Research Article

Application of Improved Simplex Quadrature Cubature Kalman Filter in Nonlinear Dynamic System

Ting Cao [iD],[1] **Huo-tao Gao** [iD],[1] **Chun-feng Sun,**[2] **Yun Ling** [iD],[1] **and Guo-bao Ru** [iD][1]

[1]*Electromagnetic Wave and Modern Radar Lab., School of Electronic Information, Wuhan University, Wuhan 430079, China*
[2]*School of Physics and Electronic Information Engineering, Hubei Engineering University, Xiaogan 432000, China*

Correspondence should be addressed to Huo-tao Gao; gaoght863@163.com

A novel spherical simplex Gauss–Laguerre quadrature cubature Kalman filter is proposed to improve the estimation accuracy of nonlinear dynamic system. The nonlinear Gaussian weighted integral has been approximately evaluated using the spherical simplex rule and the arbitrary order Gauss–Laguerre quadrature rule. Thus, a spherical simplex Gauss–Laguerre cubature quadrature rule is developed, from which the general computing method of the simplex cubature quadrature points and the corresponding weights are obtained. Then, under the nonlinear Kalman filtering framework, the spherical simplex Gauss–Laguerre quadrature cubature Kalman filter is derived. A high-dimensional nonlinear state estimation problem and a target tracking problem are utilized to demonstrate the effectiveness of the proposed spherical simplex Gauss–Laguerre cubature quadrature rule to improve the performance.

1. Introduction

Nonlinear filtering and estimation based on Bayes framework have been widely applied in many fields, such as target tracking [1–5], navigation [5, 6] and positioning [7, 8], power system [9–11], and pattern recognition [12]. Recent years have seen a surge of interest in solving the nonlinear estimation problems. As is well known, the optimal solution to nonlinear systems is impractical. The focus has been on developing suboptimal algorithms, and many approximate nonlinear filters are generally proposed. The approximate nonlinear filters can be divided into two classes: global approach and local approach. In global approximations, one intuitive scheme is to randomly select a set of weighted particles to approximate the propagation of possible density functions (PDFs), giving birth to the particle filter (PF) [13] and grid-mass filter [14]. Although filters in this approach have high precision, the computational burden is its primary obstacle to more widespread use. Another scheme is the global linearization [15] method, in which the differential inclusion theory represents the nonlinear system via a linear differential inclusion (LDI) model. This kind of method can

effectively solve the problem of local linearization error in the local linearization method. It can also handle different types of noise and model uncertainty with strong robustness. Despite the superior performance of global linearization, the problem of filter design conservatism is one of the bottlenecks. To reduce the conversion, Yu and Li [16] proposed a nested tensor product model transformation (NTPMT). The NTPMT is able to obtain multilevel nested TP models for the given systems and reduce the dimensionality of the vertex tensor such that the number of vertex systems contained in the resultant TP model is reduced effectively. The proposed transformation has indicated an attractive research direction in global linearization.

The local approximations include the analytical approximations and sampling-based approaches. The most widely used nonlinear filter is the extended Kalman filter (EKF) [17], which is the typical analytical approximate nonlinear filter. The state estimation is approximated by a Gaussian random variable (GRV), which is then propagated analytically through the first-order linearization of the nonlinear process and measurement system. However, the EKF has slow convergence and low estimated accuracy and

may suffer from large errors when the system has high nonlinearity. The unscented Kalman filter (UKF) [18], the Gauss–Hermite quadrature filter (GHQF) [19], and the cubature Kalman filter (CKF) [20] fall under the latter category. The UKF addresses the flaws of the EKF by using a set of deterministic points to calculate the mean and co-variance of nonlinear functions. Based on the Gauss–Hermite quadrature (GHQ) rule, the GHQF is more stable and accurate than UKF. Unfortunately, the experi-mental growth with the dimension of the number of quadrature points and computational complexity has pro-hibited its widespread use.

Recently, a cubature Kalman filter [20] has been proposed and widely used in many applications for its simplicity and effectiveness. The heart of the CKF is a spherical-radial cubature rule, which makes it possible to numerically compute multi-variate moment integrals met in the nonlinear Bayesian framework. Since its inception in 2009, CKF has been an active research area for many researchers all over the world resulting in various variants of the basic algorithm with improved per-formance. A new class of CKFs with arbitrary degrees of ac-curacy in computing the spherical and radial integrals was put forward by Jia [21]. Based on a spherical simplex-radial rule, Wang et al. [22] proposed the spherical simplex-radial cubature Kalman filter (SSRCKF). In SSRCKF, the transformation group of the regular simplex and the moment matching method are used to calculate the spherical and the radial integrals, re-spectively. Simulation results show that the SSRCKF can obtain higher accuracy and better efficiency than the traditional CKFs for high-dimension systems.

However, the radial integral in both CKFs and SSRCKFs is solved by the moment matching method [22], which is am-biguous and does not provide the best possible solution available [23]. To tackle this problem, Bhaumik and Swati proposed the cubature quadrature Kalman filter (CQKF) [24]. In CQKF, the spherical integral is calculated the same as the CKFs, but the radial integral is solved with arbitrary order Gauss–Laguerre quadrature rule. Thanks to the arbitrary order Gauss–Laguerre quadrature rule, the filtering accuracy has been further improved. It is also indicated that the CKF is a special case of CQKF with the first-order Gaussian–Laguerre quad-rature in radial integral. To further improve the performance of the CQKF, a higher degree cubature quadrature Kalman filter (HDCQKF) [23] is proposed.

The contribution of this paper is to propose a new class of CKFs with the spherical simplex rule to approximate the spherical integral and arbitrary order Gauss–Laguerre quadrature rule to solve the radial integral. The proposed filter is termed as spherical simplex Gauss–Laguerre quadrature cubature Kalman filter (SSGQKF). The SSGQKF would be a generalized form of SSRCKF, and under single Gauss–Laguerre quadrature point evaluation, it coincides with the third-degree SSRCKF. The accuracy of the SSGQKF depends on the number of the simplex cubature points. The higher the number of simplex quadrature points, the better the accuracy. Moreover, the computational burden of the SSGQKF does not increase exponentially with the dimen-sion, and thus it does not have the curse of dimensionality problem. In addition, it is derivative free. The proposed

nonlinear filter can provide more accurate results with less computational costs than HDCQKF in high-dimensional nonlinear state estimation systems. Two examples have been provided to demonstrate the superiority of the SSGQKF.

The rest of this paper is organized as follows. Section 2 reviews the cubature quadrature Kalman filter in the Bayesian framework of filtering and estimation. In Section 3, we present the proposed SSGQKF method. To verify the superiority of the SSGQKF method, Section 4 shows the compared performance for the conventional nonlinear fil-tering and the proposed algorithm. Finally, several con-cluding remarks are given in Section 5 to end this paper.

