM-filter for JMSs with unknown noise covariances," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 2, pp. 1952–1961, 2019.
 - [21] P. H. Leong, S. Arulampalam, T. A. Lahawewa, and T. D. Abhayapala, "A Gaussian-sum based cubature Kalman filter for bearings-only tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 2, pp. 1161–1176, 2013.
 - [22] Y. Kong and X. Feng, "Passive target tracking algorithm based on improved Gaussian mixture particle filter," *Modern Radar*, vol. 34, no. 7, pp. 44–50, 2012.
 - [23] G. Terejanu, P. Singla, T. Singh, and P. D. Scott, "Uncertainty propagation for nonlinear dynamic systems using Gaussian mixture models," *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 6, pp. 1623–1633, 2008.
 - [24] G. Terejanu, "An adaptive split-merge scheme for uncertainty propagation using Gaussian mixture models," in *Proceedings of the 49th AIAA Aerospace Sciences meeting including the New Horizons Forum and Aerospace Exposition*, Orlando, FL, USA, January 2011.
 - [25] G. Terejanu, P. Singla, Y. Singh, and P. Scott, "A novel Gaussian sum filter method for accurate solution to the nonlinear filtering problem," in *Proceedings of the International Conference on Information Fusion*, pp. 1–8, Cologne, Germany, 2008.
 - [26] J. L. Williams and P. S. Maybeck, "Cost-function-based Gaussian mixture reduction for target tracking," in *Proceedings of the International Conference on Information Fusion*, Queensland, Australia, 2003.
 - [27] K. Xiong, H. Y. Zhang, and C. W. Chan, "Performance evaluation of UKF-based nonlinear filtering," *Automatica*, vol. 42, no. 2, pp. 261–270, 2006.
 - [28] Y. Zhao, "Performance evaluation of cubature Kalman filter in a GPS/IMU tightly-coupled navigation system," *Signal Processing*, vol. 119, pp. 67–79, 2016.
 - [29] X. R. Li and Z. Zhao, "Measuring estimators credibility: noncredibility index," in *Proceedings of the International Conference on Information Fusion*, Florence, Italy, 2006.
 - [30] R. P. Blasch, A. Rice, and C. Yang, "Nonlinear track evaluation using absolute and relative metrics," in *Proceedings of the International Society for Optical Engineering*, pp. 360–365, Orlando, FL, USA, 2006.
 - [31] T. Huilin, "Application of SINS/DVL/UTP/VP integrated navigation system to UUV underwater docking," *Applied Science and Technology*, vol. 43, no. 3, 2016.