Rauch-Tung-Striebel High-Degree Cubature Kalman Smoother

Bin Jia and Ming Xin

Abstract—In this paper, a new Rauch-Tung-Striebel type of nonlinear smoothing method is proposed based on a class of high-degree cubature rules. This high-degree cubature Kalman smoother generalizes the conventional third-degree cubature Kalman smoother and considerably improves its estimation accuracy. A target tracking problem is utilized to demonstrate the performance of this new smoother and compare it with other Gaussian approximation smoothers. It will be shown that the high-degree cubature Kalman smoother outperforms the extended Kalman smoother, the unscented Kalman smoother, the third-degree cubature Kalman smoother, and maintains close performance to the Gauss-Hermite quadrature smoother with much less computational cost.

I. INTRODUCTION

Bayesian filtering and smoothing has a wide range of applications, such as tracking [1,2], navigation [3], and audio signal processing [4]. The filtering algorithm computes the estimate based on the measurements made at or before the time of the estimated state. The smoothing algorithm can improve the filtering performance by using additional measurements made after the time of the estimated state.

Like the optimal filtering, the optimal smoothing is difficult to obtain for general nonlinear dynamic systems. Hence, many different approximation methods have been proposed. With the Gaussian assumption, many smoothing algorithms can be easily designed and have been widely implemented. The most widely used Gaussian smoother is the extended Kalman smoother (EKS) [5]. Similar to the extended Kalman filter (EKF), EKS may not be accurate enough for highly nonlinear problems with large uncertainties. The unscented transformation (UT) [6,7] based unscented Kalman filter (UKF) and the unscented Kalman smoother (UKS) [8] have gained a tremendous popularity since they are more accurate than the EKF and EKS in many applications. Recently, the cubature Kalman filter (CKF) and smoother (CKS) based on the third-degree cubature rule [9,10] have been proposed. The cubature rule in the CKF and CKS exhibits more stable performance than the UT in the UKF or UKS [10]. The numerical approximation to the Gaussian type integrals plays a key role in the nonlinear Gaussian filters and smoothers. Among all the Gaussian type integration rules [11-13], the Gauss-Hermite quadrature (GHQ) rule is more accurate than others including the UT and the cubature rule [11]. However, the GHQ suffers the curse of dimensionality problem when the direct tensor product rule is used to extend the univariate GHQ to the multi-dimensional GHQ. Despite the limitation of the GHQ in high dimensional problems, the GHQ is widely used as a benchmark rule to assess other numerical rules. Similarly, the Gauss-Hermite quadrature smoother (GHQS) can be used as a benchmark to compare the performance of different Gaussian approximation smoothers.

There are two alternative forms of the Bayesian optimal smoothers: the two-filter smoother [5] as well as the Rauch-Tung-Striebel (RTS) smoother [8]. The two-filter smoother is based on a linear combination of two filters, which are run in forward and backward directions using any Kalman type of filters. However, since the backward filtering is based on the inverse of the forward dynamic model, it may not lead to the correct result [14]. The RTS smoother combines the backward filter and smoother into one single backward recursion and is more widely used in practice than the two-filter scheme. In this paper, a new RTS smoother is proposed based on high-degree cubature rules using Genz’s method [15]. Although the third-degree CKS works well for many applications [10], it may not provide accurate enough results when high nonlinearity and large uncertainty exist in the dynamic system. Hence, the new high-degree CKS can considerably improve the estimation performance. In addition, the high-degree cubature rule can achieve close accuracy to the GHQ rule. Unlike the GHQ, the high-degree cubature rule does not suffer the curse of dimensionality problem since the number of points required increases polynomially with the dimension. It can be shown that the new proposed smoother can achieve more accurate results than the EKS and UKS, and is computationally more efficient than the GHQS.

The remainder of this paper is organized as follows. In Section II, the point-based Gaussian approximation filter and smoother are introduced. The new high-degree cubature rule based smoother is proposed in Section III. In Section IV, the performance of the high-degree CKS is compared with other smoothers. Concluding remarks are given in Section V.

II. POINT-BASED GAUSSIAN APPROXIMATION ESTIMATION

In this section, point-based Gaussian approximation filters and smoothers are briefly reviewed. Consider a class of nonlinear discrete-time dynamical systems described by:

\[ x_k = f(x_{k-1}) + v_{k-1} \]  
\[ y_k = h(x_k) + n_k \]

where \( x_k \in \mathbb{R}^n; y_k \in \mathbb{R}^m; \ v_{k-1} \text{ and } n_k \) are independent white Gaussian process noise and measurement noise with covariance \( Q_{k-1} \) and \( R_k \), respectively.