2. Cubature Quadrature Kalman Filter

2.1. Nonlinear System Model.
Consider a class of nonlinear discrete estimation systems described by

$$\begin{aligned} \mathbf{x}_k &= \mathbf{f}\left(\mathbf{x}_{k-1}\right) + \mathbf{v}_{k-1}, \\ \mathbf{y}_k &= \mathbf{h}\left(\mathbf{x}_k\right) + \mathbf{n}_k, \end{aligned} \tag{1}$$

where $\mathbf{x}_k \in \mathbb{R}^{n_x}$ is the state vector of the dynamic system at discrete time k; $\mathbf{y}_k \in \mathbb{R}^m$ is the measurement vector; and $\mathbf{v}_{k-1}$ and $\mathbf{n}_k$ are independent process and measurement Gaussian white noise with zero means and covariances $\mathbf{Q}_{k-1}$ and $\mathbf{R}_k$, respectively.

Under the Gaussian domain, Bayesian filters can be generalized by computing the following integral:

$$I(f) = \frac{1}{\sqrt{|\Sigma|(2\pi)^n}} \int_{R^n} f(x) e^{-(1/2)(x-\mu)^T \Sigma^{-1}(x-\mu)} \mathrm{d}x. \tag{2}$$

Let $x = Crz + \mu$, where μ is the mean and $\Sigma = CC^T$. Then, the integral $I(f)$ can be written as

$$I(f) = \frac{1}{\sqrt{(2\pi)^n}} \int_{r=0}^{\infty} \int_{U_n} [f(Crz + \mu)\mathrm{d}\sigma(z)] r^{n-1} e^{-(r^2/2)} \mathrm{d}r. \tag{3}$$

If we assume zero mean and unity covariance, equation (3) becomes

$$I(f) = \frac{1}{\sqrt{(2\pi)^n}} \int_{r=0}^{\infty} \int_{U_n} [f(rz)\mathrm{d}\sigma(z)] r^{n-1} e^{-(r^2/2)} \mathrm{d}r, \tag{4}$$

where $U_n = \{z \in \mathbb{R}^n \mid z^T z = 1\}$ is the spherical surface and $\sigma(\cdot)$ is the spherical surface measure or the area element on U_n. To compute the integral $I(f)$, we decompose it into a surface integral $S(r)$ and a line integral. We may thus write the line integral

$$I = \int_{r=0}^{\infty} S(r) r^{n-1} e^{-(r^2/2)} \mathrm{d}r, \tag{5}$$

in which $S(r)$ is defined by the spherical integral with the unit weighting function $w(z) = 1$:

$$S(r) = \int_{U_n} f(rz)\mathrm{d}\sigma(z). \tag{6}$$

In this paper, the spherical and the line integrals are numerically computed by the spherical simplex rule adopted

from [22] and the arbitrary order Gauss–Laguerre quadrature rule, respectively.

2.2. Spherical Simplex Rule.

The spherical integral of the form $\int_{U_n}[f(rz)\mathrm{d}\sigma(z)]$ can be efficiently evaluated by the transformation group of the regular n–simplex with vertices $\mathbf{a}_j = [a_{j,1}, a_{j,2}, \cdots, a_{j,n}]^T$, $j = 1, 2, \ldots, n+1$ [25, 26], i.e.,

$$
a_{j,i} \equiv \begin{cases} -\sqrt{\dfrac{n+1}{n(n-i+2)(n-i+1)}}, & i < j, \\[4mm] \sqrt{\dfrac{(n+1)(n-j+1)}{n(n-j+2)}}, & i = j, \\[4mm] 0, & i > j. \end{cases} \tag{7}
$$

The set of midpoints of the vertices $\mathbf{a}_j$ projected onto the spherical surface yields the following point set [25]:

$$
\{\mathbf{b}_j\} \equiv \left\{ \sqrt{\frac{n}{2(n-1)}}(\mathbf{a}_i + \mathbf{a}_l) : i < l, \quad l = 1, 2, \ldots, n+1 \right\}. \tag{8}
$$

Thus, the third-degree spherical simplex rule that contains $2n+2$ integral points is as follows:

$$
S_3(r) = \frac{A_n}{2(n+1)} \sum_{j=1}^{n+1} \left[f(r\mathbf{a}_j) + f(-r\mathbf{a}_j) \right], \tag{9}
$$

where $A_n = 2\sqrt{\pi^n}/\Gamma(n/2)$ is the surface area of the unit sphere and $\Gamma(n) = \int_0^\infty x^{n-1}e^{-x}\mathrm{d}x$. Similarly, the fifth-degree spherical simplex rule with $n^2 + 3n + 2$ quadrature points is obtained as follows [25, 26]:

$$
S_5(r) = A_n \left\{ \frac{(7-n)n}{2(n+1)^2(n+2)} \sum_{j=1}^{n+1} \left(f(r\mathbf{a}_j) + f(-r\mathbf{a}_j) \right) + \frac{2(n-1)^2}{n(n+1)^2(n+2)} \sum_{j=1}^{n(n+1)/2} \left[f(r\mathbf{b}_j) + f(-r\mathbf{b}_j) \right] \right\}. \tag{10}
$$

Therefore, the spherical integral can be approximately by N_s point spherical simplex rule as

$$
\int_{U_n} [f(rz)\mathrm{d}\sigma(z)] \approx \sum_{i=1}^{N_s} \omega_{s,i} f(rz_i). \tag{11}
$$

2.3. Gauss–Laguerre Quadrature Rule.

It is known that any integral in the form of

$$
\int_{\lambda=0}^\infty f(\lambda)\lambda^\alpha e^{-\lambda}\mathrm{d}\lambda, \tag{12}
$$

can be approximately evaluated using quadrature points and weights associated with them. The quadrature points can be derived from the roots of the $n\prime$ order of Chebyshev–Laguerre polynomial equation [27, 28]:

$$
L_{n'}^\alpha(\lambda) = (-1)^{n'}\lambda^{-\alpha}e^\lambda \frac{\mathrm{d}^{n'}}{\mathrm{d}\lambda^{n'}}\lambda^{\alpha+n'}e^{-\lambda} = 0. \tag{13}
$$

The weights of the quadrature points $\lambda_{n\prime}$ can be determined as

$$
A_{i'} = \frac{n\prime! \Gamma(\alpha+n'+1)}{\lambda_{i'}\left[L_n^\alpha(\lambda_{i'})\right]^2}. \tag{14}
$$

So, integral (12) can be written approximately using the Gauss–Laguerre quadrature rule as

$$
\int_{\lambda=0}^\infty f(\lambda)\lambda^\alpha e^{-\lambda}\mathrm{d}\lambda \approx \sum_{i'=1}^{n'} A_{i'} f(\lambda_{i'}). \tag{15}
$$

3. Spherical Simplex Gauss–Laguerre Quadrature Evaluation of Integrals

3.1. Spherical Simplex Gauss–Laguerre Quadrature Rule.

From (12) and (16), the spherical simplex Gauss–Laguerre rule for $N(\mathbf{x}; 0, \mathbf{I})$ can be approximated by

$$
I(f) = \frac{1}{\sqrt{(2\pi)^n}} \int_{r=0}^\infty \left[\sum_{i=1}^{N_s} \omega_{s,i} f(rz_i) \right] r^{n-1} e^{-(r^2/2)}\mathrm{d}r, \tag{16}
$$

$$
= \frac{1}{2\sqrt{\pi^n}} \int_{t=0}^\infty \left[\sum_{i=1}^{N_s} \omega_{s,i} f(\sqrt{2t}\, z_i) \right] t^{(n/2)-1} e^{-t}\mathrm{d}t, \tag{17}
$$

$$
= \frac{1}{2\sqrt{\pi^n}} \sum_{i'=1}^{n'} \omega_i' \left[\sum_{i=1}^{N_s} \omega_{s,i} f(\sqrt{2t}\, z_i) \right]. \tag{18}
$$

