Nagavenkat Adurthi

Mem. ASME
Department of Mechanical and Aerospace
Engineering,
University at Buffalo,
Buffalo, NY 14260-4400
e-mail: nagavenk@buffalo.edu

Puneet Singla

Associate Professor
Mem. ASME
Department of Mechanical and Aerospace
Engineering,
University at Buffalo,
The State University of New York,
Buffalo, NY 14260-4400
e-mail: psingla@buffalo.edu

Tarunraj Singh

Professor
Fellow ASME
Department of Mechanical and Aerospace
Engineering,
University at Buffalo,
The State University of New York,
Buffalo, NY 14260-4400
e-mail: tsingh@buffalo.edu

Conjugate Unscented Transformation: Applications to Estimation and Control

This paper presents a computationally efficient approach to evaluate multidimensional expectation integrals. Specifically, certain nonproduct cubature points are constructed that exploit the symmetric structure of the Gaussian and uniform density functions. The proposed cubature points can be used as an efficient alternative to the Gauss–Hermite (GH) and Gauss–Legendre quadrature rules, but with significantly fewer number of points while maintaining the same order of accuracy when integrating polynomial functions in a multidimensional space. The advantage of the newly developed points is made evident through few benchmark problems in uncertainty propagation, nonlinear filtering, and control applications. [DOI: 10.1115/1.4037783]

1 Introduction

The problem of uncertainty quantification and uncertainty propagation through nonlinear system for various engineering applications requires the evaluation of multidimension expected value integrals with respect to an appropriate probability density function (PDF). The computation of multidimension expected value integrals constitute an important component in defining the computational complexity associated with many uncertainty quantification and propagation methods like stochastic collocation [1,2], polynomial chaos [3,4], Gaussian mixture model [5–7], etc. Analytical expressions for these multidimension integrals in general exist for linear systems. However, in most other cases, one often does not have analytical solution for these integrals and has to approximate the integral values numerically.

Several computation techniques exist in the literature to approximate the expectation integral with respect to a Gaussian PDF and uniform PDF, the most popular being Monte Carlo (MC) methods [8], Gauss–Hermite (GH) and Gauss–Legendre (GLgn) quadrature rules [9], sparse grid quadratures such as Gauss–Hermite Smolyak (GHS) Gauss–Legendre Smolyak (GLgnSM) quadratures [10], unscented transformation (UT) [11], and many other Cubature methods [8]. All these methods involve approximating the expectation integral as a weighted sum of integrand values at specified points within the domain of integration. These methods basically differ from each other in the generation of these specific points. For example, MC methods involve random samples from the specified PDF while the Gaussian quadrature scheme involves deterministic points carefully chosen to reproduce exactly the integrals for polynomials.

Quadrature rules in higher dimensions are usually referred to as cubature rules. A cubature rule is said to be *exact to degree d*, if it can exactly integrate all polynomials with degree $\leq d$ [9]. The Gaussian quadrature methods provide the minimal number of quadrature points to integrate the polynomial function in a one-dimensional (1$\mathbb{D}$) space. One needs N quadrature points according to the Gaussian quadrature scheme to exactly reproduce the expectation integrals of polynomials with degree $2N - 1$ or less in a 1$\mathbb{D}$ space [9]. However, one needs to take the tensor product of 1$\mathbb{D}$ quadrature points to evaluate multidimensional integrals involving polynomial functions in general n-$\mathbb{D}$ space, resulting in a total of N^n quadrature points. Even for a moderate-dimension system involving six random variables, the number of points required to evaluate the expectation integral with only five points along each direction is $5^6 = 15,625$. This is a nontrivial number of points that might make the calculation of the integral computationally expensive, especially when evaluation of the function at each cubature point itself is an expensive procedure, e.g., one may need to solve a system of partial differential equations at each cubature point to compute the function of interest. The sparse grid quadratures or Smolyak quadrature schemes in particular take the sparse product of 1$\mathbb{D}$ quadrature rules and thus have fewer points than the equivalent Gaussian quadrature rules at the cost of introducing negative weights. The negative weights are not desired as the error in computing the integral of a nonpolynomial function by making use of any cubature scheme is proportional to the sum of absolute value of weights [10]. In Refs. [12] and [13], the authors especially emphasize the necessity of cubature points with positive weights for stable and accurate computations of expectation integrals. Fortunately, the Gaussian quadrature rule is *not minimal* for $n \geq 2$ and there exists cubature rules with reduced number of points [9]. For example, the UT is a degree 3 cubature method with only $2n + 1$ points [11]. This forms the motivation for the work presented in this paper.

An extensive amount of work has been performed to develop nonproduct cubature rules, which can exactly reproduce integrals involving polynomials of degree less than or equal to $d = 2N - 1$ with fewer points, i.e., less than N^n. A good description of nonproduct Gaussian cubature rules, particularly for symmetric density functions, that are second, third, and fifth degree exact in any dimension is provided in Ref. [8]. In general, the development of a minimal cubature rule with positive weights of particular degree d that is applicable to any dimension is still an open problem and

---

often the rules are derived individually for certain degrees and dimensions. In Ref. [12], the authors provide a method to construct fully symmetric integration formulas of arbitrary degree in n-$\mathbb{D}$ space, specifically the formulas for degrees 3, 5, 7, 9 were developed in detail, but all these rules have large negative weights. A cubature rule of degree 2 with $n+1$ points and a cubature method of degree 3 with $2n$ points for a general centrally symmetric weight function (such as the Gaussian and Uniform PDF) in n-$\mathbb{D}$ is developed in Ref. [14]. Fully integration rules with minimal points for $2\mathbb{D}$ that are exact for degree 9–15 were developed in Ref. [15]. A 19-point cubature rule was developed, for symmetric PDFs in $2\mathbb{D}$ that are exact for degree 9 or less in Ref. [16]. For a $2\mathbb{D}$ integral, with symmetric weight function, a 12-point cubature rule for degree 7 was developed in Ref. [17]. It is even claimed that this is the minimal number of points required and there exist many such 12-point cubature rules for degree 7 in $2\mathbb{D}$ systems. A complete account of the cubature rules to date is out of the scope of this paper but a collection of minimal cubature rules are documented in Ref. [8] and an updated summary of cubature rules can be found in Refs. [18] and [19].

With regard to nonlinear filtering, where the integrals with Gaussian probability density functions frequently arise, cubature methods such as the unscented Kalman filter (UKF) with reduced number of points have tremendous potential especially in a real-time scenario. The highly acclaimed UT with only $2n+1$ points is a degree 3 cubature method, where these cubature points are called *sigma points* [11]. The $2n+1$ UT points can also capture all higher-order moments with odd exponents due to the symmetrical structure of the points. Julier and Uhlmann [20] derive the simplex sigma points that require only $n+1$ points in n-$\mathbb{D}$ to capture the first two moments. Unlike the conventional UT, these points do not capture higher-order (≥ 3) odd moments. Similar to the UT method, a more recent development is the cubature Kalman filter (CKF), which is again exact to degree 3 but uses only $2n$ points [21]. It can be noticed that the sigma/cubature points of $2n+1$ UT and $2n$ CKF can be observed as special cases of a more general structure of cubature points previously presented in Ref. [14] for symmetrical probability density functions. In Ref. [13], a summary of application sigma points to nonlinear filtering is described.

It is interesting to observe that most of these nonproduct cubature rule possess certain similarities that can be broadly summarized as: (1) they exploit the symmetry of the PDF such as in the Gaussian and Uniform PDFs, (2) a special structure for the cubature points is assumed, and finally (3) a set of nonlinear equations, which are polynomial in nature, are solved. Correspondingly, they also possess some similar drawbacks such as: (1) inconsistency of the set of nonlinear equations, (2) numerical solutions often possess undesirable negative weights and complex, and (3) some cubature points tend to lie outside the support of the PDF. These drawbacks make it very challenging to have a general cubature rules that can be easily constructed for any dimension and order; hence, most cubature rules are constructed for the desired dimension and order. The valuable insight gained by observing the drawbacks of these rules has influenced the development of some new cubature rules with positive weights in this paper.

Building on our prior work [1,22–24], the objective of this paper is to present an unified approach, which we label as conjugate unscented transformation (CUT), to find nonproduct sigma/cubature points with *positive weights* that can exactly integrate polynomial functions of desired order with respect to the Gaussian and uniform probability density functions. It is desired that the new sigma/cubature rule possesses fewer points than the *equivalent* Gaussian quadrature product rules. The CUT approach can be considered as the extension of the unscented transformation to provide higher-order sigma points with positive weights for multivariate Gaussian and uniform probability density functions. It is explicitly shown in this work that one cannot capture higher-order moments by considering additional points along the principal direction used in the unscented transformation. This work presents a systematic way to choose conjugate directions to complement

principal directions to synthesize symmetric higher-order cubature rules. According to the CUT methodology, the cubature points are chosen to lie symmetrically along specially defined directions. For each cubature point, two unknown variables, a weight and a scaling parameter, are assigned. The moment constraints equations (MCEs) for the desired order are derived in terms of the unknown scaling and weight variables. The primary contribution of this paper is the derivation and documentation of few novel cubature rules for degrees 4, 6, and 8. An analytical solution of degree 4 quadrature points is provided for general n-$\mathbb{D}$ space while numerical solutions are presented for degree 6 and 8 quadrature points in a specific dimension for both Gaussian and uniform PDFs. The points have the desired characteristics of all weights being positive and all points lying within the support of the PDF.

The organization of the paper can be summarized as follows: Section 2 introduces the problem of computing multidimensional expectation integrals with quadrature methods. A brief introduction to few popular quadrature rules, particularly the unscented transform, is presented in Sec. 3. An intuitive understanding of how these methods work provides invaluable insight in the development of the CUT methodology. Next, in Sec. 4, the general framework for the CUT approach is described and the newly developed higher-order cubature points are presented. Finally, Sec. 5 presents numerical results illustrating the effectiveness of the CUT approach in evaluating multidimensional expectation integrals relative to conventional approaches in terms of accuracy and computational cost. The application of the CUT methodology to nonlinear filtering and stochastic control is also discussed. Section 6 concludes the paper.

2 Expectation Integral and Cubature Methods

Let us consider the problem of computing the expected value of a real valued function $f(\mathbf{x})\colon \mathbb{R}^n \to \mathbb{R}$ with respect to a multivariate Gaussian or uniform PDF. The multidimensional expectation integral is generally approximated by a linear functional, which can be interpreted as a weighted sum of function evaluations at specific predetermined points, $\mathbf{x_i} \in \mathbb{R}^n$, and weights, $w_i \in \mathbb{R}$

$$E[f(\mathbf{x})] = \int \int \cdots \int f(\mathbf{x})p(\mathbf{x})dx_1 dx_2 \cdots dx_n \approx \sum_{i=1}^{N} w_i f(\mathbf{x_i}) \qquad (1)$$

where $p(\mathbf{x})$ is either a multivariate Gaussian density function with zero mean and identity covariance represented by $\mathcal{N}(\mathbf{x}\colon 0, \mathbf{I})$ or uniform density function with support over hypercube with opposite corners at $[-1, -1, \ldots, -1]$ and $[1, 1, \ldots, 1]$. Assuming that $f(\mathbf{x})$ has a valid Taylor series expansion about the mean $\mathbf{0}$, (1) it can be rewritten as

$$E[f(\mathbf{x})] \approx \sum_{N_1=0}^{\infty} \sum_{N_2=0}^{\infty} \cdots \sum_{N_n=0}^{\infty} \frac{E\left[x_1^{N_1} x_2^{N_2} \cdots x_n^{N_n}\right]}{N_1! N_2! \cdots N_n!} \frac{\partial^{N_1+N_2+\cdots+N_n} f}{\partial x_1^{N_1} \partial x_2^{N_2} \cdots \partial x_n^{N_n}}(\mathbf{0})$$

$$(2)$$

Notice that the problem of evaluating the expected value of a nonlinear function $f(\mathbf{x})$ has reduced to computing higher-order moments of the PDF. Thus, by increasing the number of terms in the Taylor series expansion, one can obtain an accurate value of the expectation integral. Furthermore, consider the discrete approximation of Eq. (1) as a weighted average of $f(\mathbf{x})$ evaluated at N cubature points, $\mathbf{x}_i$ with corresponding weights w_i, where each $\mathbf{x_i}$ is a n-$\mathbb{D}$ vector with $x_{(i,j)}$ representing the jth coordinate of the ith point. Now, substituting the Taylor series expansion of $f(\mathbf{x_i})$ in Eq. (1) leads to:

$$\sum_{i=1}^{N} w_i f(\mathbf{x_i}) \approx \sum_{N_1=0}^{\infty} \sum_{N_2=0}^{\infty} \cdots \sum_{N_n=0}^{\infty} \frac{\left(\sum_{i=1}^{N} w_i \{ x_{(i,1)}^{N_1} x_{(i,2)}^{N_2} \cdots x_{(i,n)}^{N_n} \} \right)}{N_1! N_2! \cdots N_n!}$$

$$\times \frac{\partial^{N_1+N_2+\cdots+N_n} f}{\partial x_1^{N_1} \partial x_2^{N_2} \cdots \partial x_n^{N_n}}(\mathbf{0})$$

$$(3)$$

Comparing the coefficients of $f(\mathbf{0})$ and derivatives of $f(\mathbf{x})$ evaluated at $\mathbf{0}$ from Eqs. (2) and (3) leads to the following set of equations known as the moment constraint equations

$$\sum_{i=1}^{N} w_i \{ x_{(i,1)}{}^{N_1} x_{(i,2)}{}^{N_2} \cdots x_{(i,n)}{}^{N_n} \} = E[x_1^{N_1} x_2^{N_2} \cdots x_n^{N_n}] \quad (4)$$

where $\{N_1 + N_2 \cdots + N_n = d\}$ represents the order/degree of the moment of the PDF. Alternatively, the MCE in Eq. (4) can be determined by stating that the sigma point set $(\mathbf{X}, \mathbf{w})$ can integrate all monomials (hence polynomials) of degree up to and including d. This is known as the moment matching method.

Notice that these constraint equations simply convey that the sigma point set $(\mathbf{X}, \mathbf{w})$ should ideally capture all the infinite moments of the PDF. Often, this is not required and one seeks to find a sigma point set $(\mathbf{X}, \mathbf{w})$ that can reproduce the first d moments of the PDF exactly to guarantee the exact evaluation of the expectation integral of Eq. (1) when $f(\mathbf{x})$ is a polynomial function of total degree d or less. On the other hand, when integrating nonpolynomial functions, one usually does not know the number of moments to be satisfied but one can expect that higher-order cubature points would have lower error than the lower-order cubature points. These moment constraint equations are high degree multivariate polynomial equations whose solution is not trivial and in some cases even intractable. The Gauss quadrature product rule in n-$\mathbb{D}$ indeed satisfies these moment constraint equations theoretically to any prescribed order d by taking tensor product of $1\mathbb{D}$ quadrature points. Hence, Gauss quadrature methods suffer from the *curse of dimensionality* with the number of points increasing exponentially with dimension. Nevertheless, it is interesting to observe that Gaussian quadratures are symmetric about the mean and thereby inherently satisfy all moments involving *any odd exponent*, i.e., N_i is odd.

As an alternative to the Gauss quadrature scheme, the nonproduct cubature methods solve these moment constraint equations by judiciously selecting quadrature points also known as *sigma points* in $\mathbb{R}^n$ and offer similar numerical accuracy with fewer points. However, the development of a cubature rule of any given degree d that is applicable to any dimension is still an open problem. The Smolyak quadrature scheme and UT are among the most widely used nonproduct quadrature schemes. The Smolyak quadrature scheme is a general cubature rule of any given degree d and involves sparse tensor product of $1\mathbb{D}$ Gauss quadrature rule. However, it can result in quadrature points with negative weights [10]. The UT is only a third-degree quadrature rule and involves the judicious selection of sigma set by constraining the points to lie on prescribed directions in $\mathbb{R}^n$. The main focus of this work is to extend the UT rules to higher-order quadrature rules by selecting additional axes or directions in $\mathbb{R}^n$. Due to the symmetry of the Gaussian and Uniform probability density functions, the odd moments are always zero and the remaining even order moments are tabulated in Tables 1 and 2. These even order moments can be simply evaluated by taking the product of $1\mathbb{D}$ moments.

Any permutation of the form $\{i, j, k, l\} \in \{1, 2, 3, \ldots, N\}$ in Tables 1 and 2 has the same set of moments as the PDFs are symmetric in all dimensions.

3 Unscented Transformation

In this section, a brief review of the conventional UT approach [11] is presented, which is developed to compute first two

moments of a Gaussian PDF. The UT algorithm involves the selection of $2n + 1$ symmetrically placed sigma points about the mean such that the first two moment constraint equations ($d = 0$, 2) are satisfied. The first ($d = 1$) and third ($d = 3$) moments being odd are already satisfied by the symmetry of the chosen points. For an arbitrary covariance matrix $\mathbf{P}$ and mean vector $\boldsymbol{\mu}$, the suggested sigma point set $(\mathbf{X}, \mathbf{w})$ is generated as:

$$\mathbf{x}_0 = \boldsymbol{\mu} \qquad \mathbf{w}_0 = \kappa / (n + \kappa) \quad (5)$$

$$\mathbf{x}_i = \boldsymbol{\mu} + (\sqrt{(n + \kappa)\mathbf{P}})_i, \qquad \mathbf{w}_i = 1/[2(n + \kappa)] \quad (6)$$

$$\mathbf{x}_{i+n} = \boldsymbol{\mu} - (\sqrt{(n + \kappa)\mathbf{P}})_i, \qquad \mathbf{w}_{i+n} = 1/[2(n + \kappa)] \quad (7)$$

These points are selected in such a way that one of the point, $\mathbf{x}_0$, lies at the mean of the Gaussian density function and the remaining $2n$ points, $\mathbf{x}_i$, are constrained to lie symmetrically at a distance $\sqrt{n + \kappa}$ about the mean on n-symmetrical axes defined by columns of $\sqrt{\mathbf{P}}_i$. When $\mathbf{P} = I$ (identity matrix), these axes are simply the n-orthogonal axes that are generally known as "principal axes." Figure 1(*a*) shows the sigma points and the corresponding length of the stems represent the weights for each point while Fig. 1(*b*) shows the role of κ that adjusts the distance of the points from the mean and their corresponding weights. Note that the size of the circle represents the weight associated with the sigma point, i.e., as κ increases, the sigma points move farther away from the mean and the associated weights decrease.

Furthermore, the tuning parameter κ is generally chosen such that one of the fourth moment constraint equation error can be minimized, which leads to the constraint of $n + \kappa = 3$. This method works well for evaluation of integrals with dimensions 3 or less, after which $\kappa < 0$ and the weight corresponding to the central point becomes negative. It is known that cubature rules with negative weight have large errors [8]. Hence, it is always desirable to have a cubature rule with all positive weights.

Tenne and Singh [25] have developed higher-order unscented transformation methods for a general PDF in $1\mathbb{D}$, which can satisfy the moment equations up to eighth-order. Lévesque generalizes this to develop unscented transformation for generic n-$\mathbb{D}$ integral [26]. However, this method fails to capture cross moments. Next, it is shown that no cross moment of any order can be satisfied by the selection of symmetric points constrained to lie only on the principle axes.