A. Point-based Gaussian approximation filter

Due to the strong relation between the filtering and smoothing, the point-based filtering algorithms are given as follows [12, 16].
\[
\hat{x}_{k|k-1} = \sum_{i=1}^{N_p} W_i f(\tilde{\xi}_{k|i})
\]
\[P_{k|k-1} = \sum_{i=1}^{N_p} W_i \left( f(\tilde{\xi}_{k|i}) - \hat{x}_{k|i} \right) \left( f(\tilde{\xi}_{k|i}) - \hat{x}_{k|i} \right)^T + Q_{k-1}
\]
where \(N_p\) is the transformed number of points; \(\tilde{\xi}_{k|i}\) is the transformed quadrature point given by
\[
P_{k|k-1} = S_{k-1} S_{k|k-1}^T; \quad \tilde{\xi}_{k|i} = S_{k-1} y_i + \hat{x}_{k|i}
\]
\[
\hat{x}_{k|i} = \hat{x}_{k|i} + L_k \left( y_k - z_k \right)
\]
\[
P_k = P_{k|i} - L_k P_{zz}^{-1} L_k^T
\]
where \(L_k = P_{zz} \left( R_k + P_{zz}^{-1} \right)^{-1}\)
\[
P_x = \sum_{i=1}^{N_p} W_i \left( \tilde{\xi}_{k|i} - \hat{x}_{k|i} \right) \left( \tilde{\xi}_{k|i} - \hat{x}_{k|i} \right)^T
\]
\[
P_z = \sum_{i=1}^{N_p} W_i \left( h(\tilde{\xi}_{k|i}) - z_k \right) \left( h(\tilde{\xi}_{k|i}) - z_k \right)^T
\]
where \(\tilde{\xi}_{k|i}\) is the transformed quadrature point given by
\[
P_{k|i} = S_k S_{k|i}^T; \quad \tilde{\xi}_{k,i} = S_k y_i + \hat{x}_{k|i}
\]
\(y_i\) and \(W_i\) are quadrature points and weights used to approximate the following Gaussian weighted integral:
\[
\int_{\mathbb{R}^n} g(x) N(x;0,1) dx \approx \sum_{i=1}^{N_p} W_i g(y_i)
\]
where \(N(x;0,1)\) denotes the Gaussian distribution with mean 0 and unit covariance.

### B. Point-based Gaussian approximation smoother

In this paper, the forward-backward RTS smoother is used and can be described as follows.
\[
p(x_k | y_{1:T}) = p(x_k | y_{1:k}) \int \frac{p(x_{k+1} | x_k) p(x_{k+1} | y_{1:k})}{p(x_{k+1} | y_{1:k})} dx_{k+1}
\]
where \(p(x_k | y_{1:k})\) and \(p(x_k | y_{1:k})\) are the smoothing pdf and the filtering pdf at the time \(k\), respectively; \(p(x_{k+1} | y_{1:k})\) is the predicted pdf at the time \(k+1\). The smoothing state can be recursively obtained backward from the last time \(k=T\).

Eq. (14) is rarely used in practice because the multivariate integrations in Eq. (14) are intractable in general. Hence, the following approximation is often made [8].

The conditional distribution of \(x_k\) is given by
\[
p(x_k | x_{k+1}, y_{1:k}) = \frac{p(x_k, x_{k+1} | y_{1:k})}{p(x_{k+1} | y_{1:k})}
\]
where \(p(x_k, x_{k+1} | y_{1:k})\) and \(p(x_{k+1} | y_{1:k})\) can be obtained by
\[
p(x_k, x_{k+1} | y_{1:k}) = p(x_{k+1} | x_k) p(x_k | y_{1:k})
\]
and
\[
p(x_{k+1} | y_{1:k}) = \int p(x_{k+1} | x_k) p(x_k | y_{1:k}) dx_k
\]
Using
\[
p(x_k, x_{k+1} | y_{1:k}) = p(x_k | x_{k+1}, y_{1:k}) p(x_{k+1} | y_{1:k})
\]
and marginalizing the joint distribution \(p(x_k, x_{k+1} | y_{1:k})\) over \(x_{k+1}\), the smoothing pdf \(p(x_k | y_{1:k})\) can be obtained by
\[
p(x_k | y_{1:k}) = \int p(x_k | x_{k+1}, y_{1:k}) p(x_{k+1} | y_{1:k}) dx_{k+1}
\]
With the Markov properties of the state-space model, one has
\[
p(x_k | x_{k+1}, y_{1:k}) = p(x_k | x_{k+1}, y_{1:k})
\]
which can be calculated by Eq. (15).