Note that to transform the integral $I(f)$ into the form of equation (12), we make an assumption that $t = r^2/2$. Let the numerical method for $\int_{R^n} f(x)N(\mathbf{x}; 0, \mathbf{I})\mathrm{d}x$ be represented by $Q(f)$. Substituting (9) and (15) into (18) yields the third-degree spherical simplex Gauss–Laguerre quadrature rule as

$$
Q(f) = \frac{1}{2\sqrt{\pi^n}} \sum_{i'=1}^{n'} A_{i''} \left(\frac{A_n}{2(n+1)} \sum_{j=1}^{n+1} \left[f(r\mathbf{a}_j) + f(-r\mathbf{a}_j) \right] \right). \tag{19}
$$

Similarly, substituting (10) and (15) into (18), we have the fifth-degree spherical simplex Gauss–Laguerre quadrature rule as

$$Q(f) = \frac{1}{2\sqrt{\pi^n}} \sum_{i_I=1}^{n'} A_{i_I} \left(A_n \left(\frac{(7-n)n}{2(n+1)^2(n+2)} \sum_{j=1}^{n+1} \left(f(r\mathbf{a}_j) + f(-r\mathbf{a}_j) \right) + \frac{2(n-1)^2}{n(n+1)^2(n+2)} \sum_{j=1}^{n(n+1)/2} \left[f(r\mathbf{b}_j) + f(-r\mathbf{b}_j) \right] \right) \right).$$

(20)

The Gaussian probability weighted integral $\int_{R^n} f(\mathbf{x}) N(\mathbf{x}; \overline{\mathbf{x}}, \mathbf{P_x}) d\mathbf{x}$ can be performed by the linear transformation of $\sqrt{\mathbf{P}}\mathbf{x}_i + \widehat{\mathbf{x}}$, where $\sqrt{\mathbf{P}}$ is the square root of the matrix $\mathbf{P}$ and $\mathbf{x}_i$ is the quadrature points from (19) or (20).

For a n dimension of state-space problem solved with third-order spherical simplex rule and $n\prime$ order Gauss–Laguerre quadrature points, total $2(n+1)n\prime$ points and associated weights need to be calculated. The fifth-degree spherical simplex Gauss–Laguerre quadrature rule has $(n^2+3n+2)n\prime$ points and shares $2(n+1)n\prime$ points with the third-degree spherical simplex Gauss–Laguerre quadrature rule. For simplicity, these simplex cubature quadrature points are termed as SCQ points in this paper.

3.2. Calculation of Simplex Cubature Quadrature Points and Weights.
The steps for calculating the simplex cubature quadrature (SCQ) points and associated weights are as follows:

(i) Determine the system dimension n, the orders of spherical simplex cubature rule (i.e., m), and Chebyshev–Laguerre polynomial ($n\prime$). Find out the set of sample points for a given specific degree of spherical simplex cubature rule.

(ii) For the third-order $(2m+1=3)$ spherical simplex rule, the sample points $\xi_i = [\mathbf{a}, -\mathbf{a}]$.

(iii) For the fifth-order $(2m+1=5)$ spherical simplex rule, the sample points $\xi_i = [\mathbf{a}, -\mathbf{a}, \mathbf{b}, -\mathbf{b}]$.

(iv) Solve the $n\prime$ order Chebyshev–Laguerre polynomial (13) for $\alpha = (n/2-1)$ to derive the quadrature points λ_{i_I}.

(v) Find the SCQ points as $\xi_j = \sqrt{2\lambda_{i_I}}\xi_i$.

(vi) The weights associated with SCQ points are

$$W_j = \frac{A_{i_I}\omega_p}{2\sqrt{\pi^n}} = \frac{n\prime!\,\Gamma(\alpha+n\prime+1)}{\lambda_{i_I}\left[\dot{L}_n^\alpha\left(\lambda_{i_I}\right)\right]^2} \times \frac{\omega_p}{2\sqrt{\pi^n}}.$$

(21)

(i) For the third-order spherical simplex rule $(j = 1, 2, \ldots, 2n\prime(n+1))$:

$$\omega_p = \frac{A_n}{2(n+1)}, \quad p = 1, 2, \ldots, 2(n+1).$$

(22)

(ii) For the fifth-order spherical simplex rule $(j = 1, 2, \ldots, (n^2+3n+2)n\prime)$:

$$\omega_p = \begin{cases} \dfrac{A_n(7-n)n}{2(n+1)^2(n+2)}, & p = 1, 2, \ldots, 2(n+1), \\[4mm] \dfrac{2A_n(n-1)^2}{n(n+1)^2(n+2)}, & p = 2n+3, \ldots, n^2+3n+2. \end{cases}$$

(23)

The number of support SCQ points required for different ordered spherical simplex cubature rule is shown in Table 1. The SCQ points and their associated weight of a second-order system for the third- and fifth-degree spherical simplex cubature and up to the third-order Gauss–Laguerre quadrature approximation are shown in Table 2. As the dimension of the state-space models increases, the computational burden also increases. We only discuss the third-degree and the fifth-degree spherical simplex rule here.

3.3. The Spherical Simplex Gauss–Laguerre Quadrature Cubature Kalman Filter (SSGQKF).
Applying the spherical simplex Gauss–Laguerre quadrature rule described in Section 3.2 in the Bayesian filter framework yields the new spherical simplex Gauss–Laguerre quadrature cubature Kalman filter (SSGQKF).

The algorithm of the proposed SSGQKF could be summarised as follows.

Step 1. Initialization

(i) Initialize the filter with $\widehat{\mathbf{x}}_{0|0}$ and $\mathbf{P}_{0|0}$.

(ii) Calculate the SCQ points ξ_j and their corresponding weights W_j as described in Section 3.2.

Step 2. Prediction step

(i) Factorise covariance matrix using Cholesky decomposition:

$$\mathbf{P}_{k|k} = \mathbf{S}_{k|k}\mathbf{S}_{k|k}^T.$$

(24)

(ii) Evaluate the SCQ points:

$$\chi_{j,k|k} = \mathbf{S}_{k|k}\xi_j + \widehat{\mathbf{x}}_{k|k}.$$

(25)

(iii) Update the SCQ points:

$$\chi_{j,k+1|k} = \mathbf{f}(\chi_{j,k|k}).$$

(26)

(iv) Estimate the predicted state and the predicted error covariance:

$$\widehat{\mathbf{x}}_{k+1|k} = \sum_{j=1}^{t} \chi_{j,k+1|k} \mathbf{W}_j,$$

$$\mathbf{P}_{k+1|k} = \sum_{j=1}^{t} \mathbf{W}_j \left[\chi_{j,k+1|k} - \widehat{\mathbf{x}}_{k+1|k} \right] \left[\chi_{j,k+1|k} - \widehat{\mathbf{x}}_{k+1|k} \right]^T + \mathbf{Q}_k, \tag{27}$$

where t is the total number of SCQ points.