3.1 Unscented Transform and Higher-Order Moment Constraint Equations. Let us consider the problem of selecting sigma points such that the first four moment equations are satisfied. Assuming i.i.d Gaussian random variables, $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$ with zero mean and identity covariance, the moments up to fourth-order are given as

$$\begin{aligned} E[x_i^2] &= 1, & E[x_i x_j] &= 0, & E[x_i^4] &= 3 \\ E[x_i^3 x_j] &= 0, & E[x_i^2 x_j^2] &= 1, & E[x_i^2 x_j x_k] &= 0 \\ E[x_i x_j x_k x_l] &= 0, \end{aligned} \quad (8)$$

Table 1 Even order moments of the Gaussian PDF

Moment	Value	Moment	Value	Moment	Value
$E[x_i^2]$	1	$E[x_i^4]$	3	$E[x_i^2 x_j^2]$	1
$E[x_i^6]$	15	$E[x_i^4 x_j^2]$	3	$E[x_i^2 x_j^2 x_k^2]$	1
$E[x_i^8]$	105	$E[x_i^6 x_j^2]$	15	$E[x_i^4 x_j^4]$	9
$E[x_i^4 x_j^2 x_k^2]$	3	$E[x_i^2 x_j^2 x_k^2 x_l^2]$	1		

Table 2 Even order moments of the uniform PDF

Moment	Value	Moment	Value	Moment	Value
$E[x_i^2]$	$\frac{1}{3}$	$E[x_i^4]$	$\frac{1}{5}$	$E[x_i^2 x_j^2]$	$\frac{1}{9}$
$E[x_i^6]$	$\frac{1}{7}$	$E[x_i^4 x_j^2]$	$\frac{1}{15}$	$E[x_i^2 x_j^2 x_k^2]$	$\frac{1}{27}$
$E[x_i^8]$	$\frac{1}{9}$	$E[x_i^6 x_j^2]$	$\frac{1}{21}$	$E[x_i^4 x_j^4]$	$\frac{1}{25}$
$E[x_i^4 x_j^2 x_k^2]$	$\frac{1}{45}$	$E[x_i^2 x_j^2 x_k^2 x_l^2]$	$\frac{1}{81}$		

where $\{i, j, k, l\} \subset \{1, 2, 3, \ldots, n\}$. As in the conventional UT scheme, a fully symmetric set of sigma points are chosen such that they lie on the orthogonal Cartesian axes at a distance r_1 from the origin and each have equal weight of w_1. The central point on the origin (mean) has weight w_0. When enumerated, the set of points are $\{(r_1, 0, \ldots, 0), (0, r_1, \ldots, 0), (0, 0, \ldots, r_1)\ (-r_1, 0, \ldots, 0), (0, -r_1, \ldots, 0), \ldots, (0, 0, \ldots, -r_1),\ (0, 0, \ldots, 0)\}$ and the corresponding MCEs are given as

$$E[x_i^0] \equiv w_0 + 2nw_1 = 1 \tag{9}$$

$$E[x_i^2] \equiv 2r_1^2 w_1 = 1 \tag{10}$$

$$E[x_i^4] \equiv 2r_1^4 w_1 = 3 \tag{11}$$

$$E[x_i^2 x_j^2] \equiv 0 \neq 1 \tag{12}$$

It can be seen that the choice of $r_1 = \sqrt{n + \kappa}$ in Eq. (10) leads to the expression for weights in Eqs. (6) and (7). It should be noticed that moments of Eq. (8) containing odd exponents are already satisfied due to the symmetry of $2n + 1$ sigma points. Furthermore, the fourth-order moment in Eq. (11) leads to $n + \kappa = 3$. The cross moment $E[x_i^2 x_j^2]$ in Eq. (12) cannot be satisfied by this particular selection of symmetric sigma points that are constrained to lie only on the principal axes.

In fact, no cross moment of any order can be satisfied by the selection of symmetric points constrained to lie only on the orthogonal Cartesian axes. Hence, by merely adding more points or scaling points, they can only minimize the error in the higher-order moments but cannot capture even order cross moments. For a symmetric PDF such as a Gaussian or uniform PDF, moments such as $E[x_1^{N_1} x_2^{N_2} \cdots x_n^{N_n}]$ are nonzero when $N_1, N_2, \ldots, N_n$ are all even numbers. But points on the orthogonal Cartesian axes, which are of the form $(\pm r, 0, 0, \ldots, 0)$, always produce a zero when substituted into the left-hand side of the MCE for all cross moments.

Also, notice that the CKF is a special case of UT when $\kappa = 0$. Hence, the UT and CKF have discrepancy in the fourth-order moments. In summary, the set of nonlinear moment constraint equations is tedious to solve for a general n-$\mathbb{D}$ system. Either one needs to break the symmetry of the sigma points or one needs to look for alternative axes to define new sigma points. Breaking the symmetry of sigma points is not desirable since this will require

one to include moment equations containing odd exponents in the formulation. In this work, the second option is pursued by including additional axes, which are labeled as *conjugate axes* such that the resultant points are more "spread out" symmetrically in space.

4 Conjugate Unscented Transformation

In this section, the conjugate unscented transform is introduced as an extension of the conventional UT to define new sigma points, which can capture higher-order moments. The CUT points have been developed for the Gaussian PDF and Uniform PDF. The points are labeled as CUT-G for the Gaussian PDF and CUT-U for the Uniform PDF. As discussed in Sec. 3, the UT provides the insight that "the points can be constrained to lie on some *carefully* selected axes" and one only needs to solve for the distances of these points from the origin and their corresponding weights. The set of moment constraint equations along with these additional constraints can make the system tractable for higher-orders and dimensions. However, choosing appropriate axes is still not a *trivial* problem. Hence, as a first step, the following important axes are defined to select higher-order sigma points.

In n-$\mathbb{D}$ Cartesian space, the *principal axes* are defined as the n orthogonal axes centered at the origin. The points on the principal axes are enumerated as

$$\sigma_i \in \{\pm (I)_k | k \in \mathcal{D}\}, \quad i = 1, 2, 3, \ldots, 2n \tag{13}$$

where $\mathcal{D} = \{1, 2, 3, \ldots, n\}$ and $(I)_k$ represents the kth row or column of the $n \times n$ dimensional identity matrix I. Each point on the principle axes is at a unit distance from the origin.

In n-$\mathbb{D}$ Cartesian space, the mth *conjugate axes* with $m \leq n$ are defined as the directions that are constructed from all the combinations of principal axes, including the sign permutations, with m-principal axes taken at a time. The set of mth conjugate axes labeled as c^m, where the points are listed as c_i^m, is written as:

$$c_i^m \in \{(\pm \sigma_{N_1} \pm \sigma_{N_2} \cdots \pm \sigma_{N_m}) | \{N_1, N_2, \ldots, N_m\} \subset \mathcal{D}\}$$
$$i = 1, 2, \ldots, 2^m \binom{n}{m}$$

In n-$\mathbb{D}$ Cartesian space, the mth *scaled conjugate axes* is defined as the set of directions constructed from all combinations,
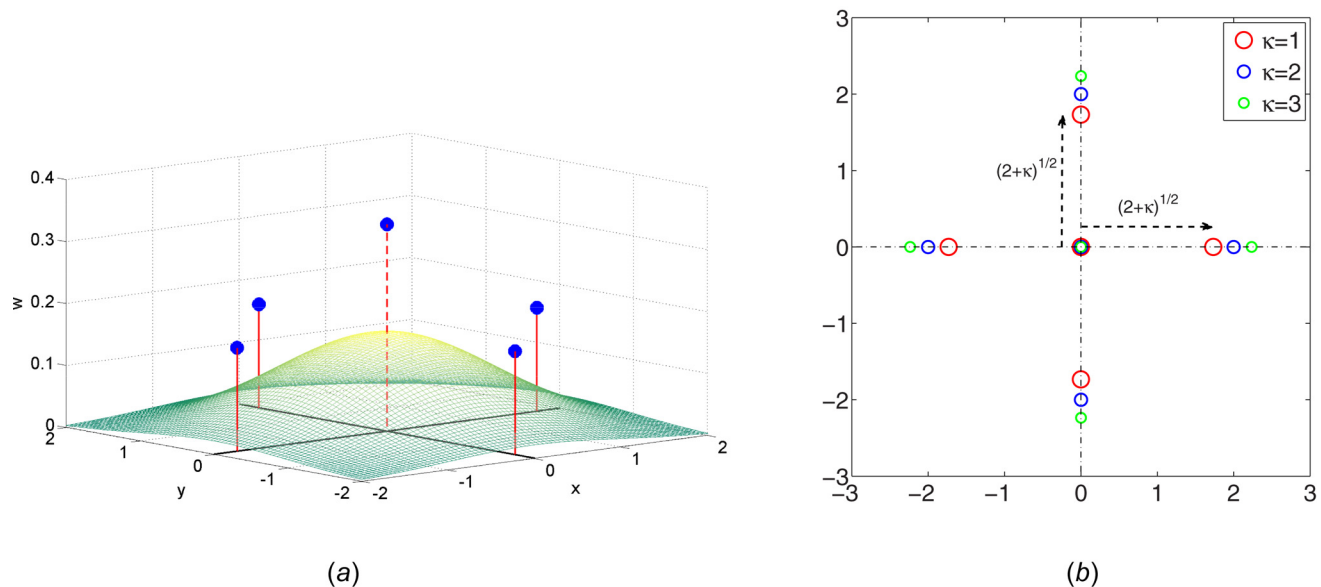

Fig. 1 Unscented transformation points in 2$\mathbb{D}$: (*a*) points with corresponding weights and (*b*) effect of κ on weight

including sign permutations, of the set of principal axes such that in every combination, exactly one principal axis is scaled by a parameter $h \in \mathbb{R}$. A set of mth scaled conjugate axes are labeled as s^m, and the points are listed as s_i^m

$$s_i^m \in \{(\pm h\sigma_{N_1} \pm \sigma_{N_2} \cdots \pm \sigma_{N_m}) | \{N_1, N_2, ..., N_m\} \subset \mathcal{D}\}$$

$$i = 1, 2, ..., n2^m \binom{n}{m}$$

The various axes for the 2$\mathbb{D}$ case are shown in Fig. 2(a) while Figs. 2(b) and 2(c) show two different perspective views of the first octant for a 3$\mathbb{D}$ case. It should be mentioned that all the eight octants in the 3$\mathbb{D}$ case are symmetrical.

The next step involves constraining the sigma points to lie on these special axes so that they satisfy higher-order MCE:

(1) As the Gaussian and uniform PDF are symmetric, choosing points on the symmetric axes inherently satisfies all the moment constraint equations involving odd exponents. If the set $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x_n}\}$ is a *fully symmetric* set [8], then it is closed under the operations of coordinate position and sign permutations. For example, in 2$\mathbb{D}$ if $\mathbf{x_i} = [a, b]^T \in \mathbf{X}$, then $\{[b,a]^T, [-a,b]^T, [a,-b]^T, [-a,-b]^T, [-b,a]^T, [b,-a]^T, [-b,-a]^T\} \in \mathbf{X}$. The various structures of symmetric sigma point sets used to develop the CUT method are summarized in Table 3. Only one sample point is shown for each type of axes. Given a single point, the fully symmetric set for each type can be easily completed by taking all possible permutations of coordinate position and sign.

An example for a 3$\mathbb{D}$ system is provided for which the symmetric points in Table 3 are explicitly illustrated in the below equation:

$$c_i^2 \in \left\{ \begin{matrix} (1,1,0) & (1,-1,0) \\ (1,0,1) & (1,0,-1) \\ (0,1,1) & (0,1,-1) \\ (-1,1,0) & (-1,-1,0) \\ (-1,0,1) & (-1,0,-1) \\ (0,-1,1) & (0,-1,-1) \end{matrix} \right\}, \; c_i^3 \in \left\{ \begin{matrix} (1,1,1) \\ (-1,1,1) \\ (1,-1,1) \\ (1,1,-1) \\ (-1,-1,1) \\ (1,-1,-1) \\ (-1,1,-1) \\ (-1,-1,-1) \end{matrix} \right\},$$

$$\sigma_i \in \left\{ \begin{matrix} (1,0,0) \\ (0,1,0) \\ (0,0,1) \\ (-1,0,0) \\ (0,-1,0) \\ (0,0,-1) \end{matrix} \right\}, s_i^3 \in \left\{ \begin{matrix} (h,1,1) & (1,h,1) & (1,1,h) \\ (-h,1,1) & (-1,h,1) & (-1,1,h) \\ (h,-1,1) & (1,-h,1) & (1,-1,h) \\ (h,1,-1) & (1,h,-1) & (1,1,-h) \\ (-h,-1,1) & (-1,-h,1) & (-1,-1,h) \\ (h,-1,-1) & (1,-h,-1) & (1,-1,-h) \\ (-h,1,-1) & (-1,h,-1) & (-1,1,-h) \\ (-h,-1,-1) & (1,-h,-1) & (1,-1,-h) \end{matrix} \right\}$$

(14)

Points on the same set of symmetric axes are equidistant from the mean and have equal weights. For the ith set of axes, the distance scaling variables are labeled as r_i and weight variables are labeled as w_i.

(2) The moment constraint equations up to the desired order are derived in terms of the unknown variables r_i and w_i by making use of the moments of a Gaussian and Uniform PDF. Each sigma point set in Table 3 can be scaled by a variable r_i, assigned a weight w_i and substituted into the left hand side of Eq. (4) resulting in a polynomial equation. Hence, each moment constraint equation would contain monomials with variables r_i and w_i corresponding to each sigma point set. The coefficients of these monomials are

tabulated in Table 4 and they can be used directly in generating the moment constraint equations. The triplet $[a, b, c]$ represents the monomial $ar_i^b w_i^c$ for the ith set of axes, where r_i and w_i are the unknown variables. Here, n is the dimension of the integral and $\binom{n}{m}$ is the binomial coefficient.

(3) This nonlinear set of equations is solved for r_i and w_i, yielding the required sigma point set. Usually, for lower-order moment constraint equations, the polynomial equations can be solved analytically or by the help of symbolic computation software. For higher-order moment constraint equations, efficient polynomial solvers such as BERTINI [27] can be used.

In Secs. 4.1–4.3, these new sigma point sets are explicitly derived by making use of these *conjugate axes* to satisfy the moment constraint equations up to order eight. The CUT points are generated for a Gaussian PDF with zero mean and identity covariance matrix. If $\mathbf{x}_i$ represent the sigma points corresponding to $\mathcal{N}(\mathbf{x}: 0, \mathbf{I})$ and $\mathbf{y}_i$ are the sigma points corresponding to the generic Gaussian PDF, $\mathcal{N}(\mathbf{y}: \boldsymbol{\mu}, \mathbf{P})$. The points $\mathbf{x}_i$ can be transformed to the points $\mathbf{y}_i$ by an affine transformation:

$$\mathbf{y}_i = \Sigma \mathbf{x}_i + \boldsymbol{\mu} \tag{15}$$

where $\Sigma = \sqrt{\mathbf{P}}$. Similarly, for the uniform PDF, let the points generated for a hyper-cube $\mathcal{H}$ with corners at $[-1, -1, ..., -1]$ and $[1, 1, ..., 1]$ be denoted as $\mathbf{x}_i$. The points $\mathbf{y}_i$ for any other hyper-rectangle, given by the lower corner $[a_1, a_2, ..., a_n]$ and upper corner as $[b_1, b_2, ..., b_n]$, can be easily generated by considering an affine transformation of the points:

$$\mathbf{y}_{i,j} = \frac{b_j - a_j}{2} \mathbf{x}_{i,j} + \frac{a_j + b_j}{2} \tag{16}$$

where $\mathbf{y}_{i,j}$ and $\mathbf{x}_{i,j}$ are the jth coordinates of the ith point $\mathbf{y}_i$ and $\mathbf{x}_i$, respectively. In Sec. 4.1, certain minimal cubature points are developed for specific order and dimension. Starting with the lowest-order, the axes are judiciously and progressively added to capture the higher-order moments. The method is illustrated primarily for the Gaussian PDF, i.e., for the CUT-G points. As the essential procedure remains the same, only a brief extension is provided for the development of the CUT-U points. The numerical solutions to both CUT-G and CUT-U points are provided in the Appendix.

4.1 Fourth-Order Conjugate Unscented Transformation (CUT4). The unscented transformation provides the minimal number of second degree equivalent quadrature points for the Gaussian PDF. In Ref. [8], a general second-order rule for any symmetric PDF is provided, which includes UT as the special case. In this section, we derive conjugate unscented transformation sigma points that are fourth moment equivalent, represented by CUT4 and can exactly integrate all polynomials with total degree 5 or less. In this section, we first develop the CUT4 points for the Gaussian PDF and then for the uniform PDF.

4.1.1 Gaussian Probability Density Function. In case of a Gaussian PDF, the moment constraint equations up to order 4 are

$$E[x_i^2] = 1, \qquad E[x_i^4] = 3 \qquad E[x_i^2 x_j^2] = 1 \tag{17}$$

As the scaling variable r and weight w appear in pairs, the equations in Eq. (17) can be solved by introducing two set of axes or equivalently five variables r_1, r_2, w_1, w_2 and the central weight w_0. Note that the sum of the weights should be equal to 1, so the central weight, which does not show up in any MCE, is used to sum all the weights to 1. This results in an under-determined problem and an optimization problem can be framed or one can arbitrarily set w_0 to 0 and solve for the remaining parameters. Since the
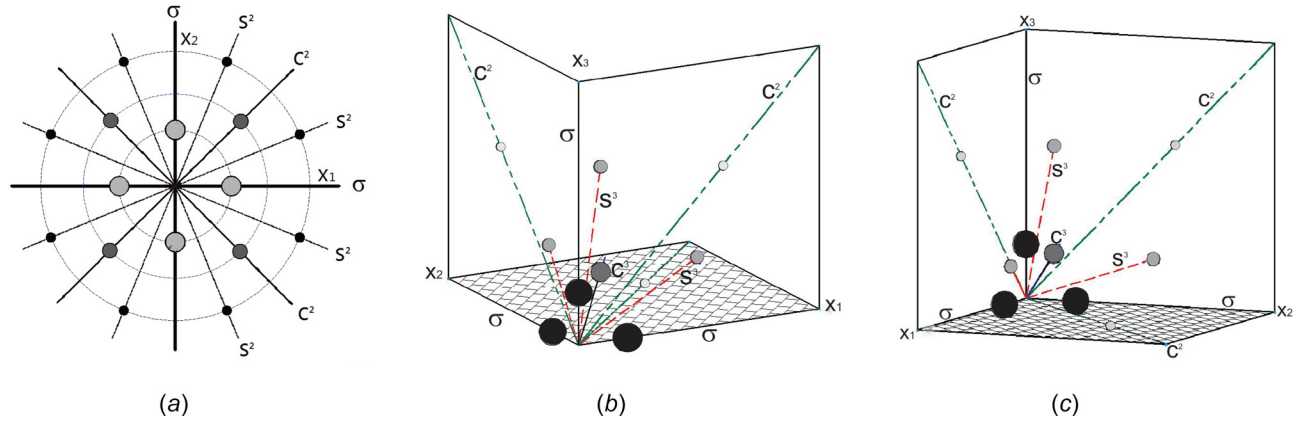
Fig. 2 Symmetric set of points and axes 2D and 3D space: (*a*) 2D axes, (*b*) 3D first quadrant view 1, and (*c*) 3D first quadrant view 2

Table 3 Fully symmetric set of points

Type	Sample point	No. of points
σ	$(1, 0, 0, ..., 0)$	$2n$
c^m	$(\underbrace{1, 1, ..., 1}_{m}, \underbrace{0, 0, ..., 0}_{(n-m)})$	$2^m \binom{n}{m}$
$s^n(h)$	$(h, 1, 1, ..., 1)$	$n2^m \binom{n}{m}$

conjugate axes provide a nonunique selection of points to solve the moment constraint equations (17), potential benefits of selection of specific axes need to be analyzed.