Under the Gaussian assumption, we assume \(p(x_k | y_{1:k})\) is represented by \(N(\hat{x}_k, \hat{P}_k)\). After some algebra, by Eqs. (15)-(19), \(\hat{x}_k^s\) and \(P_k^s\) can be obtained [16, 17].

\[
\hat{x}_k^s = \hat{x}_{k|i} + \tilde{d}_k \left( \tilde{\xi}_{k|i} - \hat{x}_{k|i} \right)
\]
\[
P_k^s = P_k + \tilde{d}_k \left( P_{zz} + P_{zz}^{-1} \right) \tilde{d}_k^T
\]
where \(\tilde{d}_k = C_{k+1|i} P_{zz}^{-1} \tilde{\xi}_{k|i}\)
\[
C_{k+1|i} = \sum_{i=1}^{N_p} W_i \left( \tilde{\xi}_{k,i} - \hat{x}_{k|i} \right) \left( f(\tilde{\xi}_{k,i}) - \hat{x}_{k+i} \right)^T
\]
Note that \(\hat{x}_{1:T} = \hat{x}_{T|T}\) and \(P_{1:T} = P_{T|T}\) are the smoother state and covariance at time \(T\), respectively.

### Remark 2.1: In both Bayesian filtering algorithm (3)-(12) and smoothing algorithm (20)-(23), the quadrature approximation to calculate the Gaussian weighted integral (13) is the core. Many numerical rules can be utilized, such as the UT [6, 7], the GHQ rule [11], the sparse-grid quadrature (SGQ) rule [12], and the cubature rule [9, 10]. In this paper, we propose a class of high-degree cubature rules to improve the performance of the smoother.

### III. HIGH-DEGREE CUBATURE RULES

For the cubature rule, the following integral is considered:
\[
I(g) = \int_{R^n} g(x) \exp(-x^T x) dx
\]
Let \(x = rs\) with \(s^T s = 1\) and \(r = \sqrt{x^T x}\). Equation (24) can be transformed in the spherical-radial coordinate system [9]
\[
I(g) = \int_{U_n} \int_{S_{n-1}} g(rs) r^{n-1} \exp(-r^2) d\sigma(s) dr
\]
where \(s = [s_1, s_2, \ldots, s_n]^T, U_n = \{s \in \mathbb{R}^n : s_1^2 + s_2^2 + \cdots + s_n^2 = 1\}\), and \(\sigma(\cdot)\) is the area element on \(U_n\).

The accuracy of the cubature rule can be assessed by the polynomial approximation degrees. A numerical integration
rule \( \int_{\mathbb{R}^d} g(x) w(x) \, dx = \sum_i w_i g(x_i) \) is defined to be a \( d \)-th degree rule if it is exact for \( g(x) \) whose components are linear combinations of monomials \( x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n} \) with the total degree up to \( d \) (\( \alpha_1, \alpha_2, \cdots, \alpha_n \) are nonnegative integers and \( 0 \leq \alpha_1 + \alpha_2 + \cdots + \alpha_n \leq d \)).

Equation (25) contains two types of integrals: the radial integral \( \int_{C_0} g_r(r) r^{n-1} \exp(-r^2) \, dr \) with the weighting function \( w_r(r) = r^{n-1} \exp(-r^2) \) and the spherical integral \( \int_{S_0} g_s(s) \, d\sigma(s) \) with the weighting function \( w_s(s) = 1 \). The cubature rule is a \( d \)-th degree rule if the radial rule and the spherical rule are all \( d \)-th degree rules.