Step 3. Measurement step

(i) Factorise the predicted error covariance matrix $\mathbf{P}_{k+1|k}$ using the Cholesky decomposition:

$$\mathbf{P}_{k+1|k} = \mathbf{S}_{k+1|k} \mathbf{S}_{k+1|k}^T. \tag{28}$$

(ii) Evaluate the SCQ points:

$$\chi_{j,k+1|k} = \mathbf{S}_{k+1|k} \xi_j + \widehat{\mathbf{x}}_{k+1|k}. \tag{29}$$

(iii) Find the propagated measurements at each SCQ points:

$$\mathbf{Y}_{j,k+1|k} = \mathbf{h}\left(\chi_{j,k+1|k} \right). \tag{30}$$

(iv) Estimate the predicted measurement:

$$\widehat{\mathbf{y}}_{k+1} = \sum_{j=1}^{t} \mathbf{Y}_{j,k+1|k} \mathbf{W}_j. \tag{31}$$

(v) Calculate the innovation covariance matrix and the cross-covariance matrix:

$$\mathbf{P}_{yy} = \sum_{j=1}^{t} \mathbf{W}_j \left[\mathbf{Y}_{j,k+1|k} - \widehat{\mathbf{y}}_{k+1} \right] \left[\mathbf{Y}_{j,k+1|k} - \widehat{\mathbf{y}}_{k+1} \right]^T + \mathbf{R}_k, \mathbf{P}_{xy} = \sum_{j=1}^{t} \mathbf{W}_j \left[\chi_{j,k+1|k} - \widehat{\mathbf{x}}_{k+1} \right] \left[\mathbf{Y}_{j,k+1|k} - \widehat{\mathbf{y}}_{k+1} \right]^T. \tag{32}$$

(vi) Compute the Kalman gain:

$$K_{k+1} = P_{xy} P_{yy}^{-1}. \tag{33}$$

(vii) Compute the filtering estimation:

$$\widehat{\mathbf{x}}_{k+1|k+1} = \widehat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} \left(\mathbf{y}_{k+1} - \widehat{\mathbf{y}}_{k+1} \right). \tag{34}$$

(viii) Compute the filtering estimation error covariance matrix:

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{P}_{yy} \mathbf{K}_{k+1}^T. \tag{35}$$

Remarks

(i) The spherical simplex cubature Gauss–Laguerre quadrature rule constructed using the above theorems is fully symmetric.

(ii) For a specific system dimension n, there are two tuning parameters, namely, m and $nı$, which could be tuned to achieve satisfied accuracy on affordable computational load.

(iii) Compared to EKF, the SSGQKF is derivative free. It is easy to implement as neither a Jacobian nor Hessian matrix needs to be computed. It is also computationally more efficient than the PF and GHF because instead of generating large number of particles or quadrature points in the state space, few SCQ points are required to be generated, updated, and predicted.

(iv) For a specific order of Chebyshev–Laguerre polynomial $nı$, the number of SCQ points of the fifth-degree SSGQKF is less than that of the existing fifth-degree HDCQKF $(2n^2nı)$ when $n > 4$. Therefore, from the perspective of computation, the proposed fifth-degree SSGQKF is much less complex than HDCQKF, making it easier to implement in high-dimensional nonlinear systems. That is to say, the SSGQKF will be a potential and promising method in practical engineering implementation.

(v) When $n > 4$, the fifth-degree spherical-radial rule has some negative weights. Negative weights may lead to divergence of filters [22]. Despite some of the weights from (20) in the fifth-degree spherical simplex cubature rule becoming negative when $n > 7$, they go to 0 when $n \longrightarrow \infty$. Thus, the proposed fifth-degree SSGQKF is more efficient than the existing HDCQKF for high-dimensional systems. Consequently, the SSGQKF can be a very promising alternative to the HDCQKF under the Gaussian assumption, especially in the case of high-dimensional cases.

(vi) For a predefined ordered cubature and Gauss–Laguerre quadrature, the SCQ points and associated weights can be calculated and stored offline. In other words, the computation burden could be further alleviated.

In the following, a few discussions are added for the practical implementation of the proposed method Like the traditional Kalman filter (KF) and its suboptimal extensions (e.g., EKF, UKF, and CKF), the proposed SSGQKF is also developed under the Gaussian assumption and the minimum mean square error (MMSE) criterion. To deal with non-Gaussian noises and for further improvement, the proposed filter can be combined with other techniques such as the extended H_∞ filtering (EH_∞F) [29], Huber's

Table 1: Support points required for different ordered spherical simplex cubature and $n\prime$ order Gauss–Laguerre quadrature rule.

Degree of spherical simplex cubature rule	Number of SCQ points
3	$2(n+1)n\prime$
5	$(n^2 + 3n + 2)n\prime$
7	$(n+1)(n^2 + 8n + 6)n\prime/3$

M-estimation theory [30], maximum correntropy criterion (MCC) [31], and global linearization method to get better and more robust performance. (1) Since the extended H_∞ filter does not make any assumptions about the statistics of the process or measurement noise but does require Jacobian matrices during the state estimation of nonlinear systems, by using the derivative-free property of SSGQKF and the linear propagation property, it is possible to embed the $EH_\infty F$ in the SSGQKF framework. (2) Similarly, a robust scheme integrating Huber's M-estimation theory and the SSGQKF framework is beneficial where Huber's M-en is adopted to reformulate the measurement information of the SSGQKF. (3)The MCC is a statistical metric of the similarity between two random variables using information from higher-order signal statistics, which may perform much better than MMSE in non-Gaussian noise environments, since the MMSE captures only the second-order statistics of the error signal and is sensitive to large outliers. (4) Based on the polytopic linear differential inclusion, the estimation error system for the nonlinear system can be represented by an uncertain polytopic linear model. Then, the rectification equations for the predicted errors are designed. Finally, the state estimates for the nonlinear system are given through updating the predictions of SSGQKF. Generally speaking, the proposed SSGQKF can be applied to the nonlinear non-Gaussian environment to obtain a robust performance, which can be our future work.

Since the proposed method is computationally more efficient than the HDCQKF, it can also be expanded to the distributed fusion estimation problem for multisensor nonlinear systems [32, 33].

The estimation problem sometimes needs to process a huge number of data, costing time and energy waste [34]. The event trigger or data driven method has been successfully applied in the practical system [9–11], which can reduce the sampling so as to reduce the computation with the guaranteed accuracy. For a proper m and $n\prime$ and a given general model in nonlinear system, the SCQ points and corresponding weight can be prepared offline. Then, similar to event-triggered cubature Kalman filter (ETCKF) proposed in [11], the event-trigger schedule can also be embedded into the SSGQKF method. Motivated by the event-trigger heterogeneous nonlinear filter (ET-HNF) [10], which utilizes a master-slave filter at generator node and estimation center and a low computational slave filter to execute the event-trigger strategy, particle flow filters and the proposed SSGQKF3 method can serve as the master and slave filter, respectively.