Table 5 illustrates the growth in the number of points as a function of dimension for various conjugate axes. The number with an underscore refers to the principal and second conjugate axes, which were used by Julier and Uhlmann [28]. Note that for $n = 3$, selection of the principal axis in conjunction with the third conjugate axis results in the smallest number of points necessary to satisfy the required constraints. Julier and Uhlmann have described an approach in the Appendix of Ref. [28] to calculate the sigma points that can capture all the fourth-order moments. But this method suffers from the presence of a negative/zero weight for

dimensions higher than or equal to 4. It is to be noted that this cubature rule with the similar drawback has been listed in Ref. [8]. To gain more insight, these equations are rederived and solved. Let us consider that one sigma point of weight w_0 lies on the origin, $2n$ points of weight w_1 lie symmetrically on each principal axes at a distance of r_1, and $2n(n-1)$ points of weight w_2 lie symmetrically on the second-conjugate axes (c^2) at a distance scaled by r_2. As the points are symmetric about the origin, the moments involving odd exponents need not be included. The moment constraint equations up to order 4 for this particular selection of points are given as

$$2r_1^2 w_1 + 4(n-1)r_2^2 w_2 = 1 \tag{18}$$

$$2r_1^4 w_1 + 4(n-1)r_2^4 w_2 = 3 \tag{19}$$

$$4r_2^4 w_2 = 1 \tag{20}$$

$$1 - 2nw_1 - 2n(n-1)w_2 = w_0 \tag{21}$$

The central weight merely helps to sum the weights to 1. The variables r_1, w_1, r_2, w_2 can be solved from Eqs. (18)–(20) and then w_0 can be found from Eq. (21). Hence, there are only four main variables r_1, w_1, r_2, w_2, and three equations leading to an

Table 4 Coefficients of the set of fully symmetric points

Moment	σ	c^m	$s^n(h)$
$E\left[x_i^p\right]_{p=1,2,3,4}$	$[2, p, 1]$	$\left[2^m \binom{n-1}{m-1}, p, 1\right]$	$\left[(n-1)2^n + h^2 2^n, p, 1\right]$
$E\left[x_i^2 x_j^2\right]$	$[0, 4, 1]$	$\left[2^m \binom{n-2}{m-2}, 4, 1\right]$	$\left[2^{n+1}h^2 + (n-2)2^n, 4, 1\right]$
$E\left[x_i^4 x_j^2\right]$	$[0, 6, 1]$	$\left[2^m \binom{n-2}{m-2}, 6, 1\right]$	$\left[2^n h^2 + 2^n h^4 + (n-2)2^n, 6, 1\right]$
$E\left[x_i^4 x_j^4\right]$	$[0, 8, 1]$	$\left[2^m \binom{n-2}{m-2}, 8, 1\right]$	$\left[2^{n+1}h^4 + (n-2)2^n, 8, 1\right]$
$E\left[x_i^6 x_j^2\right]$	$[0, 8, 1]$	$\left[2^m \binom{n-2}{m-2}, 8, 1\right]$	$\left[2^n h^6 + 2^n h^2 + (n-2)2^n, 8, 1\right]$
$E\left[x_i^2 x_j^2 x_k^2\right]$	$[0, 6, 1]$	$\left[2^m \binom{n-3}{m-3}, 6, 1\right]$	$\left[3(2^n)h^2 + (n-3)2^n, 6, 1\right]$
$E\left[x_i^4 x_j^2 x_k^2\right]$	$[0, 8, 1]$	$\left[2^m \binom{n-3}{m-3}, 8, 1\right]$	$\left[2^n h^4 + 2^{n+1}h^2 + (n-3)2^n, 8, 1\right]$
$E\left[x_i^2 x_j^2 x_k^2 x_l^2\right]$	$[0, 8, 1]$	$\left[2^m \binom{n-4}{m-4}, 8, 1\right]$	$\left[2^{n+2}h^2 + (n-4)2^n, 8, 1\right]$

underdetermined system of equations. Julier and Uhlmann [28] choose to minimize the error in one of the sixth-order moment. Before the optimization is carried out, one can simplify the constraint equations as follows:

$$w_2 = \frac{1}{4r_2^4}, \quad w_1 = \frac{4-n}{2r_1^4} \tag{22}$$

$$r_1^2 r_2^2 = r_1^2(n-1) + r_2^2(4-n) \tag{23}$$

As a consequence of this simplification, there is only one constraint equation (23) to solve in terms of the variables r_1 and r_2. Even though it can be solved by minimizing the sixth moment constraint violation, the weight w_1 in Eq. (22) would always result in a negative weight for dimension greater than 4. One can address this problem of negative weight by choosing a different set of conjugate axes.

From Table 5, it can be seen that c^n axes are the next set of axes with fewer points, which will be used to satisfy the moment constraint equations. The new sigma points are selected such that $2n$ points of weight w_1 lie symmetrically on each principal axis at a distance of r_1 and 2^n points of weight w_2 lie on the nth conjugate axes at a distance of $r_2\sqrt{n}$. Note that the boldface number correspond to the fewest quadrature points required to satisfy all the constraints. Substitution of this particular selection of sigma points as summarized in Table 6 leads to the following fourth-order moment constraint equations

$$2r_1^2 w_1 + 2^n r_2^2 w_2 = 1 \tag{24}$$

$$2r_1^4 w_1 + 2^n r_2^4 w_2 = 3 \tag{25}$$

$$2^n r_2^4 w_2 = 1 \tag{26}$$

$$1 - 2nw_1 - 2^n w_2 = w_0 \tag{27}$$

The weight w_0 corresponds to the central point at the mean and does not contribute to any of the moment constraint equations of order $d \geq 1$. There are five variables and only four equations. One can eliminate the central point by choosing $w_0 = 0$. This selection of w_0 results in an analytical solution for the points

$$r_1 = \sqrt{\frac{n+2}{2}}, \quad r_2 = \sqrt{\frac{n+2}{n-2}} \tag{28}$$

$$w_1 = \frac{1}{r_1^4} = \frac{4}{(n+2)^2}, \quad w_2 = \frac{1}{2^n r_2^4} = \frac{(n-2)^2}{2^n(n+2)^2} \tag{29}$$

Notice that r_2 is undefined for $n \leq 2$ and thereby this solution is valid for $n \geq 3$. Alternatively, one can also find the central weight w_0 by minimizing the error in one of the sixth moment constraint equation namely $\left(2r_1^6 w_1 + 2^n r_2^6 w_2 - 15\right)^2$. The solution for dimension $n = 2$ is shown in Table 7 where the sixth moment

constraint equation error has been minimized. In addition, for $n = 1$, there is no cross order moment (26) and this equation can be replaced by the sixth moment constraint equation leading to the standard Gauss–Hermite quadrature with four points. Note that, according to the CUT4-G, one would need 14 and 1044 sigma points to exactly evaluate the expectation of a polynomial function of degree 4 in three- and ten-dimensional space where as the GH Product rule would require a minimal of 27 and 59,049 points, respectively.

4.1.2 Uniform Probability Density Function. Following a similar procedure, the CUT4-U points for the uniform PDF can also be developed. But generating the points for the uniform PDF involves another challenge: all the points have to be within the support of the uniform PDF, i.e., within $\mathcal{H}$. Hence, rather than developing a general rule for all dimensions, the points and weights have been developed for each specific dimension. As there are three moment constraint equations, two sets of axes have to be chosen. With reference to Table 8, one can begin by choosing the axes with the lowest number of points, until the MCE are satisfied along with the desired characteristics of positive weights. For example, points along the principal axes and nth conjugate axes, c^n, constitute minimal number of points for $n \leq 5$ while points along principal axes and c^4 constitute minimal number of points for $n = 6$. The appropriate choice for axes is highlighted in Table 8 and is also listed in Table 9. The moment constraint equations up to fourth-order with selection of points underlined in Table 9 for dimensions 2–5 are given as follows:

$$2nw_1 + 2^n w_2 = 1 \tag{30}$$

$$2r_1^2 w_1 + 2^n r_2^2 w_2 = \frac{1}{3} \tag{31}$$

$$2r_1^4 w_1 + 2^n r_2^4 w_2 = \frac{1}{5} \tag{32}$$

$$2^n r_2^4 w_2 = \frac{1}{9} \tag{33}$$

These four equations can be solved analytically leading to the following solution:

$$r_1 = \frac{\sqrt{4+5n}}{\sqrt{30}}, \quad w_1 = \frac{40}{(4+5n)^2}, \quad r_2 = \frac{\sqrt{4+5n}}{\sqrt{-12+15n}}, \quad w_2 = \frac{(4-5n)^2}{2^n(4+5n)^2} \tag{34}$$

Unfortunately, after dimension 5, the variable r_1 becomes greater than 1 and hence moves out of the support of the uniform PDF. For higher dimensions, a different set of axes are chosen as shown in Table 9. In all these cases, the axes with fewer points were sequentially chosen to satisfy the moment constraint equations up to order 4 and such that the points are within $\mathcal{H}$ and all have positive weights. The MCE can be easily derived and their analytical solutions are given in Ref. [24]. The final solution of the scaling variables and weights is given in Table 29 of the Appendix.

Table 5 Fully symmetric set of points for CUT4-G

Dimension	σ	c^2	c^3	c^4	c^5	c^6	c^7	c^8	c^9	s^n
2	**4**	4	—	—	—	—	—	—	—	8
3	**6**	12	8	—	—	—	—	—	—	24
4	**8**	24	32	16	—	—	—	—	—	64
5	**10**	40	80	80	32	—	—	—	—	160
6	**12**	60	160	240	192	64	—	—	—	384
7	**14**	84	280	560	672	448	128	—	—	896
8	**16**	112	448	1120	1792	1792	1024	256	—	2048
9	**18**	144	672	2016	4032	5376	4608	2304	512	4608

Table 6 Sigma points for CUT4-G

	Position	Weights
$1 \leq i \leq 2n$	$\mathbf{x}_i = r_1 \sigma_i$	$\mathbf{w}_i = w_1$
$1 \leq i \leq 2^n$	$\mathbf{x}_{i+2n} = r_2 c_i^n$	$\mathbf{w}_{i+2n} = w_2$
Central weight	$\mathbf{x}_0 = \mathbf{0}$	$\mathbf{w}_0 = w_0$
	$N = 2n + 2^n \ (+1 \text{ when } w_0 \neq 0)$	

4.2 Sixth-Order Conjugate Unscented Transformation: CUT6. This section describes the CUT procedure to solve all the moment constraint equations up to sixth-order for both Gaussian and uniform density functions.

4.2.1 Gaussian Probability Density Function. The even order moments of a Gaussian PDF, up to order 6, that need to be satisfied are given as follows:

$$E[x_i^2] = 1, \qquad E[x_i^4] = 3, \qquad E[x_i^2 x_j^2] = 1$$
$$E[x_i^6] = 15 \qquad E[x_i^4 x_j^2] = 3 \qquad E[x_i^2 x_j^2 x_k^2] = 1 \tag{35}$$

These equations can be solved by introducing at least six variables r_1, r_2, r_3, w_1, w_2, w_3 or three set of axes to make the system of equation consistent. Table 10 lists the points along different axes with underlined points used by CUT6-G. The axes were chosen sequentially in order to keep the total number of points to a minimum. The sigma points on the principal axes have been assigned a weight of w_1 and are constrained to lie symmetrically at a distance r_1 from the origin. Points on the nth conjugate axes have been chosen with a weight w_2 and are constrained to lie symmetrically at a distance scaled by r_2. Finally, the third set of points are selected with weight w_3 and are constrained to lie symmetrically at a distance scaled by r_3 along the second-conjugate axes. These points are enumerated in Table 11. The set of moment constraint equations using points in Table 11 are given as

$$2r_1^2 w_1 + 2^n r_2^2 w_2 + 4(n-1)r_3^2 w_3 = 1 \tag{36}$$

$$2r_1^4 w_1 + 2^n r_2^4 w_2 + 4(n-1)r_3^4 w_3 = 3 \tag{37}$$

$$2^n r_2^4 w_2 + 4r_3^4 w_3 = 1 \tag{38}$$

$$2r_1^6 w_1 + 2^n r_2^6 w_2 + 4(n-1)r_3^6 w_3 = 15 \tag{39}$$

$$2^n r_2^6 w_2 + 4r_3^6 w_3 = 3 \tag{40}$$

$$2^n r_2^6 w_2 = 1 \tag{41}$$

$$1 - 2nw_1 - 2^n w_2 - 2n(n-1)w_3 = w_0 \tag{42}$$

Solving Eqs. (39)–(41) for unknown weights leads to following analytical expressions:

$$w_1 = \frac{8-n}{r_1^6}, \qquad w_2 = \frac{1}{2^n r_2^6}, \qquad w_3 = \frac{1}{2r_3^6} \tag{43}$$

The substitution of the analytical expression for weight variables from Eq. (43) in Eqs. (36)–(38) leads to following three polynomial equations in terms of r_1, r_2, and r_3:

Table 7 CUT4-G: optimized solution for $n=1$ and $n=2$

Variable	$n=1$	$n=2$
r_1	0.7419637843027252	2.6060099476935847
r_2	2.3344142183389796	1.190556300661233
w_0	0	0.41553535186548973
w_1	0.45412414523193145	0.021681819434216532
w_2	0.04587585476806854	0.12443434259941118
No. of points	4	9

Table 8 Fully symmetric set of points for CUT4-U

Dimension	σ	c^2	c^3	c^4	c^5	c^6	c^7	c^8	c^9	s^n
2	**4**	4	—	—	—	—	—	—	—	8
3	**6**	12	8	—	—	—	—	—	—	24
4	**8**	24	32	**16**	—	—	—	—	—	64
5	**10**	40	80	80	**32**	—	—	—	—	160
6	**12**	60	160	**240**	192	64	—	—	—	384
7	**14**	84	280	560	**672**	448	128	—	—	896
8	**16**	112	448	1120	**1792**	1792	1024	256	—	2048

$$-2r_1^4 r_2^4 + 2nr_1^4 r_2^2 + r_1^4 r_3^4 + 16r_2^4 r_3^4 - 2nr_2^4 r_3^4 - r_1^4 r_2^4 r_3^4 = 0$$
$$-2r_1^2 r_2^2 + 2nr_1^2 r_2^2 + r_1^2 r_3^2 + 16r_2^2 r_3^2 - 2nr_2^2 r_3^2 - 3r_1^2 r_2^2 r_3^2 = 0 \tag{44}$$
$$2r_2^2 + r_3^2 - r_2^2 r_3^2 = 0$$

Further by defining $r_1 = (1/\sqrt{a_1})$, $r_2 = (1/\sqrt{a_2})$ and $r_3 = (1/\sqrt{a_3})$, the aforementioned system of three equation in Eq. (44) can be rewritten as

$$2(8-n)a_1^2 + a_2^2 + 2a_3^2(n-1) = 1$$
$$2(8-n)a_1 + a_2 + 2a_3(n-1) = 3 \tag{45}$$
$$a_2 + 2a_3 = 1$$

This reduced system of equations in Eq. (45) is simpler to solve than the original system of equations given by Eqs. (36)–(42). From Eq. (43), it is evident that $w_1 < 0$ for $n > 8$. Furthermore, the central weight becomes negative for $n > 6$. Hence, the solution to Eq. (45) providing quadrature points with positive weights is valid for $n \leq 6$. These points are listed in Table 27 of the Appendix.

Furthermore, the problem of negative weight can be avoided by choosing third-conjugate axes instead of second-conjugate axes, which makes the proposed approach valid for $7 \leq n \leq 9$. The set of sixth-order moment constraint equations in terms of these new sigma points, as enumerated in Table 12, are given as

$$2r_1^2 w_1 + 2^n r_2^2 w_2 + 4(n-1)(n-2)r_3^2 w_3 = 1 \tag{46}$$

$$2r_1^4 w_1 + 2^n r_2^4 w_2 + 4(n-1)(n-2)r_3^4 w_3 = 3 \tag{47}$$

$$2^n r_2^4 w_2 + 8(n-2)r_3^4 w_3 = 1 \tag{48}$$

$$2r_1^6 w_1 + 2^n r_2^6 w_2 + 4(n-1)(n-2)r_3^6 w_3 = 15 \tag{49}$$

$$2^n r_2^6 w_2 + 8(n-2)r_3^6 w_3 = 3 \tag{50}$$

$$2^n r_2^6 w_2 + 8r_3^6 w_3 = 1 \tag{51}$$

$$1 - 2nw_1 - 2^n w_2 - \left(4n(n-1)(n-2)/3\right)w_3 = w_0 \tag{52}$$

Table 9 CUT4-U rules

n	Points	Weights
2, 3, 4, 5	$r_1 \sigma_i$	w_1
	$r_2 c_i^n$	w_2
6	$r_1 \sigma_i$	w_1
	$r_2 c_i^4$	w_2
7, 8	$r_1 \sigma_i$	w_1
	$r_2 c_i^5$	w_2

Once again exploiting the linearity in the weights, one can first solve for weights analytically from Eqs. (49) to (51)

$$w_1 = -\frac{-24r_2^2 + 3(11-n)r_3^2 + (n-3)r_2^2 r_3^2}{4r_1^6(r_2^2 - r_3^2)},$$

$$w_2 = -\frac{2^{-n}(r_3^2 - 3)}{r_2^4(r_2^2 - r_3^2)}, \quad w_3 = -\frac{3 - r_2^2}{8(n-2)r_3^4(r_2^2 - r_3^2)}$$

The solution for r_1, r_2, and r_3 can be computed from the remaining three equations (46)–(48). Notice that this aforementioned set of Eqs. (46)–(51) is preferred till dimension 9 as the central weight becomes negative for $n > 9$. But if one allows the presence of negative weight, the equations can be solved up to dimension $n = 13$ after which real roots cannot be guaranteed. The results of evaluating the expectation integral by the CUT6-G method show a contrasting difference in the number of points required. For example, the CUT6-G would need only 49 and 1203 function evaluations while the GH product rule would require a minimal of 256 and 262,144 function evaluations for the exact computation of the expectation integral of a polynomial function of degree 6 in four- and nine-dimensional space, respectively. The solution to these MCEs (36)–(41) and (46)–(51) are provided in Table 27 of the Appendix. Thereby, along with Tables 11 and 12, one can directly generate the sigma point set of order 6. In Table 10, the numbers indicted by an underscore correspond to the number of CUT6-G points required to exactly integrate all polynomials with total degree 7 or less and correspondingly the specific axes, which are required to ensure positive weights for the CUT points.