If the radial integral can be approximated by an \( N_r \)-point radial rule
\[
\int_{C_0} g_r(r) r^{n-1} \exp(-r^2) \, dr \approx \sum_{j=1}^{N_r} w_{r,j} g_r(r_j)
\]
and the spherical integral can be approximated by an \( N_s \)-point spherical rule,
\[
\int_{S_0} g_s(s) \, d\sigma(s) \approx \sum_{j=1}^{N_s} w_{s,j} g_s(s_j)
\]
then the integral (25) can be approximated by,
\[
I(g) = \int_{C_0} g_r(r) r^{n-1} \exp(-r^2) \int_{S_0} g_s(rs) \, d\sigma(s) \, dr
\approx \int_{C_0} g_r(r) r^{n-1} \exp(-r^2) \sum_{j=1}^{N_s} w_{s,j} g_s(rs_j) \, dr
\approx \sum_{i=1}^{N_r} \sum_{j=1}^{N_s} w_{r,i} w_{s,j} g_r(r_i) g_s(s_j)
\]
where \( r_i \) and \( w_{r,i} \) are the points and weights for calculating the radial integral; \( s_j \) and \( w_{s,j} \) are the points and weights for calculating the spherical integral. The Gaussian weighted integral (13) can be calculated using the following fact and the cubature rule (28),
\[
\int_{\mathbb{R}^n} g(x) N(x; x_0, 1) \, dx = \frac{1}{\pi^{n/2}} \int_{\mathbb{R}^n} g(\sqrt{2}x) \exp(-x^T x) \, dx
\approx \frac{1}{\pi^{n/2}} \sum_{i=1}^{N_r} \sum_{j=1}^{N_s} w_{r,i} w_{s,j} g(\sqrt{2}r_i s_j)
\]

With (29), the integral (3)-(4), (9)-(11), and (23) in the filtering and smoothing algorithms can be computed.

A. Arbitrary-degree spherical rule

The arbitrary degree spherical rule is difficult to obtain. Although many different methods can be used to obtain the spherical rule with specific accuracy, to the best of the authors’ knowledge, a rule that can provide arbitrary accuracy is only given by Genz’s method [15].

**Theorem 3.1** [15]: For the spherical integral \( I_{U_n}(g_s) = \int_{S_0} g_s(s) \, d\sigma(s) \), \( I_{U_n,m+1}(g_s) = \sum_{\|p\| = m} w_p G(u_p) \) (\( m \geq 1 \)) is a \( (2m+1)^{th} \)-degree rule, where \( I_{U_n} \) denotes a spherical integral and \( I_{U_n,m+1} \) denotes the \( (2m+1)^{th} \)-degree spherical rule used to approximate the integral.

\[
w_p \triangleq I_{U_n}\left( \sum_{i=1}^{n} \prod_{j=1}^{m-1} (2x_i - u_i^2 - u_j^2) \right)
\]

\[
G(u_p) \triangleq 2^{-c(p)} \sum_v g_s(v_1 u_{p_1}, v_2 u_{p_2}, \ldots, v_n u_{p_n})
\]

where the right-hand side of Eq. (30) is a spherical integral with the integral variables \( s_j \); the subscripts \( p_i \) in Eqs. (30) and (31) are natural numbers with \( p = [p_1, p_2, \cdots, p_n] \) and \( |p| = p_1 + p_2 + \cdots + p_n \); the superscript \( c(u_p) \) in Eq. (31) is the number of nonzero entries in \( u_p = (u_{p_1}, u_{p_2}, \cdots, u_{p_n}) \); \( u_{p_j} \) are chosen to be \( u_{p_j} = \sqrt{p_i/m} \) (\( p_i = 0, \cdots, m \); the points of the spherical rule \( I_{U_n,m+1} \) are given by \( [v_1 u_{p_1}, v_2 u_{p_2}, \cdots, v_n u_{p_n}] \) with \( v_i = \pm 1 \). The weight on the point \( [v_1 u_{p_1}, v_2 u_{p_2}, \cdots, v_n u_{p_n}] \) is \( 2^{-c(u_p)} w_p \).

The 1st-degree spherical rule is trivial and thus \( m \geq 1 \) is assumed in the Theorem.

By Theorem 3.1, the third-degree (\( m=1 \)) spherical rule can be obtained
\[
I_{U_n,3}(g_s) = \frac{A_n}{2n} \sum_{j=1}^{n} \left( g_s(e_j) + g_s(-e_j) \right)
\]

where \( e_j \) is the unit vector in \( \mathbb{R}^n \) with the \( j \)-th element being 1;
\( A_n = 2 \Gamma(n/2)/\Gamma(n/2) = 2 \sqrt{\pi^n} \Gamma(n/2) \) is the surface area of the unit sphere and \( \Gamma(z) \) is the gamma function defined by the integral \( \Gamma(z) = \int_0^\infty \exp(-\lambda) \lambda^{z-1} \, d\lambda \).