Generally, particle filter serves as the master due to its effectiveness to nonlinear dynamic and/or measurement filtering problem. However, as explained in [35, 36], particle filter commonly suffers from the curse of dimensionality (COD) owing to particle degeneracy, which is caused by Bayes' rule. Recent advances in particle flow filters provide a promising avenue to avoid weight degeneracy and COD. Instead of using any proposal and resample schedule, the particle flow filters compute Bayes' rule by particle flow, making it much easier to implement and perform better than the PF.

4. Simulations and Discussion

In this section, the effectiveness of the proposed SSGQKF is demonstrated by comparing the SSGQKF with the CKF, the third-degree SSRCKF (SSRCKF3), the CQKF, and the HDCQKF in terms of tracking accuracy.

Two problems are considered. The first problem is a highly nonlinear state estimation issue. The second problem is a tracking model of a vehicle. Both of the problems have significant nonlinearity.

In all the problems, the CQKF (or SSGQKF3) is implemented with the third-degree spherical (simplex) cubature rule and second-order Gauss–Laguerre quadrature rule. The HDCQKF (or SSGQKF5) is implemented with the fifth-degree spherical (simplex) cubature rule and second-order Gauss–Laguerre quadrature rule.

All algorithms were implemented on a personal computer with a 64-bit operating system, 8 GB RAM, and 2.20 GHZ i7 processor. All the computations associated with cubature points were done offline.

4.1. Problem 1. We consider the following nonlinear system for which the state and measurement models are [22, 23, 37, 38]

$$\begin{aligned} \mathbf{x}_{k+1} &= 20\cos(\mathbf{x}_k) + \mathbf{\eta}_k, \\ y_k &= \sqrt{1 + \mathbf{x}_k^T \mathbf{x}_k} + v_k, \end{aligned} \tag{36}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ is an n dimensional state vector at k-th instant of time and $\cos(\mathbf{x}_k) = [\cos(x_{1,k}), \cos(x_{2,k}), \ldots, \cos(x_{n,k})]^T$; y_k is the measurement at time k; and η_k and v_k are independent process and measurement Gaussian noise sequences with zero means and covariances $\mathbf{I}_n$ and 1, respectively.

The true initial state is $\mathbf{x}_0 = 0.1 \times \mathbf{I}_{n\times 1}$, and the filter is initialized with a value of $\widehat{\mathbf{x}}_0$ and $\mathbf{P}_0$, where $\widehat{\mathbf{x}}_{0|0} = 0_{n\times 1}$ and $\mathbf{P}_{0|0} = \mathbf{I}_n$. The states are estimated for 100 steps and the accuracy of the estimators is compared in terms of root mean square error (RMSE). For M number of Monte Carlo runs, the RMSE is defined as

$$\text{RMSE}_k = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(x_{i,k} - \widehat{x}_{i,k}\right)^2}, \quad i = 1, 2, \ldots, M, k = 1, 2, \ldots, N. \tag{37}$$

During the simulation, a ten (i.e., $n = 10$) dimensional system is considered and 500 Monte Carlo runs are used to evaluate the RMSE. Considering the limitation of this article,

TABLE 2: Points and weights for different degree of simplex cubature and higher-order Gauss–Laguerre rules for a second-order system.

$2m+1$	$n\prime$	SCQ points	Weights
3	1	(1.4142, 0), (−0.7071, 1.2247), (−0.7071, −1.2247), (−1.4142, 0), (0.7071, −1.2247), (0.7071, 1.2247)	1/6, 1/6, 1/6, 1/6, 1/6, 1/6
	2	(2.6131, 0), (−1.3066, 2.2630), (−1.3066, −2.2630), (−2.6131, 0), (1.3066, −2.2630), (1.3066, 2.2630), (1.0824, 0), (−0.5412, 0.9374), (−0.5412, −0.9374), (−1.0824, 0), (0.5412, −0.9374), (0.5412, 0.9374)	0.0244, 0.0244, 0.0244, 0.0244, 0.0244, 0.0244, 0.1423, 0.1423, 0.1423, 0.1423, 0.1423, 0.1423
	3	(3.5468, 0), (−1.7734, 3.0716), (−1.7734, −3.0716), (−3.5468, 0), (1.7734, −3.0716), (1.7734, 3.0716), (2.1421, 0), (−1.0710, 1.8551), (−1.0710, −1.8551), (−2.1421, 0), (1.0710, −1.8551), (1.0710, 1.8551), (0.9119, 0), (−0.4559, 0.7897), (−0.4559, −0.7897), (−0.9119, 0), (0.4559, −0.7897), (0.4559, 0.7897)	0.0017, 0.0017, 0.0017, 0.0017, 0.0017, 0.0017, 0.0464, 0.0464, 0.0464, 0.0464, 0.0464, 0.0464, 0.1185, 0.1185, 0.1185, 0.1185, 0.1185, 0.1185
5	1	(1.4142, 0), (−0.7071, 1.2247), (−0.7071, −1.2247), (−1.4142, 0), (0.7071, −1.2247), (0.7071, 1.2247), (0.7071, 1.2247), (0.7071, −1.2247), (−1.4142, 0), (−0.7071, −1.2247), (1.4142, 0)	5/36, 5/36, 5/36, 5/36, 5/36, 5/36, 1/36, 1/36, 1/36, 1/36, 1/36, 1/36
	2	(2.6131, 0), (−1.3066, 2.2630), (−1.3066, −2.2630), (−2.6131, 0), (1.3066, −2.2630), (1.3066, 2.2630), (1.3066, 2.2630), (1.3066, −2.2630), (−2.6131, 0), (−1.3066, −2.2630), (−1.3066, 2.2630), (2.6131, 0), (1.0824, 0), (−0.5412, 0.9374), (−0.5412, −0.9374), (−1.0824, 0), (0.5412, −0.9374), (0.5412, 0.9374), (0.5412, 0.9374), (0.5412, −0.9374), (−1.0824, 0), (−0.5412, −0.9374), (−0.5412, 0.9374), (1.0824, 0)	0.0203, 0.0203, 0.0203, 0.0203, 0.0203, 0.0203, 0.0041, 0.0041, 0.0041, 0.0041, 0.0041, 0.0041, 0.1185, 0.1185, 0.1185, 0.1185, 0.1185, 0.1185, 0.0237, 0.0237, 0.0237, 0.0237, 0.0237, 0.0237
	3	(3.5468, 0), (−1.7734, 3.0716), (−1.7734, −3.0716), (−3.5468, 0), (1.7734, −3.0716), (1.7734, 3.0716), (1.7734, 3.0716), (1.7734, −3.0716), (−3.5468, 0), (−1.7734, −3.0716), (−1.7734, 3.0716), (3.5468, 0), (2.1421, 0), (−1.0710, 1.8551), (−1.0710, −1.8551), (−2.1421, 0), (1.0710, −1.8551), (1.0710, 1.8551), (1.0710, 1.8551), (1.0710, −1.8551), (−2.1421, 0), (−1.0710, −1.8551), (−1.0710, 1.8551), (2.1421, 0), (0.9119, 0), (−0.4559, 0.7897), (−0.4559, 0.7897), (−0.9119, 0), (0.4559, −0.7897), (0.4559, 0.7897), (0.4559, 0.7897), (0.4559, −0.7897), (−0.9119, 0), (−0.4559, −0.7897), (−0.4559, 0.7897), (0.9119, 0)	0.0014, 0.0014, 0.0014, 0.0014, 0.0014, 0.0014, 0.0003, 0.0003, 0.0003, 0.0003, 0.0003, 0.0003, 0.0387, 0.0387, 0.0387, 0.0387, 0.0387, 0.0387, 0.0077, 0.0077, 0.0077, 0.0077, 0.0077, 0.0077, 0.0988, 0.0988, 0.0988, 0.0988, 0.0988, 0.0988, 0.0198, 0.0198, 0.0198, 0.0198, 0.0198, 0.0198