4.2.2 Uniform Probability Density Function. Table 13 shows the choice of axes, for the uniform PDF, that would lead to the least number of points within $\mathcal{H}$ which have positive weights. The axes are selected sequentially from Table 14 to satisfy the moment constraint equations with minimal number of points. The underlined number in Table 14 corresponds to points along the specific axes for which solution to MCEs is found with all positive weights. For $n = 2$, there are only five moment constraint equations up to order 6 because one of the sixth-order cross moment $E\left[x_i^2 x_j^2 x_k^2\right]$ does not exist in $2\mathbb{D}$. An example of the MCE, using the choice of axes from Table 13, in the $2\mathbb{D}$ case is

$$2r_1^2 w_1 + 4r_2^2 w_2 + 4r_3^2 w_3 = \frac{1}{3} \tag{53}$$

$$2r_1^4 w_1 + 4r_2^4 w_2 + 4r_3^4 w_3 = \frac{1}{5} \tag{54}$$

$$4r_2^4 w_2 + 4r_3^4 w_3 = \frac{1}{9} \tag{55}$$

$$2r_1^6 w_1 + 4r_2^6 w_2 + 4r_3^6 w_3 = \frac{1}{7} \tag{56}$$

$$4r_2^6 w_2 + 4r_3^6 w_3 = \frac{1}{15} \tag{57}$$

$$w_0 + 4w_1 + 4w_2 + 4w_3 = 1 \tag{58}$$

As there are six equations and seven variables, an optimized solution is sought such that the square of the error in the eighth moment $\left(2r_1^8 w_1 + 4r_2^8 w_2 + 4r_3^8 w_3 - (1/9)\right)^2$ is minimized subject to the constraints in Eqs. (53)–(58). The solution to this optimization problem is given in Table 30 of the Appendix. For $n \geq 3$, there are six moment constraint equations and one equation for the weights to add up to one. One additional set of axes was used to satisfy the positive weight constraint and guarantee that all the points lie within $\mathcal{H}$. When the points are selected as in Table 13, it can be seen that the system is underdetermined with more variables than equations. One can again pose an optimization problem with the objective to minimize the error in higher moment constraint equations. But due to the polynomial nature of the set of MCE, the higher the degree of the equations the higher is the computational complexity. One cannot always guarantee solutions with real or the positive weight or even a solution with all points within the support of the uniform PDF. Hence, appropriate values for some variables are assumed to reduce the computation complexity. The resultant system of equations can easily be solved by numerical polynomial solvers such as BERTINI [27]. The final numerical solutions for the particular selection of points for dimensions $n = 3, 4, 5, 6, 7, 8, 9$ in Table 13 are provided in Table 30 of the Appendix.

4.3 Eighth-Order Conjugate Unscented Transform: CUT8. In this section, sigma points are selected such that they satisfy all the moment constraint equations up to order 8 for both Gaussian and uniform PDFs.

4.3.1 Gaussian Probability Density Function. The 11 nonzero moments up to the eighth-order for a Gaussian PDF with zero mean and identity covariance are

$$\begin{aligned}
&E\left[x_i^2\right] = 1, \qquad E\left[x_i^4\right] = 3, \quad E\left[x_i^2 x_j^2\right] = 1, \\
&E\left[x_i^6\right] = 15, \quad E\left[x_i^4 x_j^2\right] = 3, \quad E\left[x_i^8\right] = 105, \\
&E\left[x_i^6 x_j^2\right] = 15, \quad E\left[x_i^4 x_j^4\right] = 9, \quad E\left[x_i^4 x_j^2 x_k^2\right] = 3, \\
&E\left[x_i^2 x_j^2 x_k^2\right] = 1, \quad E\left[x_i^2 x_j^2 x_k^2 x_l^2\right] = 1
\end{aligned} \tag{59}$$

To solve the system of 11 moment constraint equations, one would require at least 11 variables. As the distance and weight variables appear in pairs, six sets of points including the scaled conjugate axes parameter h would lead to 13 variables. Table 15 shows the axes, which are used for the CUT8-G points as a function of the dimension of the random variables. The specific structure of points chosen to capture these 11 MCEs is shown in Table 16. For instance, this particular selection of points in $5\mathbb{D}$ space leads to the following 11 moment constraint equations:

Table 10 Fully symmetric set of points for CUT6-G

Dimension	σ	c^2	c^3	c^4	c^5	c^6	c^7	c^8	c^9	s^n
2	**4**	**4**	—	—	—	—	—	—	—	8
3	**6**	**12**	**8**	—	—	—	—	—	—	24
4	**8**	**24**	**32**	**16**	—	—	—	—	—	64
5	**10**	**40**	**80**	**80**	**32**	—	—	—	—	160
6	**12**	**60**	**160**	**240**	**192**	**64**	—	—	—	384
7	**14**	84	**280**	**560**	**672**	**448**	**128**	—	—	896
8	**16**	112	**448**	1120	1792	1792	1024	**256**	—	2048
9	**18**	144	**672**	2016	4032	5376	4608	2304	**512**	4608

Table 11 Points for CUT6-G, ($n \leq 6$)

	Position	Weights
$1 \leq i \leq 2n$	$\mathbf{x}_i = r_1 \sigma_i$	$\mathbf{w}_i = w_1$
$1 \leq i \leq 2^n$	$\mathbf{x}_{i+2n} = r_2 c_i^n$	$\mathbf{w}_{i+2n} = w_2$
$1 \leq i \leq 2n(n-1)$	$\mathbf{x}_{i+2n+2^n} = r_3 c_i^2$	$\mathbf{w}_{i+2n+2^n} = w_3$
Central weight	$\mathbf{x}_0 = \mathbf{0}$	$\mathbf{w}_0 = w_0$
	$N = 2n^2 + 2^n + 1$	

$$2r_1^2w_1 + 32r_2^2w_2 + 16r_3^2w_3 + 32r_4^2w_4 + 48r_5^2w_5 + 128r_6^2w_6 + 32h^2r_6^2w_6 = 1$$
$$2r_1^4w_1 + 32r_2^4w_2 + 16r_3^4w_3 + 32r_4^4w_4 + 48r_5^4w_5 + 128r_6^4w_6 + 32h^4r_6^4w_6 = 3$$
$$32r_2^4w_2 + 4r_3^4w_3 + 32r_4^4w_4 + 24r_5^4w_5 + 96r_6^4w_6 + 64h^2r_6^4w_6 = 1$$
$$2r_1^6w_1 + 32r_2^6w_2 + 16r_3^6w_3 + 32r_4^6w_4 + 48r_5^6w_5 + 128r_6^6w_6 + 32h^6r_6^6w_6 = 15$$
$$32r_2^6w_2 + 4r_3^6w_3 + 32r_4^6w_4 + 24r_5^6w_5 + 96r_6^6w_6 + 32h^2r_6^6w_6 + 32h^4r_6^6w_6 = 3$$
$$32r_2^6w_2 + 32r_4^6w_4 + 8r_5^6w_5 + 64r_6^6w_6 + 96h^2r_6^6w_6 = 1 \tag{60}$$
$$2r_1^8w_1 + 32r_2^8w_2 + 16r_3^8w_3 + 32r_4^8w_4 + 48r_5^8w_5 + 128r_6^8w_6 + 32h^8r_6^8w_6 = 105$$
$$32r_2^8w_2 + 4r_3^8w_3 + 32r_4^8w_4 + 24r_5^8w_5 + 96r_6^8w_6 + 32h^2r_6^8w_6 + 32h^6r_6^8w_6 = 15$$
$$32r_2^8w_2 + 4r_3^8w_3 + 32r_4^8w_4 + 24r_5^8w_5 + 96r_6^8w_6 + 64h^4r_6^8w_6 = 9$$
$$32r_2^8w_2 + 32r_4^8w_4 + 8r_5^8w_5 + 64r_6^8w_6 + 64h^2r_6^8w_6 + 32h^4r_6^8w_6 = 3$$
$$32r_2^8w_2 + 32r_4^8w_4 + 32r_6^8w_6 + 128h^2r_6^8w_6 = 1$$

These 11 moment constraint equations can be solved in a similar manner as mentioned in Secs. 4.1 and 4.2. One can analytically solve for six unknown weights from the last six equations of Eq. (60). The substitution of these expressions of weight variables in the first five equations leads to a system of five equations with seven variables r_1, r_2, r_3, r_4, r_5, r_6, and h. One can minimize the error in the tenth moment constraint equation and find a solution for all seven variables. A numerical optimization procedure with high degree polynomial equations as constraints is often tedious and in some cases even intractable. A work around this issue is to assume appropriate values for h and any one of the distance variables r_i that renders the weights positive. Such assumptions make the system of polynomials equations consistent and hence solvable by highly efficient polynomial solvers. We were able to found sigma points with all positive weights for $n \leq 6$ and these points are listed in Table 28 of the Appendix.

It should be noticed that there are only eight moment constraint equations in 2$\mathbb{D}$ and hence the sigma points corresponding to r_5 and w_5 are unnecessary; in effect, there are only eight variables r_1, r_2, r_3, r_4 and w_1, w_2, w_3, w_4. Similarly, there are ten moment constraint equations for 3$\mathbb{D}$ system and hence the sigma points

corresponding to r_5 and w_5 are dropped. Using Tables 16 and 28 of the Appendix, the CUT8-G method would need only 355 and 745 function evaluations to compute the expectation integral for a polynomial function of degree 8 in the 5$\mathbb{D}$ and 6$\mathbb{D}$ space, respectively, whereas the GH product rule would need 3125 and 15,625 function evaluations for the same 5$\mathbb{D}$ and 6$\mathbb{D}$ space, respectively. It is clear that the number of function evaluations required by the CUT methods is significantly less than that required by the GH quadrature rule while having the same order of accuracy when integrating multivariate polynomial functions.

4.3.2 Uniform Probability Density Function. The eighth-order conjugate unscented transform points for the uniform PDF are shown in Table 17. The axes are selected sequentially in a judicious way to satisfy the moment constraint equations, have positive weights, and have all the points within the hypercube $\mathcal{H}$. There are 11 MCEs corresponding to the 11 even order moments as shown in Table 2. There are only eight MCE and10 MCE for 2$\mathbb{D}$ and 3$\mathbb{D}$ cases, respectively. For example, eighth-order moment constraint equations for the uniform PDF in 4$\mathbb{D}$ space are given as follows:

$$2r_1{}^2w_1 + 16r_2{}^2w_2 + 12r_3{}^2w_3 + 24r_4{}^2w_4 + 48r_5{}^2w_5 + 16h^2r_5{}^2w_5 + 16r_6{}^2w_6 = \frac{1}{3}$$
$$2r_1{}^4w_1 + 16r_2{}^4w_2 + 12r_3{}^4w_3 + 24r_4{}^4w_4 + 48r_5{}^4w_5 + 16h^4r_5{}^4w_5 + 16r_6{}^4w_6 = \frac{1}{5}$$
$$16r_2{}^4w_2 + 4r_3{}^4w_3 + 16r_4{}^4w_4 + 32r_5{}^4w_5 + 32h^2r_5{}^4w_5 + 16r_6{}^4w_6 = \frac{1}{9}$$
$$2r_1{}^6w_1 + 16r_2{}^6w_2 + 12r_3{}^6w_3 + 24r_4{}^6w_4 + 48r_5{}^6w_5 + 16h^6r_5{}^6w_5 + 16r_6{}^6w_6 = \frac{1}{7}$$
$$16r_2{}^6w_2 + 4r_3{}^6w_3 + 16r_4{}^6w_4 + 32r_5{}^6w_5 + 16h^2r_5{}^6w_5 + 16h^4r_5{}^6w_5 + 16r_6{}^6w_6 = \frac{1}{15}$$
$$16r_2{}^6w_2 + 8r_4{}^6w_4 + 16r_5{}^6w_5 + 48h^2r_5{}^6w_5 + 16r_6{}^6w_6 = \frac{1}{27} \tag{61}$$
$$2r_1{}^8w_1 + 16r_2{}^8w_2 + 12r_3{}^8w_3 + 24r_4{}^8w_4 + 48r_5{}^8w_5 + 16h^8r_5{}^8w_5 + 16r_6{}^8w_6 = \frac{1}{9}$$
$$16r_2{}^8w_2 + 4r_3{}^8w_3 + 16r_4{}^8w_4 + 32r_5{}^8w_5 + 16h^2r_5{}^8w_5 + 16h^6r_5{}^8w_5 + 16r_6{}^8w_6 = \frac{1}{21}$$
$$16r_2{}^8w_2 + 4r_3{}^8w_3 + 16r_4{}^8w_4 + 32r_5{}^8w_5 + 32h^4r_5{}^8w_5 + 16r_6{}^8w_6 = \frac{1}{25}$$
$$16r_2{}^8w_2 + 8r_4{}^8w_4 + 16r_5{}^8w_5 + 32h^2r_5{}^8w_5 + 16h^4r_5{}^8w_5 + 16r_6{}^8w_6 = \frac{1}{45}$$
$$16r_2{}^8w_2 + 64h^2r_5{}^8w_5 + 16r_6{}^8w_6 = \frac{1}{81}$$

Table 12 Points for CUT6-G ($7 \leq n \leq 9$)

	Position	Weights
$1 \leq i \leq 2n$	$\mathbf{x}_i = r_1\sigma_i$	$\mathbf{w}_i = w_1$
$1 \leq i \leq 2^n$	$\mathbf{x}_{i+2n} = r_2 c_i^n$	$\mathbf{w}_{i+2n} = w_2$
$1 \leq i \leq 4n(n-1)(n-2)/3$	$\mathbf{x}_{i+2n+2^n} = r_3 c_i^3$	$\mathbf{w}_{i+2n+2^n} = w_3$
Central weight	$\mathbf{x}_0 = \mathbf{0}$	$\mathbf{w}_0 = w_0$
	$N = 2n + 2^n + 4n(n-1)(n-2)/3 + 1$	

Table 13 CUT6-U rules

n	Point selection			
2	$r_1\sigma_i$	$r_2 c_i^n$	$r_3 c_i^n$	—
	w_1	w_2	w_3	—
3, 4	$r_1\sigma_i$	$r_2 c_i^n$	$r_3 c_i^2$	$r_4 c_i^n$
	w_1	w_2	w_3	w_4
5, 6, 7	$r_1\sigma_i$	$r_2 c_i^n$	$r_3 c_i^3$	$r_4 c_i^n$
	w_1	w_2	w_3	w_4
8, 9	$r_1\sigma_i$	$r_2 c_i^n$	$r_3 c_i^4$	$r_4 c_i^n$
	w_1	w_2	w_3	w_4

Table 14 Fully symmetric set of points for CUT6-U

Dimension	σ	c^2	c^3	c^4	c^5	c^6	c^7	c^8	c^9	s^n
2	**4**	**4**	—	—	—	—	—	—	—	8
3	**6**	**12**	**8**	—	—	—	—	—	—	24
4	**8**	**24**	32	**16**	—	—	—	—	—	64
5	**10**	40	**80**	80	**32**	—	—	—	—	160
6	**12**	60	**160**	240	192	**64**	—	—	—	384
7	**14**	84	**280**	560	672	448	**128**	—	—	896
8	**16**	112	448	**1120**	1792	1792	1024	**256**	—	2048
9	**18**	144	672	**2016**	4032	5376	4608	2304	**512**	4608

Table 15 Fully symmetric set of points for CUT8-G

Dimension	σ	c^2	c^3	c^4	c^5	c^6	c^7	c^8	c^9	s^n
2	**4**	**4**	—	—	—	—	—	—	—	**8**
3	**6**	**12**	**8**	—	—	—	—	—	—	**24**
4	**8**	**24**	32	**16**	—	—	—	—	—	**64**
5	**10**	40	**80**	80	**32**	—	—	—	—	**160**
6	**12**	60	**160**	240	192	**64**	—	—	—	**384**

The CUT8-U points have been successfully developed only for dimensions $n = 2, 3, 4, 5$, and the solution for the particular selection of points shown in Table 17 is documented in Table 31 of the Appendix.

4.4 Comparison to Other Cubature Methods. For the dimensions and order described in this paper, the CUT-G is compared to equivalent Gauss–Hermite quadrature, Gauss Hermite Smolyak quadrature, and Kronrod Patterson Smolyak quadrature (KPS) [29,30]. The recently developed high-order cubature Kalman filter [31] for fifth (denoted as HCKF-5) and seventh (denoted as HCKF-7) degrees is also considered for a thorough comparison.

Tables 18–20 show the number of points, N, used by each of the methods considered for specific dimensions. In addition, the stability factor for each method is also shown. The stability factor, as described in Refs. [8] and [13], is defined as $\sum_{i=0}^{N} |w_i|$, i.e., the sum of absolute values of the weights. If some weights are negative, then the stability factor would be greater than 1 leading to a cubature rule or formula that is inaccurate. It is clear that GHS has all negative weights for the dimensions and orders considered. The KPS method has negative weights for $n \geq 5$ for fifth-order quadrature rules and negative weights from $n \geq 3$ for seventh-order quadrature rules. From Tables 18–20 and Figs. 3(a)–3(c), it is evident that CUT4-G, CUT6-G, and CUT8-G provide minimal number of quadrature points with all positive weights.

In the case of the uniform PDF, Figs. 4(a)–4(d) compare the number of points required to exactly evaluate the expectation integrals of polynomials up to total degree 5, 7, and 9. Figure 4(c), in particular, compares only the CUT6 points and the equivalent Sparse Gauss–Legendre quadrature with four points in each dimension. It can be seen that the CUT-U points, similar to the CUT-G points, are much lower than the equivalent Gauss–Legendre quadrature. Though in some cases the sparse grid (Smolyak scheme) Gauss–Legendre quadrature points achieve fewer points than the CUT-U points, they do so by introducing negative weights. On the other hand, the CUT-U points have all positive weights.

5 Numerical Results

In this section, a few benchmark nonpolynomial functions are considered to illustrate the effectiveness of the proposed CUT method in numerical evaluation of expectation integrals involving Gaussian as well as uniform PDFs. Furthermore, a nonlinear filtering problem and an optimal control problem with uncertain parameters are also considered to show the diversity of applications which can benefit from CUT. The notations such as GH3, GH4, … imply Gauss–Hermite product rule with $3, 4, \ldots$ quadrature points in a single dimension and hence the tensor product would result in a total of $3^n, 4^n, \ldots$ points, respectively. Similar notation is adopted for the GHS, GLgn, and GLgnSM points.