Similarly, the fifth-degree spherical rule when \( m=2 \) can be obtained by
\[
I_{U_n,5}(g_s) \triangleq \sum_{j=1}^{n(n+2)/2} \left( g_s(s_j) + g_s(-s_j) + g_s(s_j) + g_s(-s_j) \right)
+ \sum_{j=1}^{n} \left( g_s(e_j) + g_s(-e_j) \right)
\]

where \( s_j \) and \( e_j \) are the weights given by
\[
\overline{w}_{s1} = \frac{A_n}{n(n+2)}
\]
\[
\overline{w}_{s2} = \frac{(4-n)A_n}{2n(n+2)}
\]

and the point sets of \( s_j \) and \( e_j \) are given by
\( \{ s_j \} \triangleq \left\{ \frac{1}{\sqrt{2}} (e_k + e_l) : k < l, k, l = 1, 2, \cdots, n \right\} \) \hspace{1cm} (36)

\( \{ s_j \} \triangleq \left\{ \frac{1}{\sqrt{2}} (e_k - e_l) : k < l, k, l = 1, 2, \cdots, n \right\} \) \hspace{1cm} (37)

The fifth-degree spherical rule has \( 2n^2 \) points and shares \( 2n \) points with the third-degree spherical rule.

### B. Radial rule

The points \( r_i \), and weights \( w_{r,j} \) for the radial rule can be obtained by the moment matching method, which is to satisfy the moment equation of the form

\[
\sum_{i=1}^{N_r} w_{r,j} g_r(r_i) = \int_{0}^{\infty} g_r(r) r^{n-1} \exp(-r^2) dr
\]

where \( g_r(r) = r^l \) is a monomial in \( r \), with \( l \) an even integer. Note that for \( g_r(r) = r^l \), the right-hand side of Eq. (38) reduces to \( \frac{1}{2} \Gamma \left( \frac{n+l}{2} \right) \). Only even-degree monomials need to be matched because the spherical rule and the resultant spherical-radial cubature rule are fully symmetric.

To obtain a \( (2m+1)^{st} \)-degree radial rule for the \( (2m+1)^{st} \)-degree spherical-radial cubature rule, Eq. (38) needs to be exact for \( l = 0, 2, \ldots, 2m \), which contains \((m+1)\) equations.

The minimum number of points needed to satisfy the \((m+1)\) equations is \((m+1)/2\) (for odd \( m \)) or \((m/2+1)\) (for even \( m \)). In this paper, the minimum number of radial quadrature points is used so that the number of points of the spherical-radial cubature rule is minimized.

Now, the third-degree and the fifth-degree radial rules are derived using the moment matching method. For the third-degree radial rule \((m = N_r = 1)\), the following equations need to be satisfied:

\[
w_{r,1} \Gamma \left( \frac{1}{2} \right) = \frac{1}{2} \Gamma \left( \frac{1}{2} n \right)
\]

\[
w_{r,2} \Gamma \left( \frac{1}{2} \right) = \frac{n}{4} \Gamma \left( \frac{1}{2} n \right)
\]

where the last equality follows the identity \( \Gamma (z+1) = z \Gamma (z) \).

Solving Eq. (39) gives the point and weight for the third-degree radial rule,

\[
r_1 = \sqrt{\frac{n}{2}} \quad , \quad w_{r,1} = \Gamma \left( \frac{n}{2} \right)
\]

For the fifth-degree radial rule \((m = N_r = 2)\), the points and weights satisfy the following three equations:

\[
w_{r,1} \Gamma \left( \frac{1}{2} \right) = \frac{1}{2} \Gamma \left( \frac{1}{2} n \right)
\]

\[
w_{r,2} \Gamma \left( \frac{1}{2} \right) = \frac{n}{4} \Gamma \left( \frac{1}{2} n \right)
\]

\[
w_{r,3} \Gamma \left( \frac{1}{2} \right) = \frac{1}{2} \Gamma \left( \frac{1}{2} n + 1 \right) = \frac{n}{4} \Gamma \left( \frac{1}{2} n \right)
\]