the RSMEs of the first six states using different filters in an interval of 20–70 s are shown in Figures 1–3, respectively.

The average RMSE of the six filters in 100 steps is listed in Table 3. The average Monte Carlo run times of CKF, SSRCKF3, CQKF, HDCQKF, SSGQKF3, and SSGQKF5 for the above problem are 0.1823, 0.2083, 0.3016, 1.5313, 0.3262, and 1.1719 seconds, respectively.

It is clear to see from Figures 1–3 and Table 3 that RMSEs of SSGQKF3 are smaller than those of CKF, SSRCKF3, and CQKF. Similarly, the RMSEs of SSGQKF5 are smaller than those of HDCQKF. The RMSE reflects the accuracy of the algorithm. Besides, we can also see that the proposed SSGQKF3 (or SSGQKF5) is more efficient than the CQKF (or HDCQKF) in terms of computation burden. Thus, the proposed filter has better filtering performance as compared with the CQKFs and existing third-degree SSRCKF.

4.2. Problem 2. In this section, a computer simulation is performed in a target tracking application, which has been used as a benchmark problem to validate the performance of different nonlinear filtering algorithms.

We consider a typical air-traffic control scenario, where an aircraft makes maneuvering turn in a horizontal plane to the ground, i.e., during maneuver, the height of the vehicle remains constant. The turn rate ω is an unknown constant, which needs to be estimated, making the process model to be a set of nonlinear equations. The target dynamic equation is given by [17, 20, 21, 39–41]

$$x_k = \begin{bmatrix} 1 & \dfrac{\sin(\omega\Delta t)}{\omega} & 0 & \dfrac{\cos(\omega\Delta t)-1}{\omega} & 0 \\[2mm] 0 & \cos(\omega\Delta t) & 0 & -\sin(\omega\Delta t) & 0 \\[2mm] 0 & \dfrac{1-\cos(\omega\Delta t)}{\omega} & 1 & \dfrac{\sin(\omega\Delta t)}{\omega} & 0 \\[2mm] 0 & \sin(\omega\Delta t) & 0 & \cos(\omega\Delta t) & 0 \\[2mm] 0 & 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + v_k. \tag{38}$$

The state of the aircraft denotes $\mathbf{x} = [\xi, \dot{\xi}, \eta, \dot{\eta}, \omega]^T$; ξ and η represent positions, and $\dot{\xi}$ and $\dot{\eta}$ represent velocities in the x and y directions, respectively; Δt is the sampling time. $\mathbf{v}_k$ is the white Gaussian noise with zero mean and covariance $\mathbf{Q}_k$, where

$$\mathbf{Q}_k = \begin{bmatrix} \dfrac{\Delta t^3}{3} & \dfrac{\Delta t^2}{2} & 0 & 0 & 0 \\[2mm] \dfrac{\Delta t^2}{2} & \Delta t & 0 & 0 & 0 \\[2mm] 0 & 0 & \dfrac{\Delta t^3}{3} & \dfrac{\Delta t^2}{2} & 0 \\[2mm] 0 & 0 & \dfrac{\Delta t^2}{2} & \Delta t & 0 \\[2mm] 0 & 0 & 0 & 0 & 1.75 \times 10^{-3}\Delta t \end{bmatrix}. \tag{39}$$

In this problem, a sensor is used to measure the range and angle between the target and the sensor. The measurement equation for the sensor can be expressed by

$$\begin{pmatrix} r_k \\ \theta_k \end{pmatrix} = \begin{pmatrix} \sqrt{(\xi_k - x_o)^2 + (\eta_k - y_o)^2} \\[2mm] a\tan\dfrac{\xi_k - y_o}{\eta_k - x_o} \end{pmatrix} + n_k, \tag{40}$$

where $(\mathbf{x_o}, \mathbf{y_o}) = (200\,\text{m}, 300\,\text{m})$ is the sensor position and $\mathbf{n}_k$ is the white Gaussian measurement noise with zero mean and covariance $\mathbf{R}_k = \text{diag}([1000\,\text{m}^2, 100\,\text{mrad}^2])$. The measurement sampling interval $\Delta t = 1\,\text{s}$.

The initial true state $\mathbf{x}_0 = [\,1000\,\text{m}, \ 300\,\text{m/s}, \ 1000\,\text{m}, \ 0, -3\,°/\text{s}]^T$, and the associated covariance $\mathbf{P}_{0|0} = \text{diag}([\,100\,\text{m}^2, \ 10\,\text{m}^2/\text{s}^2, \ 100\,\text{m}^2, \ 10\,\text{m}^2/\text{s}^2, \ 100\,\text{m rad}^2/\text{s}^2])$. The initial state estimate $\hat{\mathbf{x}}_{0|0}$ is chosen randomly from the Gaussian distribution $N(\mathbf{x}_0, \mathbf{P}_{0|0})$ in each run. For a fair comparison, we make 100 independent Monte Carlo runs.

Performance Metrics. To compare the performances of all filters, we use the root mean square error (RMSE) of the position, velocity, and the turn rate. The RMSE in position at time index k is defined as follows:

$$\text{RMSE}_{\text{pos}}(k) = \sqrt{\frac{1}{\text{MC}} \sum_{n=1}^{\text{MC}} \left[\left(\xi_k^n - \widehat{\xi}_k^n\right)^2 + \left(\eta_k^n - \widehat{\eta}_k^n\right)^2 \right]}, \tag{41}$$

where n is the number of Monte Carlo runs; ξ_k^n and η_k^n are the position elements of the true state at time k in the n-th Monte Carlo run; and $\widehat{\xi}_k^n$ and $\widehat{\eta}_k^n$ are the corresponding estimated state vectors. Similarly, we may easily write the formulas of RMSEs in velocity and the turn rate.

The simulation results, including the position RMSE, velocity RMSE, and the turn rate RMSE of various filters, are shown in Figures 4–6, respectively. Due to the fact that there is not much difference among the filters in the interval of 0–30 s, the RMSEs of these steps are not shown here. The average time in Table 4 is the running time for 100 times' simulation, each time 100 steps.