Table 16 Points for CUT8-G, ($2 \leq n \leq 6$)

	Position	Weights
$1 \leq i \leq 2n$	$\mathbf{x}_i = r_1\sigma_i$	$\mathbf{w}_i = w_1$
$1 \leq i \leq 2^n$	$\mathbf{x}_{i+2n} = r_2 c_i^n$	$\mathbf{w}_{i+2n} = w_2$
$1 \leq i \leq 2n(n-1)$	$\mathbf{x}_{i+2n+2^n} = r_3 c_i^2$	$\mathbf{w}_{i+2n+2^n} = w_3$
$1 \leq i \leq 2^n$	$\mathbf{x}_{i+2n+2^n+2n(n-1)} = r_4 c_i^n$	$\mathbf{w}_{i+2n+2^N+2n(n-1)} = w_4$
$1 \leq i \leq n_1$	$\mathbf{x}_{i+2n+2^n+2n(n-1)+2^n} = r_5 c_i^3$	$\mathbf{w}_{i+2n+2^n+2n(n-1)+2^n} = w_5$
$1 \leq i \leq n2^n$	$\mathbf{x}_{i+2n+2^n+2n(n-1)+2^n+n_1} = r_6 \,_{\sin}$	$\mathbf{w}_{i+2n+2^n+2n(n-1)+2^n+n_1} = w_6$
Central weight	$\mathbf{x}_0 = \mathbf{0}$	$\mathbf{w}_0 = w_0$
	$N = 2n + 2^n + 2n(n-1) + n_1 + 2^n + n2^n + 1 : \{n_1 = 4n(n-1)(n-2)/3\}$	

Table 17 CUT8-U points

N	Point selection					
2	$r_1\sigma_i$ / w_1	$r_2c_i^2$ / w_2	$r_{3\,\sin}(h)$ / w_3	$r_4c_i^2$ / w_4	— / —	— / —
3	$r_1\sigma_i$ / w_1	$r_2c_i^n$ / w_2	$r_3c_i^2$ / w_3	$r_4c_i^n$ / w_4	$r_{5\,\sin}(h)$ / w_5	— / —
4	$r_1\sigma_i$ / w_1	$r_2c_i^n$ / w_2	$r_3c_i^2$ / w_3	$r_4c_i^3$ / w_4	$r_{5\,\sin}(h)$ / w_5	$r_6c_i^n$ / w_6
5	$r_1\sigma_i$ / w_1	$r_2c_i^n$ / w_2	$r_3c_i^2$ / w_3	$r_4c_i^3$ / w_4	$r_{5\,\sin}(h)$ / w_5	$r_6c_i^n$ / w_6

5.1 Expectation Integral Evaluations

5.1.1 Example 1: Polynomial Function: Gaussian Probability Density Function. In the first example, the problem of evaluation of expectation value of an eighth-degree polynomial function in $6\mathbb{D}$ space is considered

$$Q = \int 0.1 \sum_{i=1}^{n} x_i^8 \mathcal{N}(x\!:0,I)\,dx$$

The true value of this integral is 63, and Table 21 lists the numerical value of the integral computed with the help of different methods. Table 21 also lists the number of quadrature points (N), summation of weights, and minimum value of quadrature weight. As the integrand is always non-negative, the integral value should also be always non-negative. As expected, the evaluation of the integral becomes more and more accurate as the order of the cubature rule is increased and cubature rule of order 9 (i.e., GH5, CUT8 and GHS5) exactly reproduces the true integral value. Notice that the fifth-order cubature rule, HCKF–5, in Ref. [31] estimates a negative value -28.8 due to negative quadrature weights. The presence of negative weights may reduce the accuracy, but can even pose a more significant disadvantage when they are used to integrate positive functions (such as estimating the covariance matrix). This example clearly illustrates the importance of quadrature points with positive weights.

5.1.2 Example 2: Nonpolynomial Function: Gaussian Probability Density Function. This example is taken from Ref. [31], where the following nonpolynomial function is integrated with respect to a zero mean identity covariance Gaussian PDF:

$$Q = \int \cos(\|\mathbf{x}\|_2)\mathcal{N}(x\!:0,I)\,dx$$

The integral is of dimension $n = 6$. The true solution as given in Ref. [31] is -0.543583844. Table 22 lists the integral value

Table 18 Number of points for fifth-order accurate cubature methods

Dim	GH		GHS		KPS		CUT4-G		HCKF-5	
	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N
2	1	**9**	5	13	1	**9**	1	**9**	1	**9**
3	1	27	13	25	1	19	1	**14**	1	19
4	1	81	25	41	1	33	1	**24**	1	33
5	1	243	41	61	2.11	51	1	**42**	1.2041	51
6	1	729	61	85	3.67	**73**	1	76	1.37	**73**
7	1	2187	85	113	5.67	**99**	1	142	1.52	**99**
8	1	6561	113	145	8.11	**129**	1	272	1.64	**129**
9	1	19,683	145	181	11	**163**	1	530	1.74	**163**

Note: $N \Rightarrow$ minimal number of quadrature points and $\underline{N} \Rightarrow$ minimal number of quadrature points with positive weight.

Table 19 Number of points for seventh-order accurate cubature points

Dim	GH		GHS		KPS		CUT6-G		HCKF-7	
	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N
2	1	16	7	29	1	17	1	**13**	1	24
3	1	64	25	69	1.654	39	1	**27**	1	76
4	1	256	63	137	2.293	81	1	**49**	1	176
5	1	1024	129	241	2.931	151	1	**83**	1	340
6	1	4096	231	389	4.754	257	1	**137**	1.56	584
7	1	16,384	377	589	7.614	**407**	1	**423**	2.1	924
8	1	65,536	575	849	11.808	**609**	1	**721**	2.57	1376
9	1	262,144	833	1177	17.63	**871**	1	**1203**	3.01	1956

Note: $N \Rightarrow$ minimal number of quadrature points and $\underline{N} \Rightarrow$ minimal number of quadrature points with positive weight.

obtained by the various methods. The fifth degree cubature rule HCKF–5 with 73 points has negative weights and has 14.72% relative error. In Table 1 of Ref. [31], the seventh-order higher cubature rule with 584 points achieves 1% relative error while CUT4-G achieves the 1% error with only 76 points as all the *weights are positive*. Further, CUT6-G achieves 0.3% error with only 137 points, again because all the weights are positive. Sparse grid quadrature that is accurate up to ninth degree achieves the 1% relative error with 1433 points due to the presence of negative weights. The superior performance of the CUT method can again be attributed to all positive weights.

5.1.3 Polar to Cartesian Coordinates ($x = r\cos\theta, y = r\sin\theta$)

$$E\left[\begin{bmatrix} x \\ y \end{bmatrix}\right] = \iint \begin{bmatrix} r\cos(\theta) \\ r\sin(\theta) \end{bmatrix} \mathcal{N}\left(\begin{bmatrix} r \\ \theta \end{bmatrix}\!:\begin{bmatrix} \mu_r \\ \mu_\theta \end{bmatrix}, \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix}\right) dr\,d\theta \tag{62}$$

This problem of uncertainty transformation under a coordinate change [28] was introduced to illustrate the effectiveness of the conventional UT. The example is resimulated in Fig. 5(*a*) while making use of same parameters as described in Ref. [28]. The mean and 1-σ contours are plotted for each method in Fig. 5(*a*). The truth is taken to be the result of Monte Carlo integration using 3×10^6 random samples in two-dimensional space. All the methods, except CKF, seem to do equally well compared to the MC simulations. CKF predicts the mean correctly but covariance is slightly under estimated due to error in all the fourth moments but UT can capture one fourth-order moment for dimensions ≤ 3. For the same example when the parameters are changed as $E[r] = 50\,\text{m}$, $E[\theta] = 0\,\text{deg}$, $\sigma_r^2 = 0.02^2\,\text{m}^2$, $\sigma_\theta^2 = 30^2\,\text{deg}^2$, the inclusion of higher moment constraints tends to do better than the conventional UT as seen in Fig. 5(*b*). The CKF slightly under estimates the covariance while the UT and GH3 slightly over estimate the covariance. CUT4-G in $2\mathbb{D}$ can additionally capture one

Table 20 Number of points for ninth-order accurate cubature points

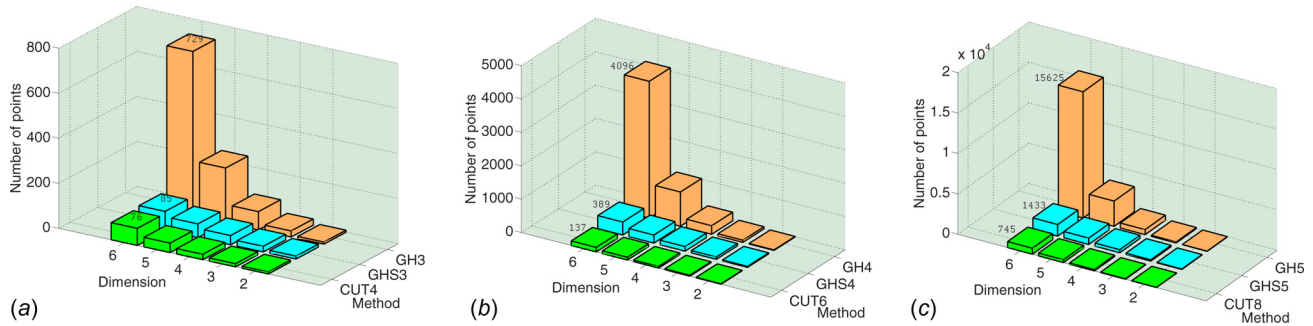| Dim	GH		GHS		KPS		CUT8-G	
	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N	$\sum\|w_i\|$	N
2	1	25	9	53	1.485	37	1	**21**
3	1	125	41	165	2.6360	93	1	**59**
4	1	625	129	385	3.825	201	1	**161**
5	1	3125	321	781	4.95	401	1	**355**
6	1	15,625	681	1433	6.07	749	1	**745**

Fig. 3   Comparison of number of points for Gaussian PDF: (*a*) fifth-order, (*b*) seventh-order, and (*c*) ninth-order

of the sixth-order moments and hence does better than GH3 and UT. The mean and covariance can be evaluated analytically [32] by the following expressions where $(\bar{x}, \bar{y}) = (\mu_r \cos(\mu_\theta), \mu_r \sin(\mu_\theta))$, $\lambda_1 = \exp[-\sigma_\theta^2/2]$ $\lambda_2 = \frac{1}{2}(1 + \exp[-2\sigma_\theta^2])$ and $\lambda_3 = (1/2)(1 - \exp[-2\sigma_\theta^2])$

$$E\left[\begin{bmatrix} x \\ y \end{bmatrix}\right] = \lambda_1 \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$$

$$E\left[(x - E[x])^2\right] = -\lambda_1^2 \bar{x}^2 + \bar{x}^2 \lambda_2 + \frac{\sigma_r^2 \bar{x}^2 \lambda_2}{\bar{x}^2 + \bar{y}^2} + \bar{y}^2 \lambda_3 + \frac{\sigma_r^2 \bar{y}^2 \lambda_3}{\bar{x}^2 + \bar{y}^2}$$

$$E\left[(y - E[y])^2\right] = -\lambda_1^2 \bar{y}^2 + \bar{y}^2 \lambda_2 + \frac{\sigma_r^2 \bar{y}^2 \lambda_2}{\bar{x}^2 + \bar{y}^2} + \bar{x}^2 \lambda_3 + \frac{\sigma_r^2 \bar{x}^2 \lambda_3}{\bar{x}^2 + \bar{y}^2}$$

$$E\left[(x - E[x])(y - E[y])\right] = -\lambda_1^2 \bar{x}\bar{y} + \bar{x}\bar{y}\lambda_2 + \frac{\sigma_r^2 \bar{x}\bar{y}\lambda_2}{\bar{x}^2 + \bar{y}^2} - \bar{x}\bar{y}\lambda_3 - \frac{\sigma_r^2 \bar{x}\bar{y}\lambda_3}{\bar{x}^2 + \bar{y}^2}$$

In Table 23, the various methods are compared to the analytical result and the %-absolute relative error is tabulated for the means and variances clearly illustrating the improved accuracy of the CUT method relative to the other techniques.

5.1.4   Nonpolynomial Function: Uniform Probability Density Function. To assess the performance of CUT-U points, the

Table 21   Comparison of expectation integrals for Example 1

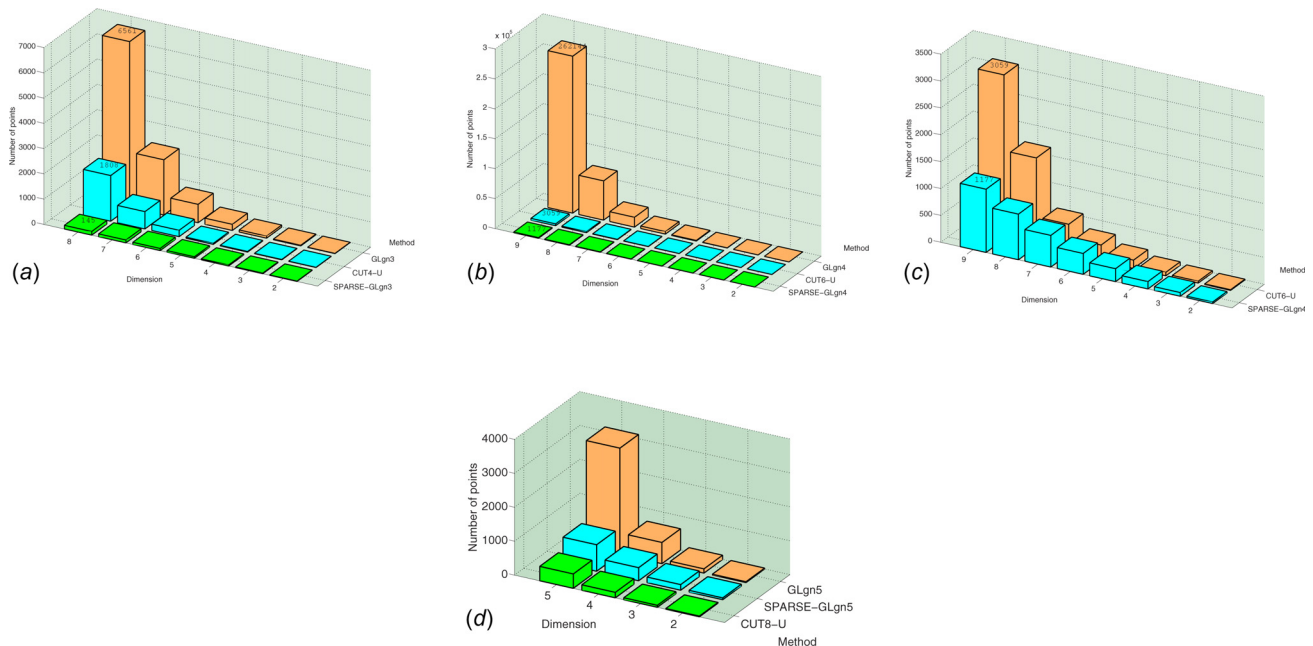| Method	Q_{approx}	min(w)	$\sum \lvert w_i \rvert$	N
UT	205.8000	0.0714	1	13
HCKF–5	−28.8000	−0.0156	1.3750	73
CUT4–G	21.6000	0.0039	1	76
CUT6–G	60.5981	8.2885×10^{-004}	1	137
CUT8–G	63.0000	7.8492×10^{-005}	1	745
GH3	16.2000	2.1433×10^{-005}	1	729
GH4	48.6000	9.3219×10^{-009}	1	4096
GH5	63.0000	2.0353×10^{-012}	1	15,625
GH6	63.0000	2.7871×10^{-016}	1	46,656
GH7	63.0000	2.7162×10^{-020}	1	117,649
GH8	63.0000	2.0397×10^{-024}	1	262,144
GHS3	16.2000	−2.5000	61	85
GHS4	48.6000	−30.0000	231	389
GHS5	63.0000	−13.3333	681	1433
GHS6	63.0000	−90.3333	1683	4541
GHS7	63.0000	−34.9444	3653	12,841
GHS8	63.0000	−199.3439	7183	33,193

Fig. 4   Comparison of number of points for uniform PDF: (*a*) fifth-order, (*b*) seventh-order, (*c*) seventh-order (CUT versus Sparse GLgn4), and (*d*) ninth-order

computation of the integral of $f(\mathbf{x}) = \left(\sqrt{1 + \mathbf{x}^T\mathbf{x}}\right)^{-3}$ over $\mathcal{H}$, i.e., a hypercube in n-$\mathbb{D}$ with opposite corners at $[-1, -1,\ldots, -1]$ and $[1, 1,\ldots, 1]$ is considered

$$E[f(\mathbf{x})] = \frac{1}{2^n}\int_{\mathcal{H}}\left(\sqrt{1 + \mathbf{x}^T\mathbf{x}}\right)^{\alpha} d\mathbf{x}, \quad \alpha = -3$$

This problem was introduced in Ref. [21] where it had been discussed that computing the expectation of $f(\mathbf{x})$ for negative values of α is a challenging task since $\alpha < 0$ leads to delta-sequence functions. The convergence of the integral using the GLgn rules for various dimensions is shown in Fig. 6(a) and the converged value for each dimension is used to compute the relative error for other methods. Figure 6(b) shows the relative error using the sparse-GLgn method (GLgnSM). The convergence in the integral is very slow and in higher dimensions such as 8 and 9, the integral does not seem to converge. The error in the integral value first rapidly increases to huge values and then gradually decreases for a very large number of points. This could be due to the presence of large negative weights for many points. Figure 6(c) shows the number of CUT-U points required to achieve at least 10% relative error in contrast to GLgn. For GLgnSM method, the relative error is below 10% only for dimensions 2 and 3 and hence the results are not shown in this figure.

5.2 Nonlinear Filtering Examples. The ability of CUT methodology to compute multidimensional expectation integrals in an accurate and computationally efficient manner makes it an attractive tool to solve nonlinear filtering problems. Let us consider the following discrete time dynamic system:

$$\mathbf{x}_{k+1} = \phi(\mathbf{x}_k) + \boldsymbol{\omega}_k \qquad (63)$$

$$\mathbf{y}_{k+1} = \mathbf{h}(\mathbf{x}_{k+1}) + \boldsymbol{v}_{k+1} \qquad (64)$$

where $\phi(\mathbf{x}_k): R^{n_x} \to R^{n_x}$ is the discrete model for the dynamics of the system and can be thought of as the flow of nonlinear continuous time dynamics of the system. $h(x): R^{n_x} \to R^{n_y}$ is the measurement model, n_x and n_y being the dimensions of the process model and measurement model, respectively. $\boldsymbol{\omega}_k$ and $\boldsymbol{v}_{k+1}$ are assumed to be zero-mean Gaussian white noise sequences with covariances $\mathbf{Q}_k$ and $\mathbf{R}_{k+1}$, respectively. The filtering algorithm is initiated with uncertain initial condition $\mathbf{x}_0 \sim \mathcal{N}\left(\mathbf{x}_0 : \hat{\mathbf{x}}_{0/0}, \mathbf{P}_{0/0}^x\right)$, a Gaussian

PDF with mean $\hat{\mathbf{x}}_{0/0}$ and covariance matrix $\mathbf{P}_{0/0}^x$ and corresponds to computing posterior mean, $\hat{\mathbf{x}}_{k/k}$, and covariance, $\mathbf{P}_{k/k}$ given the measurements up to time k. The notation used in this section is adopted from Ref. [33].