Since there are three equations and four variables in Eq. (41), there is one free variable. We can choose \( r_1 \) as the free variable and set it to 0. Solving these three equations gives the points and weights for the fifth-degree radial rule,

\[
r_1 = 0
\]

\[
r_2 = \sqrt{\frac{n}{2}} n + 1
\]

\[
w_{r,1} = \frac{1}{2} \Gamma \left( \frac{1}{2} n \right) - \frac{n}{2} \Gamma \left( \frac{1}{2} n + 1 \right) = \frac{1}{2} \Gamma \left( \frac{1}{2} n \right)
\]

\[
w_{r,2} = \frac{n}{2} \Gamma \left( \frac{1}{2} n + 1 \right) = \frac{n}{2} \Gamma \left( \frac{1}{2} n \right)
\]

### Remark 3.1: The free variable \( r_1 \) can take on other values than 0. However, the number of points of the spherical-radial cubature rule is minimum when \( r_1 \) is set to 0.

### C. Cubature rules

Combining Eqs. (29), (32), (39), and (40), the third-degree cubature rule \((N_r = 1, N_s = 2n)\) is given by

\[
\int_{x} g(x) N(x; 0, 1) dx = \frac{1}{\pi n^2} \sum_{i=0}^{N_r} w_{r,i} g \left( \sqrt{n} s_i \right)
\]

\[
= \frac{1}{\pi n^2} \sum_{i=0}^{N_r} \sum_{j=0}^{N_s} w_{r,j} w_{s,j} g \left( \sqrt{n} s_i \right)
\]

\[
= \frac{1}{\pi n^2} \sum_{i=0}^{N_r} \sum_{j=0}^{N_s} \left( \Gamma \left( \frac{n}{2} \right) \sqrt{\frac{n}{2}} e_j + g \left( -\sqrt{\frac{n}{2}} e_j \right) \right)
\]

\[
= \frac{1}{\pi n^2} \sum_{i=0}^{N_r} \sum_{j=0}^{N_s} \left[ g \left( \sqrt{n} e_j \right) + g \left( -\sqrt{n} e_j \right) \right]
\]

Combining Eqs. (29), (33), (42), and (43), the fifth-degree cubature rule \((N_r = 2, N_s = 2n^2)\) is given by

\[
\int_{x} g(x) N(x; 0, 1) dx \approx \frac{1}{\pi n^2} \sum_{i=0}^{N_r} \sum_{j=0}^{N_s} w_{r,i} w_{s,j} g \left( \sqrt{n} s_i \right)
\]

\[
= \frac{2}{n+2} \Gamma \left( \frac{n}{2} \right) + \frac{1}{(n+2)^{s-1}/2} \sum_{j=1}^{s} \left( g \left( \sqrt{n+2} s_i \right) + g \left( -\sqrt{n+2} s_i \right) \right)
\]

\[
+ \frac{2-4}{2(n+2)^{s-1}/2} \sum_{j=1}^{s} \left( g \left( \sqrt{n+2} e_j \right) + g \left( -\sqrt{n+2} e_j \right) \right)
\]

Note that \( w_{r,j} = \overline{w}_{r,j} \) for \( j = 1, \ldots, 2n(n-1) \) and \( w_{s,j} = \overline{w}_{s,j} \) for \( j = 2n(n-1)+1, \ldots, 2n^2 \). The number of points of the fifth-degree cubature rule is \( 2n^2 + 1 \).
Remark 3.2: The third-degree cubature rule used in the CKF or CKS of ([9, 10]) has the same form as Eq. (44) and can be viewed as a special case of the proposed cubature rules. For convenience, the procedures to generate cubature points and weights can be summarized in Algorithm I.

Algorithm I: Generate Cubature Points and Weights

\[ \chi, W = \text{CubatureRule}[n, m] \]

( \chi : Cubature point set with the element of \( \chi_k \); \( W \): weight sequence with the element of \( W_k \))

Obtain the spherical points and weights as follows:

Obtain the radial points and weights \( r_i \) and \( w_{r,i} \).

FOR each possible \( [p_1, p_2, \cdots, p_n] \), form

\[ s_j = \left[ v_1 u_{p_1}, v_2 u_{p_2}, \cdots, v_n u_{p_n} \right]^T \]

Calculate the weight \( w_{s,j} = 2^{-n}(s_r) w_p \)

END

Obtain the radial points and weights \( r_i \) and \( w_{r,i} \).

END

Note that the number of points of the cubature rule increases polynomially with the dimension and thus it can avoid the curse of dimensionality existing in the GHQ rule.