From the results, we can see that SSGQKF5 improves the CKF by 7.05% in position, 4.7684% in velocity, and 1.4368% in turn rate. We can also learn that the RMSE of position estimation error, velocity error, and turn rate error of the SSGQKF5 is much smaller than that of the HDCQKF, which means the SSGQKF5 can provide more accurate estimation results. That is because the fifth-degree simplex rule is utilized to compute the spherical integral; hence, the filter precision can be improved.

Table 4 shows the runtime of the six filters; the runtime of SSGQKFs is slower than that of HDCQKF, which is because the fifth-degree spherical simplex rule requires fewer points than the fifth-degree CKF when the dimension is larger than four. To sum up, we claim that the SSGQKFs is a balance between filtering accuracy and computational cost for some high-dimensional applications.
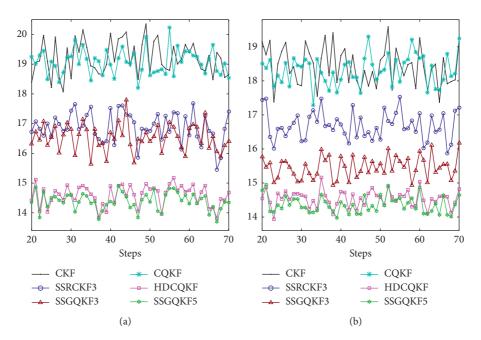
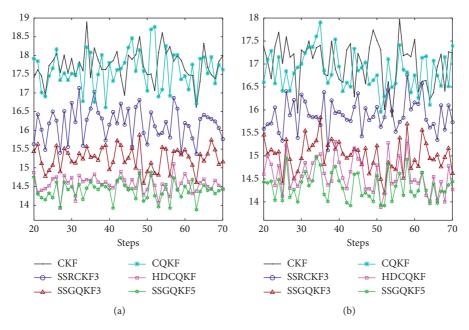Figure 1: (a) RMSE of first state. (b) RMSE of second state.

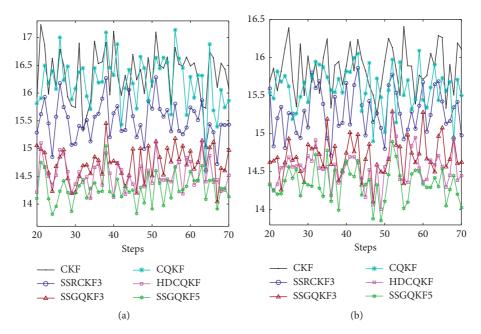Figure 2: (a) RMSE of third state. (b) RMSE of fourth state.

FIGURE 3: (a) RMSE of fifth state. (b) RMSE of sixth state.

TABLE 3: RMSE averaged over time horizon.

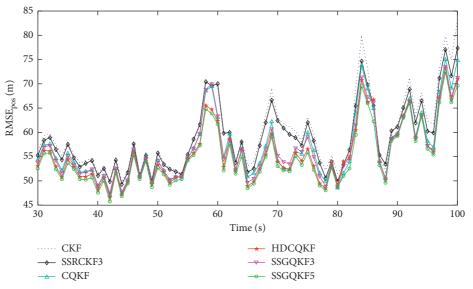| State	CKF	SSRCKF3	CQKF	HDCQKF	SSGQKF3	SSGQKF5
1	19.2544	16.8918	19.0977	14.6671	16.5971	14.4069
2	18.4710	16.7303	18.3860	14.6332	15.5015	14.4000
3	17.8151	16.2794	17.6828	14.6456	15.3515	14.4251
4	17.1051	15.9543	16.9210	14.6522	15.0779	14.3768
5	16.4188	15.5462	16.2267	14.5895	14.8098	14.3143
6	15.8976	15.3297	15.6327	14.6457	14.7384	14.3693
7	15.3305	15.0235	15.1787	14.6328	14.6056	14.3941
8	15.0005	14.8623	14.8227	14.6442	14.5846	14.3755
9	14.7918	14.7190	14.6049	14.6931	14.5072	14.4273
10	14.7144	14.6500	14.5308	14.6531	14.4309	14.3736

FIGURE 4: RMSE for position.

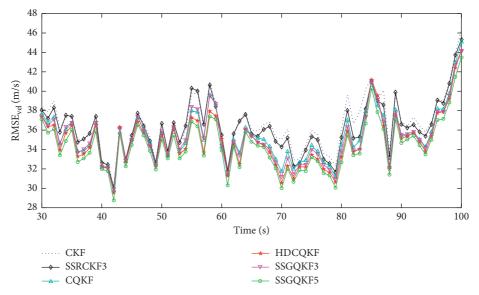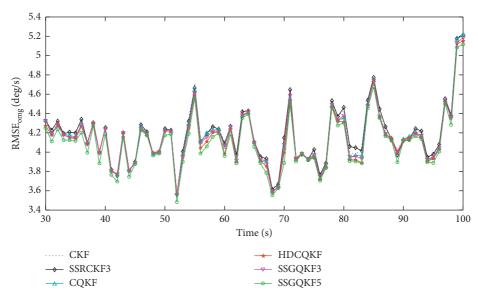Figure 5: RMSE for velocity.

Figure 6: RMSE for turn rate.

Table 4: The RMSE and average time of various filters.

RMSE	CKF	SSRCKF3	CQKF	HDCQKF	SSGQKF3	SSGQKF5
Position (m)	54.5312	53.4788	51.9089	51.1859	51.7158	50.6870
Velocity (m/s)	34.9573	34.6936	34.0029	33.6358	33.9200	33.2904
ω (°)	4.1760	4.1726	4.1537	4.1412	4.1534	4.1160
Average time (s)	0.1094	0.1250	0.1563	0.4688	0.2031	0.4063

5. Conclusions

In this paper, a new method has been proposed for solving the nonlinear filtering problems. The proposed filter is named as spherical simplex Gauss–Laguerre quadrature cubature Kalman filter. The intractable integrals encountered during the filtering are decomposed into surface and line integrals. The surface integral is solved by spherical simplex cubature rule, while the line integral is solved by any ordered Gauss–Laguerre quadrature rule. SSGQKFs use the spherical simplex Gauss–Laguerre quadrature rule and choose a set of SCQ points, which do not require the evaluation of Jacobian matrices during the prediction and update stage. The proposed filter is tested in two problems that involve a high nonlinear state estimation problem and the use of a target tracking problem of practical significance.

The accuracy of the proposed filter is found to be better than the CKF, the CQKF, and the HDCKF. The proposed nonlinear filters can provide more accurate results with lower computational costs than HDCQKF in high-dimensional nonlinear systems. To enhance accuracy, the proposed filter can be potentially used in state estimation where high nonlinearity exists in systems.

Data Availability

The raw/processed data required to reproduce these findings cannot be shared at this time as the data also form part of an ongoing study.

Conflicts of Interest

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

Acknowledgments

References

[1] Y. Wang, "Multitarget tracking by improved particle filter based on H_∞ unscented transform," *Mathematical Problems in Engineering*, vol. 171, Article ID 483913, 7 pages, 2013.