In this work, we utilize the linear minimum variance estimator formulation to simplify the nonlinear filtering problem. The details of the nonlinear filtering algorithm are similar to the UKF [11], where the UT points are replaced by higher-order CUT-G points to compute the expectation integrals involved in every iteration of the filter. The minimum variance formulation is valid for any PDF although it makes use of only the mean and covariance information of the density function. According to the minimum variance formulation, the posterior mean and covariance can be computed, given an estimate of the prior mean and covariance [24,34,35]:

$$\hat{\mathbf{x}}_{k+1/k+1} = \hat{\mathbf{x}}_{k+1/k} + \mathbf{K}_{k+1}\left(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k=1}\right) \qquad (65)$$

$$\mathbf{P}_{k+1/k+1} = \mathbf{P}_{k+1/k} - \mathbf{K}_{k+1}\left(\mathbf{P}_{k+1/k}^{xy}\right)^T \qquad (66)$$

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1}^{xy}\left(\mathbf{P}_{k+1/k}^{y}\right)^{-1} \qquad (67)$$

where $\mathbf{K}_{k+1}$ is known as the Kalman gain matrix and other quantities are defined as

$$\hat{\mathbf{y}}_{k=1} = E\left[\mathbf{h}(\mathbf{x}_{k+1/k})\right] \qquad (68)$$

$$\mathbf{P}_{k+1/k}^{y} = E\left[\mathbf{h}(\mathbf{x}_{k+1/k})\mathbf{h}(\mathbf{x}_{k+1/k})^T\right] - \hat{\mathbf{y}}_{k+1}\hat{\mathbf{y}}_{k+1}^T + \mathbf{R}_{k+1} \qquad (69)$$

$$\mathbf{P}_{k+1}^{xy} = E\left[\mathbf{x}_{k+1/k}\mathbf{h}(\mathbf{x}_{k+1/k})^T\right] - \hat{\mathbf{x}}_{k+1/k}\hat{\mathbf{y}}_{k+1}^T \qquad (70)$$

It should be emphasized that nonlinear filters like the extended Kalman filter (EKF), UKF, CKF, and quadrature Kalman filter (QKF) all make use of the minimum variance formulation. However, each of these methods makes different approximations to evaluate the expectations of the quantities involved. The EKF involves the linearization of both, the system dynamics and measurement model, about the mean and assumes the state PDF to be Gaussian. The EKF has been used successfully in many applications; however, it may result in poor performance due to large errors in state estimates or sparse measurement data [33]. Furthermore, the linearization involves the calculation of Jacobians that could be computationally expensive procedure. On the other hand, nonlinear filters such as UKF, CKF, and QKF do not involve linearization but numerically evaluate the expectation integrals by making use of the specific quadrature scheme. When UT-derived sigma points are used for numerical integration, the resultant filter is called UKF [11] and when $\kappa = 0$, the filter is equivalent to CKF [21]. When Gauss–Hermite quadrature is used, the filter corresponds to QKF [36] and when Sparse grid quadrature such as Smolyak quadrature is used, the resultant filter is equivalent to the filter developed in Ref. [37]. As discussed earlier, UKF and CKF are only third-order cubature rules and can evaluate the state mean $\hat{\mathbf{x}}_{k/k}$ exactly if the process model $\phi(\mathbf{x}_k)$ and measurement model $\mathbf{h}(\mathbf{x}_{k+1})$ are polynomials of degree 3 or less. As the degree of the $\phi(\mathbf{x}_k)$ and $\mathbf{h}(\mathbf{x}_{k+1})$ increases above 3, errors will be introduced at each stage. Ideally, one would desire to use the Gauss–Hermite product rule as in QKF, but the number of quadrature points grows exponentially with dimension. The minimal cubature rules and CUT sigma points achieve higher accuracy than UKF or CKF with only a fraction of the points used by QKF. Unlike the Smolyak quadratures, the CUT points have all positive weights making it as ideal choice for nonlinear filtering. After each measurement update, the state density function is approximated as Gaussian PDF with mean $\hat{\mathbf{x}}_{k+1/k}$ and covariance $\mathbf{P}_{k+1/k}$. A new set of quadrature points are generated from this new

Table 22 Comparison of expectation integrals for Example 2

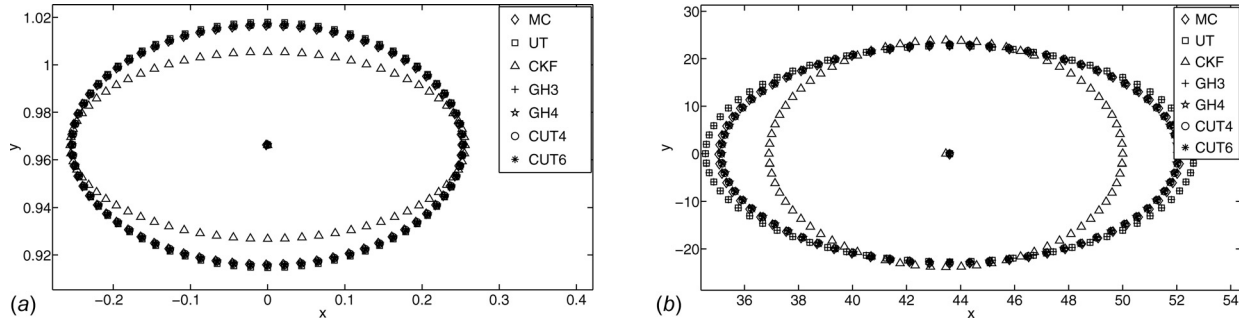| Method	Q_{approx}	%err	min(w)	$\sum|w_i|$	N
UT	-0.6111	12.4130	0.0714	1	13
HCKF–5	-0.4635	14.7285	-0.0156	1.3750	73
CUT4–G	-0.5492	1.0370	0.0039	1	76
CUT6–G	-0.5419	0.3013	8.2885×10^{-004}	1	137
CUT8–G	-0.5430	0.0995	7.8492×10^{-005}	1	745
GH3	-0.5162	5.0418	2.1433×10^{-005}	1	729
GH4	-0.5457	0.3918	9.3219×10^{-009}	1	4096
GH5	-0.5435	0.0229	2.0353×10^{-012}	1	15,625
GH6	-0.5436	0.0011	2.7871×10^{-016}	1	46,656
GH7	-0.5436	4.2494×10^{-005}	2.7162×10^{-020}	1	117,649
GH8	-0.5436	1.4920×10^{-006}	2.0397×10^{-024}	1	262,144
GHS3	-0.1910	64.8579	-2.5000	61	85
GHS4	-0.6023	10.7967	-30.0000	231	389
GHS5	-0.5370	1.2203	-13.3333	681	1433
GHS6	-0.5441	0.1026	-90.3333	1683	4541
GHS7	-0.5435	0.0068	-34.9444	3653	12,841
GHS8	-0.5436	3.6928×10^{-004}	-199.3439	7183	33,193

Fig. 5 Simulation results for polar to Cartesian transformation: (*a*) $\mu_r = 1$ m, $\mu_\theta = 90$ deg, $\sigma_r^2 = 0.02^2$ m^2, $\sigma_\theta^2 = 15^2$ deg^2 and (*b*) $\mu_r = 50$ m, $\mu_\theta = 0$ deg, $\sigma_r^2 = 0.02^2$ m^2, $\sigma_\theta^2 = 30^2$ deg^2

Table 23 Comparison of various methods for polar to Cartesian conversion

% Relative error	UT	CKF	GH3	GH4	CUT4–G	CUT6–G
$E[x]$	0.0185	0.3246	0.0185	0.0004	0.0002	0.0002
std. dev. in x	6.7088	22.7811	6.7087	0.5722	0.2288	0.2490
std. dev. in y	1.0163	3.8434	1.0163	0.0790	0.0317	0.0345

Gaussian density function, which are propagated through the nonlinear system equations until the next measurement instant. The filter algorithm that directly incorporates these cubature/sigma points by evaluating the expectation integral as a weighted average is described in Fig. 8. In case there are no measurements available at time $k+1$, the measurement step is avoided altogether and the algorithm proceeds to the time evolution step. It is interesting to see that the sigma points can be generated from any quadrature method and the filter algorithm would remain unchanged. In Ref. [4], a polynomial chaos-based filtering algorithm is presented, which makes use of quadrature schemes to update higher-order moments without any assumption on initial density function.

It should be noticed that all the cubature/sigma points can be generated offline and stored at time $k = 0$ for zero mean and identity covariance Gaussian PDF of appropriate dimension. At each given time k, the required cubature/sigma points can be generated by a simple affine transformation as in Eq. (15).

5.2.1 Air Traffic Tracking Scenario. A typical air traffic control scenario [35] is simulated using various cubature/sigma points. The simulation consists of an aircraft that executes the following maneuvers in order: Horizontal motion for 125 s at a speed of 120 m/s toward west, then a coordinated turn (CT) of 90 deg toward south for 90 s with a turn rate of 1 deg/s. The aircraft continues to head south at a speed of 120 m/s for 125 s, finally takes another coordinated turn toward west with turn rate of -3 deg/s for 30 s, thus completing a 90 deg turn and then heading west at a speed of 120 m/s for 125 s. The trajectory of the flight is shown in Fig. 7(*a*). The kinematics of the turning motion is modeled by the set of nonlinear equations called coordinated turn as described in

Ref. [35]. The CT model is characterized by constant speed and constant turn rate. The turn rate Ω is usually unknown and is hence appended to the state vector making the model nonlinear. The system equations for CT model, where the state vector is $x = \begin{bmatrix} \xi & \dot{\xi} & \eta & \dot{\eta} & \Omega \end{bmatrix}^T$ and radar model that measures the range and bearing are:

$$x_k = \begin{bmatrix} 1 & \dfrac{\sin(\Omega T)}{\Omega} & 0 & -\dfrac{1 - \cos(\Omega T)}{\Omega} & 0 \\ 0 & \cos(\Omega T) & 0 & -\sin(\Omega T) & 0 \\ 0 & \dfrac{1 - \cos(\Omega T)}{\Omega} & 1 & \dfrac{\sin(\Omega T)}{\Omega} & 0 \\ 0 & \sin(\Omega T) & 0 & \cos(\Omega T) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + \omega_{k-1}$$

(71)

$$\begin{bmatrix} r_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} \sqrt{(\xi_k)^2 + (\eta_k)^2} \\ \tan^{-1}\left(\dfrac{\eta_k}{\xi_k}\right) \end{bmatrix} + \nu_k$$

(72)

The process noise ω_{k-1} and measurement noise ν_k are independent zero mean Gaussian noise processes with covariance matrices

$$Q_{k-1} = L_1 \begin{bmatrix} \dfrac{T^3}{3} & \dfrac{T^2}{2} & 0 & 0 & 0 \\ \dfrac{T^2}{2} & T & 0 & 0 & 0 \\ 0 & 0 & \dfrac{T^3}{3} & \dfrac{T^2}{2} & 0 \\ 0 & 0 & \dfrac{T^2}{2} & T & 0 \\ 0 & 0 & 0 & 0 & \dfrac{L_2}{L_1}T \end{bmatrix}, \quad R_k = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix}$$
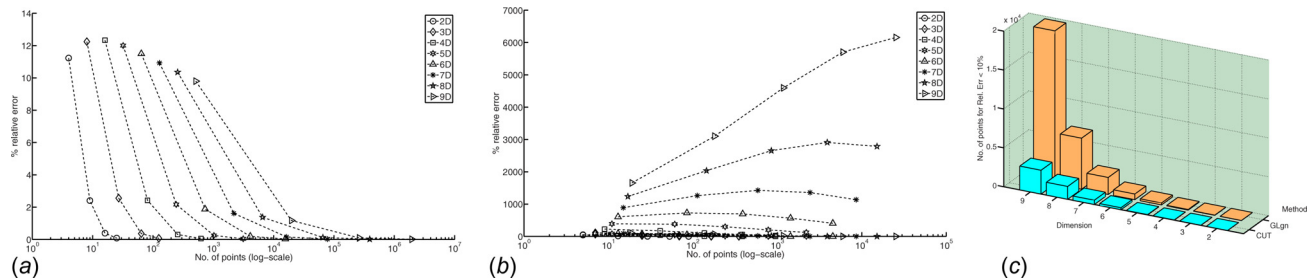

Fig. 6 Nonpolynomial function-uniform PDF: % relative error: (*a*) Gauss–Legendre, (*b*) sparse grid Gauss–Legendre, and (*c*) 10% relative error GLgn versus CUT-U
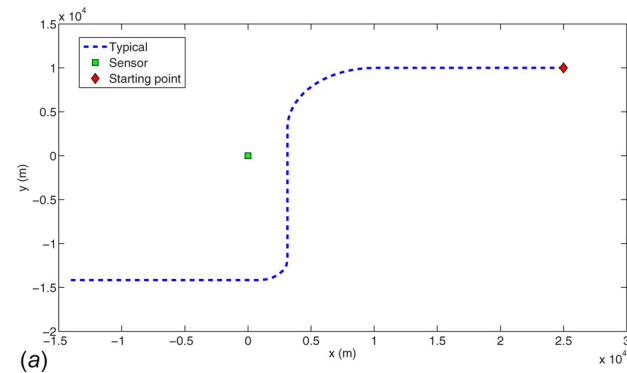
The parameters used in the simulation are $L_1 = 0.16$, $\sigma_r = 100$ m, $L_2 = 0.01$, $\sigma_\theta = 1$ deg. The initial condition uncertainty is $x_0 = [25{,}000\,\text{m}, -120\,\text{m/s}, 10{,}000\,\text{m}, 0\,\text{m/s}, 0.000001\,\text{rad/s}]^T$ and $P_{0/0} = \text{diag}([1000^2\,\text{m}^2, 100\,\text{m}^2/\text{s}^2, 1000^2\,\text{m}^2, 100\,\text{m}^2/\text{s}^2, (1\pi/180)^2\,\text{rad}^2/\text{s}^2])$. The following filters are used in the simulations: CKF with 10 points, UKF with 11 points, CUT4-G with 42 points, CUT6-G with 83 points, CUT8-G with 355 points, and particle filter (PF)—with 5000 sample points. A simulation of all these filters is shown in Fig. 7(b). The trajectories of each filter shown in Fig. 7(b) are the average over 100 runs.

As the dimension of the process model is 5, a kappa value of $\kappa = 1$ is considered during the implementation of the UKF to avoid a negative weight in the UT. Each run consists of initiating all the filters with the same mean and covariance of the initial time Gaussian PDF and generating random measurements from the true trajectory by the sensor model (72). All the filters are subjected to the same set of measurements for a particular run. A batch of $N_R = 100$ runs are performed for each considered filter. The root-mean-square-error (RMSE) in position, velocity, and turn rate for the batch of runs is calculated as

$$\text{RMSE}_{\text{pos}}(k) = \sqrt{\frac{1}{N_R}\sum_{i=1}^{N_R}((\xi_i(k) - \hat{\xi}_i(k))^2 + (\eta_i(k) - \hat{\eta}_i(k))^2)}$$

$$\text{RMSE}_{\text{vel}}(k) = \sqrt{\frac{1}{N_R}\sum_{i=1}^{N_R}((\dot{\xi}_i(k) - \hat{\dot{\xi}}_i(k))^2 + (\dot{\eta}_i(k) - \hat{\dot{\eta}}_i(k))^2)}$$

$$\text{RMSE}_\Omega(k) = \sqrt{\frac{1}{N_R}\sum_{i=1}^{N_R}((\Omega_i(k) - \hat{\Omega}_i(k))^2)}$$

$$\|\text{RMSE}_{\text{pos}}\|_2 = \sqrt{\frac{1}{N_t}\sum_{k=1}^{N_t}\text{RMSE}_{\text{pos}}(k)^2}$$

$$\|\text{RMSE}_{\text{vel}}\|_2 = \sqrt{\frac{1}{N_t}\sum_{k=1}^{N_t}\text{RMSE}_{\text{vel}}(k)^2} \qquad (73)$$

$$\|\text{RMSE}_\Omega\|_2 = \sqrt{\frac{1}{N_t}\sum_{k=1}^{N_t}\text{RMSE}_\Omega(k)^2}$$

where N_t is the number of time steps. The RMSE in position and velocity is calculated individually for each filter estimate $[\hat{\xi}_i(k), \hat{\dot{\xi}}_i(k), \hat{\eta}_i(k), \hat{\dot{\eta}}_i(k), \hat{\Omega}_i(k)]$, where i corresponds to the run number, with respect to the true trajectory $[\xi_i(k), \dot{\xi}_i(k), \eta_i(k), \dot{\eta}_i(k), \Omega_i(k)]$. In Fig. 9, the 2-norm of the RMSE is plotted against the measurement time interval T. It can be observed that as the measurement interval increases, the error in each filter increases. The UKF and CKF have large errors when compared to the higher-order filters. Furthermore, the particle filter estimates have the least error. CUT8-G has the similar order of error compared to the particle filter with 5000 particles.

In Table 24, the 2-norm of the RMSE in position, velocity, and angular rate is shown for the case when $T = 5$ s. With significant time between consecutive measurements, the lower-order sigma point filters CKF and UKF seem to diverge with large velocity errors. Further, CUT4-G and CUT6-G also seem to have large velocity errors rendering the estimates unusable. But, CUT8-G has performed significantly better with similar order of errors compared to the particle filter. Further there is a gradual improvement as the order of the filters is increased. CKF and UKF being third-order rules have the maximum error.

5.2.2 Lorenz Model. Lorenz model consists of a system with three coupled nonlinear differential equations

$$\dot{x} = \sigma(y - x), \quad \dot{y} = \rho x - y - xz, \quad \dot{z} = xy - \beta z \qquad (74)$$

The system parameters are $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$. The initial condition considered is

$$[x, y, z]^T = [1.50887, -1.531271, 25.46091]^T \qquad (75)$$

A sample trajectory for the Lorenz system with these parameters is shown in Fig. 10. The uncertainty lies in σ, ρ, and the initial conditions. The parameters σ and ρ are considered to be uncertain and hence appended to the state vector $[x, y, z]$. The uncertainty in the initial condition $\mathbf{x} = [x, y, z, \sigma, \rho]$ is modeled by a Gaussian PDF $\mathcal{N}(\mathbf{x}: \boldsymbol{\mu}_0, P_0)$ where

$$\boldsymbol{\mu}_0 = [1.50887, -1.531271, 25.46091, 10, 28]^T; \ P_0$$
$$= \text{Diag}([4, 4, 4, 2, 4]^T) \qquad (76)$$

The measurement model consists of the three states $h(\mathbf{x}) = [x, y, z]^T$, where the measurements are made every T seconds. The measurement noise covariance is $R = \text{Diag}([3, 3, 3])$. The process noise covariance for the continuous dynamics is $Q = 2I_{5\times5}$. As measurement times are discrete, the process noise covariance for discrete implementation of the dynamics is approximated as
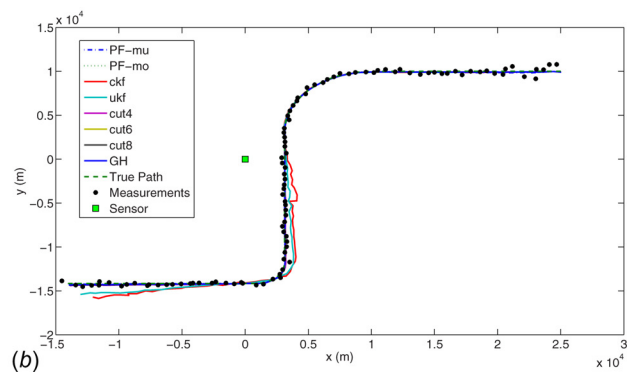

Fig. 7 Air traffic scenario: (a) typical aeroplane trajectory and (b) estimated aeroplane trajectory

$Q = 2TI_{5 \times 5}$. Between the measurements, the continuous dynamics are used to propagate the particle/sigma/cubature points. The filters PF, CKF, UKF, CUT4-G, CUT6-G, and CUT8-G are simultaneously initiated from the same mean and initial Gaussian PDF. A value of $\kappa = 1$ is chosen for the UKF. The bootstrap particle filter [38] is initiated with 5000 samples and has resampling with a threshold of $0.6N_p$ where N_p is the number of particles. A total of $N_R = 100$ runs are performed for each filter with each run having a different set of random measurements. The RMSE error for each time step is evaluated as $\mathrm{RMSE}(k) = \sqrt{(1/N_R)\sum_{i=1}^{N_R}||\hat{\mathbf{x}}_k^{(i)} - \mathbf{x}_k^{(i)}||^2}$ where $\hat{\mathbf{x}}_k^{(i)} = [\hat{x}_k, \hat{y}_k, \hat{z}_k, \hat{\sigma}_k, \hat{\rho}_k]$ is the estimate from the ith run and $\mathbf{x}_k^{(i)} = [x_k, y_k, z_k, \sigma_k, \rho_k]$ is the corresponding truth. The 2-norm and max of RMSE are evaluated as in Eq. (73).