The high-degree cubature smoother can be obtained when the high-degree cubature rule is used in the Gaussian approximation filter and smoother framework, i.e., used in Eqs. (3)-(4), (9)-(11), and (23) to approximate the integrals.

IV. SIMULATION RESULTS

In this section, a target tracking problem is considered to demonstrate performance of the high-degree CKS. It has been used as a benchmark problem to validate the performance of the third-degree CKF [9]. The fifth-degree CKS based on Eq. (45) is compared with the EKS, the UKS, the third-degree CKS, and the GHQS. The UKF and UKS parameter \( \kappa \) is chosen to be the suggested \( \kappa = 3 - n \) [7].

The dynamic equation of the target tracking is given by

\[
x_k = \begin{bmatrix}
1 & \sin(\omega_k \Delta t) & 0 & \cos(\omega_k \Delta t) - 1 & 0 \\
\omega_{k-1} & 0 & \cos(\omega_k \Delta t) & 0 & \sin(\omega_k \Delta t) \\
0 & 1 & \sin(\omega_k \Delta t) & 0 & \sin(\omega_k \Delta t) \\
0 & 0 & \omega_{k-1} & 0 & \omega_{k-1} \\
0 & 0 & 0 & 0 & 1
\end{bmatrix} x_{k-1} + v_{k-1}
\]

where \( x_k = [x_k, \dot{x}_k, y_k, \dot{y}_k, \omega_k]^T \) and \([\dot{x}_k, \dot{y}_k] \) are the position and velocity at time \( k \), respectively; \( \omega_{k-1} \) is the unknown turn rate at time \( k-1 \); \( v_{k-1} \) is the white Gaussian noise with mean zero and covariance \( Q_{k-1} \).

\[
Q_{k-1} = \begin{bmatrix}
\Delta t^3/2 & \Delta t^2/2 & 0 & 0 & 0 \\
\Delta t^2/2 & \Delta t & 0 & 0 & 0 \\
0 & 0 & \Delta t^3/2 & \Delta t^2/2 & 0 \\
0 & 0 & \Delta t^2/2 & \Delta t & 0 \\
0 & 0 & 0 & 0 & 1.75 \times 10^{-3} \Delta t
\end{bmatrix}
\]

The measurement equation is given by

\[
y_k = \left( \sqrt{x_k^2 + y_k^2} \right) + n_k
\]

where atan2 is the four-quadrant inverse tangent function; \( n_k \) is the white Gaussian measurement noise with zero mean and covariance \( R_k = \text{diag}(1000 \text{ m}^2, 100 \text{ mrad}^2) \). The measurement sampling interval is \( \Delta t = 1 \text{ s} \).

The simulation results are based on 100 Monte Carlo runs. The initial estimate \( \hat{x}_0 \) is generated randomly from the normal distribution \( N(\hat{x}_0, x_0, P_0) \) with \( x_0 \) being the true initial value \( x_0 = \left[ 1000 \text{ m}, 300 \text{ m/s}, 1000 \text{ m}, 0, -3^\circ/\text{s} \right]^T \) and \( P_0 \) being the initial covariance

\[
P_0 = \text{diag}\left( \left[ 100 \text{ m}^2, 10 \text{ m}^2/\text{s}^2, 100 \text{ m}^2, 10 \text{ m}^2/\text{s}^2, 100 \text{ mrad}^2/\text{s}^2 \right] \right)
\]

The metric used to compare the performance of various filters and smoothers is the root mean square error (RMSE). The RMSEs of the position, velocity, and turn rate are shown in Figs. 1, 2, and 3, respectively. The results of the EKF and EKS are not shown because they fail to converge in many runs.
Although the GHQS has very close performance to the 5th-degree CKS, it uses much more points than the 5th-degree CKS. Hence, the 5th-degree CKS performs the best for this problem when both the accuracy and the computation complexity are concerned.

V. Conclusion

In this paper, a new Rauch-Tung-Striebel type of high-degree cubature Kalman smoother is proposed based on Genz’s method and the moment matching. It has been shown that the new cubature Kalman smoother provides more accurate estimation results than the extended Kalman smoother, the 3rd-degree cubature smoother, and the uncented Kalman smoother. In addition, it can achieve the same performance as the Gauss-Hermite quadrature smoother but is computationally more efficient since it uses much fewer quadrature points.

REFERENCES