[2] Z. Li, W. Yang, and D. Ding, "A novel fifth-degree strong tracking cubature Kalman filter for two-dimensional maneuvering target tracking," *Mathematical Problems in Engineering*, vol. 2018, Article ID 5918456, 10 pages, 2018.

[3] Y. Han and C. Han, "A Bayesian approach to track multiple extended targets using particle filter for nonlinear system," *Mathematical Problems in Engineering*, vol. 2018, Article ID 7424538, 10 pages, 2018.

[4] X. Wu and K. Ma, "Improved Student's *t*-based unscented filter and its application to trajectory estimation for maneuvering target," *Applied Sciences*, vol. 9, no. 11, p. 2186, 2019.

[5] Y. Shalom, X. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, Wiley, New York, NY, USA, 2001.

[6] X. Kang, G. He, and X. Li, "A SINS/BDS integrated navigation method based on classified weighted adaptive filtering," *Mathematical Problems in Engineering*, vol. 2019, Article ID 2158351, 6 pages, 2019.

[7] H. Jiang and Y. Cai, "Adaptive fifth-degree cubature information filter for multi-sensor bearings-only tracking," *Sensors*, vol. 18, no. 10, p. 3241, 2018.

[8] F. Gustafsson, F. Gunnarsson, N. Bergman et al., "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425–437, 2002.

[9] X. Liu, L. Li, Z. Li et al., "Event-trigger particle filter for smart grids with limited communication bandwidth infrastructure,"

[10] S. Li, L. Li, Z. Li et al., "Event-trigger heterogeneous nonlinear filter for wide-area measurement systems in power grid," *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 2752–2764, 2019.

[11] S. Li, Y. Hu, L. Zheng et al., "Stochastic event-triggered cubature Kalman filter for power system dynamic state estimation," *IEEE Transactions on Circuits & Systems II Express Briefs*, vol. 66, no. 9, pp. 1552–1556, 2019.

[12] P. A. Ejegwa and J. A. Awolola, "Novel distance measures for Pythagorean fuzzy sets with applications to pattern recognition problems," *Granular Computing*, vol. 1, pp. 1–9, 2019.

[13] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[14] M. Simandl, J. Karlovec, and T. Soderstorm, "Advanced pointmass method for nonlinear state estimation," *Automatica*, vol. 42, no. 7, pp. 1133–1145, 2006.

[15] S. P. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, "Linear matrix inequalities in system and control theory," *Studies in Applied Mathematics*, vol. 15, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, USA, 1994.

[16] Y. Yin, Z. Li, X. Liu et al., "A nested tensor product model transformation," *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 1, pp. 1–15, 2019.

[17] J. D. Watson and F. H. C. Crick, "Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid," *Nature*, vol. 171, no. 4356, pp. 737-738, 1953.

[18] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[19] K. Ito and K. Xiong, "Gaussian filters for nonlinear filtering problems," *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 910–927, 2000.

[20] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1254–1269, 2009.

[21] B. Jia, M. Xin, and Y. Cheng, "High-degree cubature Kalman filter," *IEEE Conference on Decision and Control*, vol. 171, no. 4356, pp. 4095–4100, 2012.

[22] S. Wang, J. Feng, and C. K. Tse, "Spherical simplex-radial cubature Kalman filter," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 43–46, 2014.

[23] S. Bhaumik, "Higher degree cubature quadrature Kalman filter," *International Journal of Control Automation and Systems*, vol. 13, no. 5, pp. 1–9, 2015.

[24] S. Bhaumik and fnm Swati, "Cubature quadrature Kalman filter," *IET Signal Processing*, vol. 7, no. 7, pp. 533–541, 2013.

[25] J. Lu and D. L. Darmofal, "Higher-dimensional integration with Gaussian weight for applications in probabilistic design," *SIAM Journal on Scientific Computing*, vol. 26, no. 2, pp. 613–624, 2004.

[26] A. Genz and J. Monahan, "Stochastic integration rules for infinite regions," *SIAM Journal on Scientific Computing*, vol. 19, no. 2, pp. 426–439, 1998.

[27] I. Mysovskih, *Approximate Calculation of Integrals*, Springer, Dordecht, The Netherlands, 1969.

[28] F. B. Hildebrand, *Introduction to Numerical Analysis*, Dover Publications, Mineola, NY, USA, 2nd edition, 2008.

[29] J.-S. Hu and C.-H. Yang, "Second-order extended-filter for nonlinear discrete-time systems using quadratic error matrix

approximation," *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3110–3119, 2011.

[30] C. H. Tseng, S. F. Lin, and D. J. Jwo, "Robust Huber-based cubature Kalman filter for GPS navigation processing," *Journal of Navigation*, vol. 70, no. 3, pp. 1–20, 2016.

[31] H. Wang, H. Li, W. Zhang, J. Zuo, and H. Wang, "Maximum correntropy derivative-free robust Kalman filter and smoother," *IEEE Access*, vol. 6, pp. 70794–70807, 2018.

[32] G. Hao and S. Sun, "Distributed fusion cubature Kalman filters for nonlinear," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 1, pp. 5979–5991, 2019.

[33] G. Hao, S.-l. Sun, and Y. Li, "Nonlinear weighted measurement fusion unscented Kalman filter with asymptotic optimality," *Information Sciences*, vol. 299, pp. 85–98, 2015.

[34] B. Li, Z. Li, X. Chen, Y. Huang, and X. Liu, "Recognition and vulnerability analysis of key nodes in power grid based on complex network centrality," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 3, pp. 346–350, 2018.

[35] F. Daum and J. Huang, "Particle degeneracy: root cause and solution," in *Proceeding of SPIE*, vol. 8050, pp. 80500W–80500W-11, Orlando, FL, USA, May 2011.

[36] F. Daum, J. Huang, and A. Noushin, "Generalized Gromov method for stochastic particle flow filters," in *SPIE Conference on Signal Processing*, vol. 10200, pp. 1–13, Anaheim, CA, USA, April 2017.

[37] L. Chang, B. Hu, A. Li, and F. Qin, "Transformed unscented Kalman filter," *IEEE Transactions on Automatic Control*, vol. 58, no. 1, pp. 252–257, 2013.

[38] A. Singh and S. Bhaumik, "Improved high-degree cubature Kalman filter," in *Proceedings of 2015 Sensor Signal Processing for Defence*, pp. 9-10, Edinburgh, UK, September 2015.

[39] D. Meng, L. Miao, H. Shao, and J. Shen, "A seventh-degree cubature Kalman filter," *Asian Journal of Control*, vol. 20, no. 1, pp. 250–262, 2017.

[40] S. Sarkka and J. Hartikainen, "On Gaussian optimal smoothing of non-linear state space models," *IEEE Transactions on Automatic Control*, vol. 55, no. 8, pp. 1938–1941, 2010.

[41] A. Singh and S. Bhaumik, "Cubature quadrature Kalman filter for maneuvering target tracking," in *2015 International Conference on Smart Sensors and Application*, pp. 26–28, Kuala Lumpur, Malaysia, May 2015.