In Fig. 11, the 2-norm of the RMSE and the max norm of the RMSE are plotted for each filter. It can be observed that the particle filter performs the best. The low-order filters UKF and CKF have large errors compared to the higher-order filters. As the measurement time interval T increases, the errors in the estimates of each filter increase, with the lower-order filters performing the worst.

A simulation with $T = 0.2$ and a low process noise of $Q = 0.002I_{5 \times 5}$ is also considered for each filter. From Table 25, it can be seen that the particle filter has degraded performance. This problem can be easily corrected by artificially inflating the state covariance at every time step as described in Ref. [39]. A second simulation with a slightly larger process noise of $Q = 0.005I_{5 \times 5}$ is considered with the same measurement time interval of $T = 0.2$ s. As expected, in this case, the particle filter outperforms other filters, while higher-order CUT filters are able to perform better than the UKF and the CKF. This inconsistency in the performance of particle filter as a function of process noise is a well-recognized fact and common among sequential Monte Carlo class of filters, which require special remedial actions as discussed in Ref. [39].

5.3 Helicopter Hovering Problem: Uncertainty Quantification.
An example of a hovering helicopter model is taken from Ref. [40] to assess the performance of the CUT-U points for uncertainty propagation through dynamical systems. The approximate closed-loop dynamic equations for the longitudinal motion of a hovering helicopter are given as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\delta, \quad \delta = -\mathbf{K}\mathbf{x} \qquad (77)$$

$$\mathbf{x} = \begin{bmatrix} u_h \\ q_h \\ \theta_h \\ y \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} p_1 & p_2 & -g & 0 \\ p_3 & p_4 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} p_5 \\ p_6 \\ 0 \\ 0 \end{bmatrix} \qquad (78)$$

with g being the acceleration due to gravity. The uncertainty lies in the first four parameters of the model, namely, the parameters $\mathbf{p} = [p_1, p_2, p_3, p_4]$. The various simulation parameters (initial condition and the controller gain) are given as follows:

$$\mathbf{x}_0 = [0.7929, -0.0466, -0.1871, 0.5780]^T \qquad (79)$$

$$\mathbf{K} = [1.9890, -0.2560, -0.7589, 1.0] \qquad (80)$$

For simulation's sake, the uncertainty in the first four parameters is assumed to be uniformly distributed within following bounds $\mathbf{p}_{lbd} = [-0.0488, 0.0013, 0.126, -3.3535]$ and $\mathbf{p}_{ubd} = [-0.0026, 0.0247, 2.394, -0.1765]$.

Two methods, GLgn and CUT-U rules, are used to propagate the statistical moments up to third-order for the state variables over a period of 15 s. 100,000 MC runs are assumed to provide the ground truth and are used to calculate the relative error. The 2-norms of all the moments at regular time intervals of 0.1 s up to

third-order are calculated individually. The 2-norm of the relative error over time is taken for each order of moment individually and the results are shown in Table 26. For each method, N corresponds to the number of points. It can be seen that CUT8-U and GLgn5, being of similar order, achieve the lowest error, but CUT8-U points achieve this with just one-fourth of the number of points used by GLgn5.

5.4 Control Under Parametric Uncertainty.
In this example, the rest-to-rest motion of a three mass spring damper system as shown in Fig. 12 is considered. The system is assumed to have uncertainties in both the spring constants and the damping constants, i.e., $\rho = [k_1, k_2, c_1, c_2]$. The uncertainty in ρ is considered to be modeled by a Gaussian PDF $p(\rho) = \mathcal{N}(\rho : \mu, P)$. The control input is applied to the first mass. Our objective is to find the control input such that the final residual energy is minimized

$$J = 5||\mathbf{x}_{t_f} - \mathbf{x}_{ref}||_2 + \frac{1}{2}m_1\dot{x}_1(t_f)^2 + \frac{1}{2}m_2\dot{x}_2(t_f)^2 + \frac{1}{2}m_3\dot{x}_3(t_f)^2$$
$$+ \frac{1}{2}k_1(x_1(t_f) - x_2(t_f))^2 + \frac{1}{2}k_2(x_2(t_f) - x_3(t_f))^2$$

Due to uncertainty in the model parameters, the state vector $\mathbf{x}_t = [x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3]$ and the cost function J are also uncertain. Hence, we seek a robust controller, which minimizes the maximum residual energy at final time due to uncertainty in system parameters. In Ref. [41], a polynomial chaos-based approach is described which minimizes the higher-order moments of residual energy to achieve this. This approach utilizes the conventional Gaussian quadrature methods to compute the higher-order moments of the final time residual energy and can be computationally expensive. In this work, we utilize the proposed CUT methodology to reduce the computation burden of this approach. We also compare the performance of this approach with conventional min–max controller.

The cost functions considered corresponding to min–max (see Eq. (81)) and min–moments cost (see Eq. (82)) are given as:

$$\text{min–max}: \quad \min_{\mathbf{u}} \max_{1 \leq i \leq N_\rho} p(\rho_i)J(\rho_i) \qquad (81)$$

$$\text{min–moments}: \quad \min_{\mathbf{u}}\{c_1 E[J] + c_2 E[J^2] + \cdots + c_M E[J^M]\} \quad (82)$$

$$\text{constraint to} \begin{cases} \dot{\mathbf{x}} = A(\boldsymbol{\rho})\mathbf{x} + B(\boldsymbol{\rho})u, \quad \rho \sim p(\rho) \\ -c \leq u \leq c \\ \mathbf{x} = \mathbf{x}_0 \quad \text{at} \quad t = 0 \end{cases} \qquad (83)$$

The optimization problems in Eqs. (81) and (82) share the same set of constraints (83). The various parameters assumed in this simulation are $m_1 = 1, m_2 = 1, m_3 = 1, \mu = [1, 1, 0.4, 0.4]^T$, $P = \mathrm{diag}([0.2^2, 0.2^2, 0.1^2, 0.1^2]), c = 1, \mathbf{x}_0 = [0, 0, 0, 0, 0, 0]^T$ and $\mathbf{x}_{ref}(t_f) = [1, 1, 1, 0, 0, 0]^T$. The control u is modeled as a bang–bang profile (of amplitude $c = 1$) with six unknown switch times. The min–max cost function in Eq. (81) is computed at a set of chosen points ρ_i within the parameter space with corresponding weight $p(\rho_i)$. In this simulation, GH5 with 625 points and weights are used in the min–max problem. The first two ($M = 2$) and four ($M = 4$) moments of J are minimized in Eq. (82) with the relative weights being ($c_1 = 0.67$, $c_2 = 0.33$) and ($c_1 = 0.4$, $c_2 = 0.3$, $c_3 = 0.2$, $c_4 = 0.1$), respectively. (In particular, the weights were selected as $c = ([M, M-1, ..., 1])/(\sum_{m=1}^{M} m)$.) The moments here are computed by only 161 CUT8-G points. The results of the optimization problems are shown in Fig. 13, where the optimal/suboptimal solutions for each method are used to evaluate the empirical distribution of J from 100,000 runs starting at random points from the parameter distribution $p(\rho)$. From Fig. 13, it can be observed that the min–max and second-order moment cost function achieve better performance (low mean) but have wider

$$* \ T_{k/k}(\mathrm{x}) = \left(\sqrt{P_{k/k}}\right)x + \hat{x}_{k/k}$$
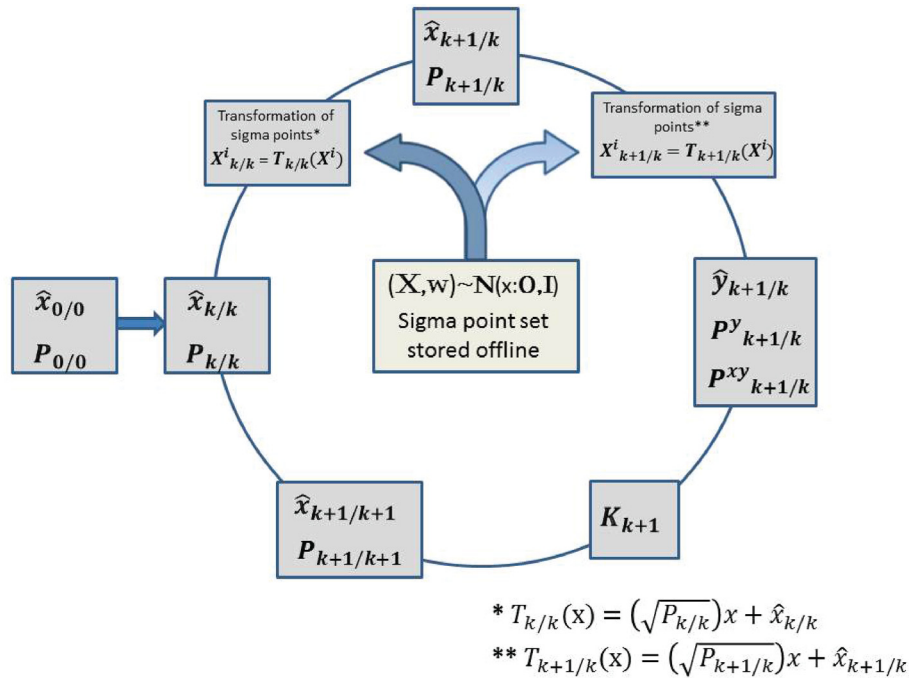$$** \ T_{k+1/k}(\mathrm{x}) = \left(\sqrt{P_{k+1/k}}\right)x + \hat{x}_{k+1/k}$$

Fig. 8   Iterative block schematic of the filtering algorithm

distribution with heavy tail implying lower robustness. On the other hand, the higher-order moments such as the sixth-order moment cost can result in a PDF that is much slimmer, indicating better robustness, but achieves this at the cost of a higher mean indicating lower performance.

5.5 Discussion on Computational Complexity and Limitations of Polynomial System of Equations. The growth of CUT is indeed exponential but with base 2. Notice that this is the minimal exponential growth with integer base that one can have. The proposed CUT points for the dimensions and orders described are much lower (by several factors) than equivalent Gaussian quadrature. Tables 18–20 show the number of points for each method. Further, all the weights are positive. The polynomial system is solved by first solving for the weights symbolically in terms of the other scaling variables r_i and then substituted into the remaining moment constraint polynomial equations. This reduces the number of equations but increases the degree of the others that are a function of r_i. The order can be further reduced by simple substitutions. The resultant system of equations is easier to solve by BERTINI [27] while the original system is often computationally intractable. As one goes to higher dimensions and order, the number of equations increases and the process of symbolically solving for the weights becomes intractable. Hence, the entire polynomial system of equations has to be solved by the polynomial solver, which is often numerically intractable. Further, the nature of the polynomial roots (real and positive) is generally known only after the system is solved. In the case of the uniform PDF, the problem of having all the points within the support of the uniform PDF poses an additional challenge. The axes were judiciously chosen sequentially in order to keep the number of points as low as possible and yet have a computationally feasible solution with all the desired characteristics. Nevertheless, once solved, the points can be saved and directly used. Further, the CUT points are fewer than many standard quadrature rules and offer tremendous potential in application of uncertainty quantification as made evident in numerical examples of Sec. 5. In Ref. [1], the problem of quantifying the uncertainty of the satellite motion was considered. The CUT8-G points proved to be very efficient in computing the higher-order moments when compared to the Gauss–Hermite and Monte Carlo methods. With just 741 points, the CUT points were able to outperform the 200,000 MC points to compute the moments up to fourth-order. In Refs. [2] and [42], the CUT methodology is successfully applied to compute a probabilistic spatial-temporal estimate of ash presence during the April 2010 eruption of the Eyjafjallajökull volcano in Iceland with the help of only 161 simulations of transport and diffusion model.

The examples related to filtering and precision control illustrated in this paper do not include the comparison to other quadratures methods as the required number of points for the same accuracy are presented in Tables 18–20. These tables reveal the relative computational burden of each quadrature method.

6 Conclusions

The curse of dimensionality, which is inherent in the conventional Gauss-quadrature methods, has motivated numerous researchers to develop nonproduct quadrature rules to embed computational efficiency in evaluating multidimensional expectation integrals. Often, these approaches result in negative weights associated with the quadrature points, which have been shown to result in inaccurate evaluation of the expectation integrals. Although negative weights associated with a particular quadrature approach may not be a problem for a specific application, one cannot be guaranteed of their accuracy a priori in evaluating multidimensional expectational integrals. The unscented transformation, a popular second-order quadrature rule for Gaussian random variables, has a linear growth in quadrature points with respect to the dimension of the uncertain space. Motivated by the unscented-transforms and labeled conjugate unscented transformation, this paper outlines a systematic approach to develop nonproduct quadrature points of desired order with positive weights to evaluate expectation integrals involving Gaussian and uniform density functions. Analytical solutions have been developed for fourth-order quadrature rules for all dimensions to evaluate expectation integrals involving Gaussian as well as uniform density functions. For sixth- and eighth-order quadrature rules, although solutions have been developed only for certain dimensions, they offer tremendous advantage in speed and accuracy for uncertainty
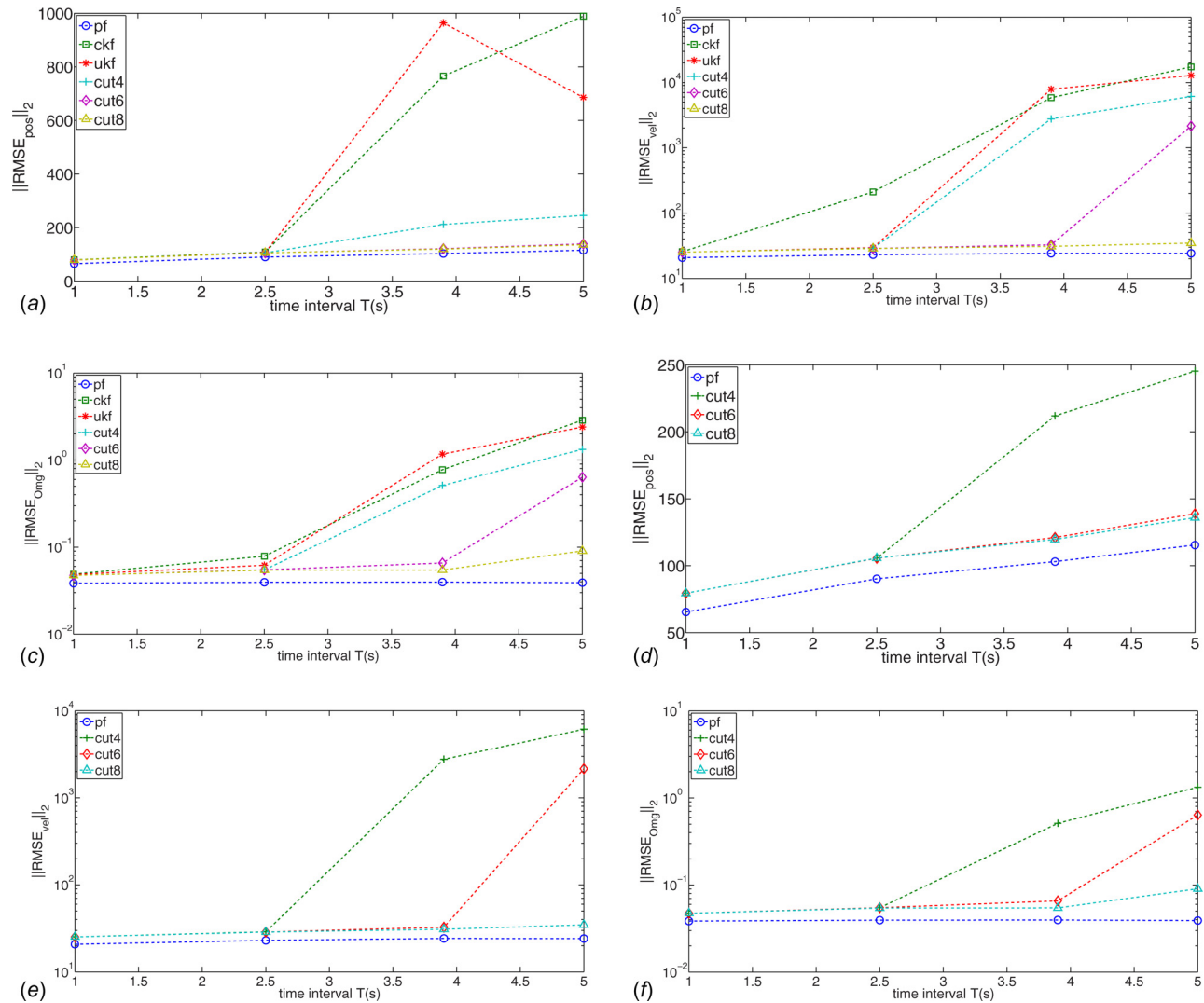
Fig. 9 State estimation error versus the measurement time interval, *T* for the air traffic problem: (*a*) position (all filters), (*b*) velocity (all filters), (*c*) angular rate (all filters), (*d*) position (PF, CUT4, CUT6, CUT8), (*e*) velocity (PF, CUT4, CUT6, CUT8), and (*f*) angular rate (PF, CUT4, CUT6, CUT8)

Table 24 Comparison of 2-norms of RMSE in position, velocity, and turn rate for *T* = 5 s for air traffic problem

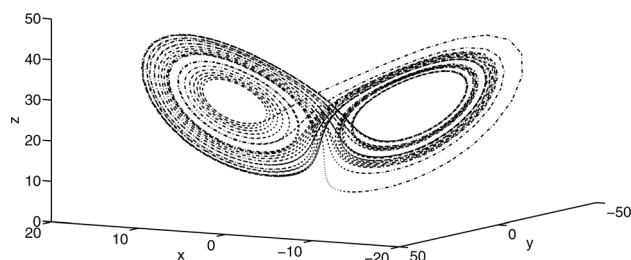| $\|\mathrm{RMSE}\|_2$ in	PF–mean	CKF	UKF	CUT4–G	CUT6–G	CUT8–G
Position	115.47	989.15	685.90	245.30	138.82	135.89
Velocity	24.16	17,330.23	12,849.90	6127.86	2153.53	34.73
Ω	0.0393	2.873	2.396	1.329	0.636	0.090
No. of points	5000	10	11	42	83	355

Fig. 10   Lorenz system

quantification in many engineering applications. The developed CUT quadrature rules provide minimal number of quadrature points with positive weights as compared to widely used quadrature methods (e.g., Gauss quadrature and sparse Gauss quadrature methods). Numerous examples representing static mapping involving polynomial and transcendental functions are used to illustrate the efficacy of the developed methodology and provides the evidence in support of the fact that the proposed CUT methodology provides a significant reduction in function evaluations to compute multidimension expectation integrals in comparison to widely used quadrature methods. Application such as uncertainty quantification, nonlinear filtering, and robust control design are also used to illustrate the potential suite of applications that can benefit from the proposed CUT methodology.
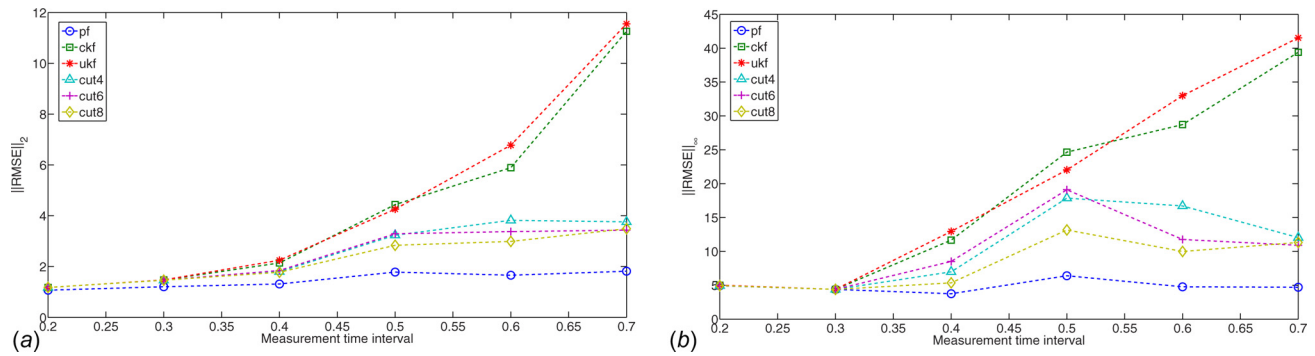
Fig. 11 Comparison of filters for Lorenz model with varying measurement time intervals T: (a) 2-norm of RMSE and (b) max of RMSE

Table 25 Comparison of RMSE for various filters

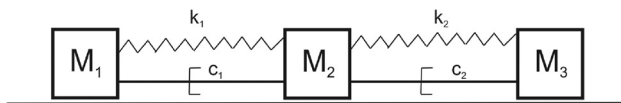|	PF–mean	CKF	UKF	CUT4–G	CUT6–G	CUT8–G
$\|\|RMSE\|\|_2$ with $Q = 0.002I_{5\times5}$	14.7986	0.8945	0.8934	0.8974	0.9004	0.9002
$\|\|RMSE\|\|_2$ with $Q = 0.005I_{5\times5}$	1.0826	6.3419	4.2319	1.9162	1.8156	1.7225
No. of pts	5000	10	11	42	83	355

Fig. 12 Example 2: three mass system: uncertain spring and damping constants

Table 26 2-norm % relative error

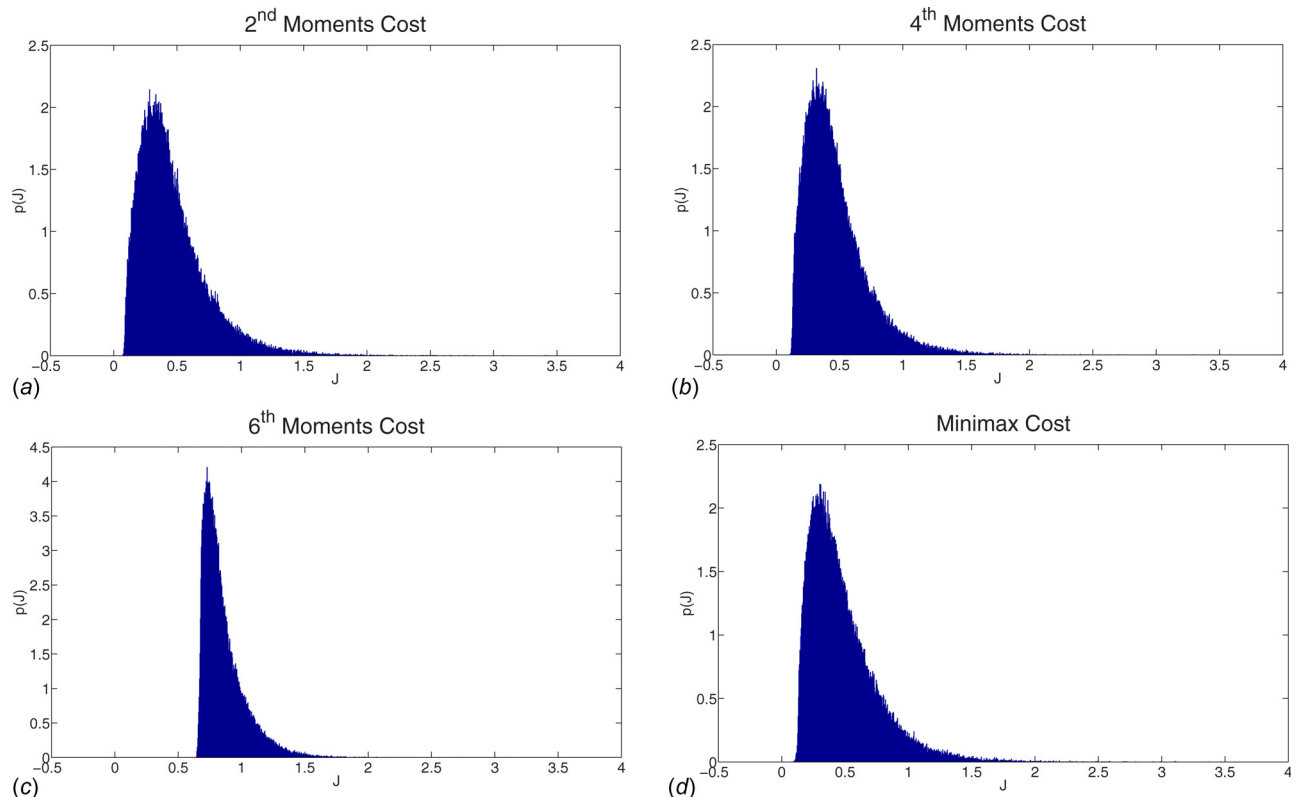|	GLgn3	CUT4-U	GLgn4	CUT6-U	GLgn5	CUT8-U
M_1	0.0176	0.0171	0.0161	0.0160	0.0160	0.0160
M_2	0.1150	0.0216	0.2338	0.2741	0.2576	0.2599
M_3	12.8793	9.9926	1.7471	0.4916	0.9488	0.9122
N	81	24	256	65	625	161

Fig. 13 Distribution of cost function J under parametric uncertainty: (a) up to second moment) cost, (b) up to fourth moment) cost, (c) up to sixth moment) cost, and (d) min–max control cost

Funding Data

Appendix

The solutions of the MCE (36)–(41) and (46)–(51) of order 6 are given in Table 27. The solutions listed are accurate to the tenth decimal point. Solving the MCE accurately to include more decimal places would provide a better result in numerical evaluations. These solutions when used along with Tables 11 and 12 can generate the sigma point sets that are sixth-order accurate for dimensions $2 \leq n \leq 9$.

The solution for the CUT8 sigma points has been computed by first deriving the MCE of order 8 for each dimension individually by using the CUT8 sigma point selection Table 16 and numerically solving them as described in Sec. 4.3. The solutions of CUT8 for each dimension that are given in Table 28 are tenth decimal accurate.

Table 27 Solutions for 2 ≤ n ≤ 9, sixth CUT6-G

Variable	2D	3D	4D	5D	6D	7D	8D	9D
r_1	2.4494897427	2.3587090379	2.2520650012	2.1213203430	1.9488352799	2.5512003554	2.4494897427	2.3439073215
r_2	1.1147379454	1.1198362859	1.1260325006	1.1338934189	1.1445968942	0.9642630979	1	1.0232622230
r_3	3.2004125801	3.1421303838	3.0763780026	3	2.9068006056	2.3255766977	2.449489742	2.5342864499
w_1	0.0277777777	0.0290351301	0.0306601632	0.0329218107	0.0365072564	0.0126940628	0.0138888888	0.0150763910
w_2	0.1302876649	0.0633844605	0.0306601632	0.0147033607	0.0069487173	0.0048594459	0.00234375	0.0011342717
w_3	0.0004653012	0.0005195469	0.0005898367	0.0006858710	0.0008288549	0.0003950899	0.0002314814	0.0001572731

Table 28 Solutions for 2 ≤ n ≤ 6, eighth CUT8-G

Variable	2D	3D	4D	5D	6D
r_1	2.0681360611	2.2551372655	2.2017090714	2.3143708172	2.4494897427
r_2	0.8491938499	0.7174531274	0.7941993714	0.8390942773	0.8938246941221211
r_3	1.1386549808	1.8430194370	1.8725743605	1.8307521253	1.7320508075
r_4	1.8616199350	1.5584810327	1.3291164300	1.3970397430	1.531963037906212
r_5	—	—	2	2	2
r_6	—	1.3055615004	1.1258655812	1.1134786327	1.0954451150
w_1	0.0438226426	0.0246319934	0.0181100873	0.0105290342	0.0061728395
w_2	0.1405096621	0.081510094	0.0320632733	0.0151440196	0.0069134430
w_3	0.0009215768	0.00976723555	0.006614353	0.0052828996	0.0041152263
w_4	0.0124095396	0.0057724893	0.0034899065	0.0010671298	0.0002183265
w_5	—	—	0.0006510416	0.0006510416	0.00065104166
w_6	—	0.0002794729	0.0002521833	0.00013776017	0.00007849171
h	3	2.74	3	3	3

Table 29 6 ≤ n ≤ 8, CUT4-U

	r_1	r_2	w_1	w_2
6D	0.795484480	0.772995860	0.018498622	0.003241735
7D	0.983072689	0.746798459	0.017844575	0.001116333
8D	0.752276560	0.775263910	0.008673360	0.000480594

Table 30 2 ≤ n ≤ 9, sixth CUT6-U

	r_1	r_2	r_3	r_4	w_1	w_2	w_3	w_4
2D	0.9258200997	0.8749414957	0.5332579116	—	0.0604938271	0.0310942224	0.1181727528	—
3D	0.9281932822	0.5908222639	0.9221273153	1	0.0364049422	0.0120481650	0.0618234831	0.002
4D	0.9393949834	0.5908515039	0.9220484019	1	0.0123188100	0.0309025641	0.0120543532	0.001
5D	0.9232975798	0.5814647380	0.9276346108	1	0.0256217782	0.0153805184	0.0029063276	0.0001
6D	0.889484914	0.5896097525	0.937047336	1	0.0170933292	0.0077209324	0.0018236680	0.0001
7D	1	0.5067680528	0.9239364353	0.7488642940	0.0010582010	0.0032539121	0.0014884137	0.001
8D	1	0.5649236231	0.9074852129	0.8958617999	0.0084656084	0.0018753991	0.0003315651	0.00005
9D	1	0.5638796285	0.9191450300	0.8512874568	0.0035273368	0.0009382475	0.0002047437	0.00005

Table 31 Solutions for 2 ≤ n ≤ 5, CUT8-U

	r_1	r_2	r_3	r_4	r_5	r_6	h
2𝔻	0.8094513751	0.4908311733	0.48591160472	0.8565014348	—	—	2
3𝔻	0.7466221822	0.8585800181	0.8611091583	0.4977491909	0.4812805993	—	2
4𝔻	0.9185985004	0.4056290098	0.7897970163	0.9182313590	0.56103196822	0.8770580193	1.7
5𝔻	0.8451542547	0.7381963342	0.9151432251	0.8189442986	0.3940256098	0.5000712983	1.9

	w_1	w_2	w_3	w_4	w_5	w_6
2𝔻	0.0637331008	0.0917208419	0.0170247219	0.0276153989	—	—
3𝔻	0.0394515056	0.0131260421	0.0068117619	0.0348810086	0.0091901236	—
4𝔻	0.0080621257	0.0145953448	0.0130470117	0.0017907843	0.0046995728	0.0006502632
5𝔻	0.0005942913	0.0076242084	0.0013243454	0.0007889833	0.0031006152	0.0024757515

References

[1] Adurthi, N., and Singla, P., 2015, "Conjugate Unscented Transformation-Based Approach for Accurate Conjunction Analysis," J. Guid. Control Dyn., **38**, pp. 1642–1658.

[2] Madankana, R., Pouget, S., Singla, P., Bursik, M., Dehn, J., Jones, M., Patra, A., Pavolonis, M., Pitman, E. B., Singh, T., and Webley, P., 2014, "Computation of Probabilistic Hazard Maps and Source Parameter Estimation for Volcanic Ash Transport and Diffusion," J. Comput. Phys., **271**, pp. 39–59.

[3] Xiu, D., and Karniadakis, G. E., 2002, "The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations," SIAM J. Sci. Comput., **24**(2), pp. 619–644.

[4] Madankan, R., Singla, P., Singh, T., and Scott, P., 2013, "Polynomial Chaos Based Bayesian Approach for State and Parameter Estimation," AIAA J. Guid. Navigation Control, **36**(4), pp. 1058–1074.

[5] Terejanu, G., Singla, P., Singh, T., and Scott, P. D., 2008, "Uncertainty Propagation for Nonlinear Dynamical Systems Using Gaussian Mixture Models," J. Guid. Control Dyn., **31**(6), pp. 1623–1633.

[6] Terejanu, G., Singla, P., Singh, T., and Scott, P. D., 2011, "Adaptive Gaussian Sum Filter for Nonlinear Bayesian Estimation," IEEE Trans. Autom. Control, **56**(9), pp. 2151–2156.

[7] Vishwajeet, K., Singla, P., and Jah, M., 2014, "Nonlinear Uncertainty Propagation for Perturbed Two-Body Orbits," AIAA J. Guid. Control Dyn., **37**(5), pp. 1415–1425.

[8] Stroud, A. H., 1971, Approximate Calculation of Multiple Integrals, Prentice Hall, Englewood Cliffs, NJ, pp. 23–53.

[9] Stroud, A. H., and Secrest, D., 1966, Gaussian Quadrature Formulas, Prentice Hall, Englewood Cliffs, NJ, pp. 17–37.

[10] Gerstner, T., and Griebel, M., 1998, "Numerical Integration Using Sparse Grids," Numer. Algorithms, **18**, pp. 209–232.

[11] Julier, S., Uhlmann, J., and Durrant-Whyte, H., 2000, "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators," IEEE Trans. Autom. Control, **45**(3), pp. 477–482.

[12] McNamee, J., and Stenger, F., 1967, "Construction of Fully Symmetric Numerical Integration Formulas of Fully Symmetric Numerical Integration Formulas," Numer. Math., **10**(4), pp. 327–344.

[13] Wu, Y., Hu, D., Wu, M., and Hu, X., 2006, "A Numerical-Integration Perspective on Gaussian Filters," IEEE Trans. Signal Process., **54**(8), pp. 2910–2921.

[14] Stroud, A. H., 1960, "Numerical Integration Formulas of Degree Two," Math. Comput., **14**(69), pp. 21–26.

[15] Rabinowitz, P., and Richter, N., 1969, "Perfectly Symmetric Two-Dimensional Integration Formulas With Minimal Numbers of Points," Math. Comput., **23**(108), pp. 765–779.

[16] Piessens, R., and Haegemans, A., 1975, "Cubature Formulas of Degree Nine for Symmetric Planar Regions," Math. Comput., **29**(131), pp. 810–815.

[17] Franke, R., 1973, "Minimal Point Cubatures of Precision Seven for Symmetric Planar Regions," SIAM J. Numer. Anal., **10**(5), pp. 849–862.

[18] Cools, R., and Rabinowitz, P., 1993, "Monomial Cubature Rules Since Stroud a Compilation," J. Comput. Appl. Math., **48**(3), pp. 309–326.

[19] Cools, R., 1999, "Monomial Cubature Rules Since Stroud a Compilation—Part 2," J. Comput. Appl. Math., **112**(1–2), pp. 21–27.

[20] Julier, S., and Uhlmann, J., 2002, "Reduced Sigma Point Filters for the Propagation of Means and Covariances Through Nonlinear Transformations," American Control Conference (ACC), Anchorage, AK, May 8–10, pp. 887–892.

[21] Arasaratnam, I., and Haykin, S., 2009, "Cubature Kalman Filter," IEEE Trans. Autom. Control, **54**(6), pp. 1254–1269.

[22] Adurthi, N., Singla, P., and Singh, T., 2012, "The Conjugate Unscented Transform—An Approach to Evaluate Multi-Dimensional Expectation Integrals," American Control Conference (ACC), Montreal, QC, Canada, June 27–29, pp. 5556–5561.

[23] Adurthi, N., Singla, P., and Singh, T., 2013, "Conjugate Unscented Transform Rules for Uniform Probability Density Functions," American Control Conference (ACC), Washington, DC, June 17–19, pp. 2454–2459.

[24] Adurthi, N., 2012, "The Conjugate Unscented Transform—A Method to Evaluate Multidimensional Expectation Integrals," Master's thesis, University at Buffalo, Buffalo, NY.

[25] Tenne, D., and Singh, T., 2003, "The Higher Order Unscented Filter," American Control Conference (ACC), Denver, CO, June 4–6, pp. 2441–2446.

[26] Levesque, J. F., 2006, "Second Order Simplex Sigma Points for Nonlinear Estimation," Guidance, Navigation, and Control Conference, Keystone, CO, Aug. 21–24, pp. 819–830.

[27] Bates, D. J., Hauenstein, J. D., Sommese, A. J., and Wampler, C. W., Numerically solving polynomial systems with Bertini, SIAM, Philadelphia, PA.

[28] Julier, S., and Uhlmann, J., 2004, "Unscented Filtering and Nonlinear Estimation," Proc. IEEE, **92**(3), pp. 401–422.

[29] Patterson, T. C., 1968, "The Optimum Addition of Points to Quadrature Formulae," Math. Comput., **22**(104), pp. 847–856.

[30] Heiss, F., and Winschel, V., 2008, "Likelihood Approximation by Numerical Integration on Sparse Grids," J. Econ., **144**(1), pp. 62–80.

[31] Jia, B., Xin, M., and Cheng, Y., 2013, "High-Degree Cubature Kalman Filter," Automatica, **49**(2), pp. 510–518.

[32] Zhao, Z., Li, X. R., and Jilkov, V. P., 2004, "Best Linear Unbiased Filtering With Nonlinear Measurements for Target Tracking," IEEE Trans. Aerosp. Electron. Syst., **40**(4), pp. 1324–1336.

[33] Jazwinski, A. H., 1970, Stochastic Processes and Filtering Theory, Academic Press, New York.

[34] Gelb, A., 1974, Applied Optimal Estimation, MIT Press, Cambridge, MA.

[35] Bar-Shalom, Y., Kirubarajan, T., and Li, X.-R., 2002, Estimation With Applications to Tracking and Navigation, Wiley, New York.

[36] Ito, K., and Xiong, K., 2000, "Gaussian Filters for Nonlinear Filtering Problems," IEEE Trans. Autom. Control, **45**(5), pp. 910–927.

[37] Jia, B., Xin, M., and Cheng, Y., 2012, "Sparse-Grid Quadrature Nonlinear Filtering," Automatica, **48**(2), pp. 327–341.

[38] Candy, J., 2007, "Bootstrap Particle Filtering," IEEE Signal Process. Mag., **24**(4), pp. 73–85.

[39] Arulampalam, M., Maskell, S., Gordon, N., and Clapp, T., 2002, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," IEEE Trans. Signal Process., **50**(2), pp. 174–188.

[40] Bryson, A., and Mills, R., 1998, "Linear-Quadratic-Gaussian Controllers With Specified Parameter Robustness," J. Guid. Control Dyn., **21**(1), pp. 11–18.

[41] Singh, T., Singla, P., and Konda, U., 2010, "Polynomial Chaos Based Design of Robust Input Shapers," ASME J. Dyn. Syst. Meas. Control, **132**(5), p. 051010.

[42] Bursik, M., Jones, M., Carn, S., Dean, K., Patra, A., Pavolonis, M., Pitman, E. B., Singh, T., Singla, P., Webley, P., Bjornsson, H., and Ripepe, M., 2012, "Estimation and Propagation of Volcanic Source Parameter Uncertainty in an Ash Transport and Dispersal Model: Application to the Eyjafjallajokull Plume of 14–16 April 2010," Bull. Volcanol., **74**(10), pp. 2321–2338.