Sigma Point Transformation for Gaussian Mixture Distributions

Richard Linares,* John L. Crassidis†
University at Buffalo, State University of New York, Amherst, NY, 14260-4400

This paper describes the development of an approximate method for propagating uncertainty through stochastic dynamical systems using a quadrature rule integration based method. The development of quadrature rules for Gaussian mixture distributions is discussed. A numerical solution to this problem is considered that uses a Gram-Schmidt process. Simulation results are presented where the quadrature points are calculated in two different ways, one using an unscented transformation and the other using the method discussed in this work. The proposed method outperforms the unscented transformation and provides signs of optimism for improving nonlinear filtering.

I. Introduction

Accurate propagation of uncertainty through nonlinear dynamical systems is a central problem in many applications in science and engineering. Filtering algorithms for state estimation, such as the extended Kalman Filter[1] (EKF), the Unscented Kalman Filter[2] (UKF) and particle filters[3] are commonly used to both estimate hidden (indirectly observable) states and filter noisy measurements. The basic difference between the EKF and the UKF results from the manner in which the state distribution of the nonlinear models is approximated. The UKF, introduced by Julier and Uhlmann,[2] uses a nonlinear transformation called the unscented transform, in which the state probability density function (pdf) is represented by a set of weighted sigma points (state vectors deterministically sampled about a mean). These are used to parameterize the true mean and covariance of the state distribution.

When the sigma points are propagated through the nonlinear system, the posterior mean and covariance are obtained up to second order for any nonlinearity. The EKF and UKF assume that the process noise terms are represented by zero-mean Gaussian white-noise processes and the measurement noise is also represented by a zero-mean Gaussian random variable. Furthermore both approaches assume that the *a posteriori* density function is Gaussian in a linear domain. This is true given the previous assumptions but under the effect of nonlinear measurement functions and system dynamics the initial Gaussian state uncertainty may quickly become non-Gaussian.

Both filters only provide approximate solutions to the nonlinear filtering problem, since the *a posteriori* density function is most often non-Gaussian due to nonlinear effects. The EKF typically works well only in the region where the first-order Taylor-series linearization adequately approximates the non-Gaussian pdf. The UKF provides higher-order moments for the computation of the *a posteriori* pdf without the need to calculate Jacobian matrices as required in the EKF. This work seeks to make use of similar mathematical technique to the unscented kalman filter to establish concepts for propagating uncertainty through nonlinear and non-Gaussian stochastic systems. This work will study the problem of developing an unscented-like transformation which can generate sigma points with respect to a Gaussian mixture probability distribution. The technique behind the unscented transformation is related to Gaussian quadrature integration[4] and will be used in this work to extend the unscented transformation to Gaussian mixture distributions.

The problem studied here involves determining the following functional:

$$Q[f] = \int f(x)w(x)dx \tag{1}$$

*Graduate Student, Department of Mechanical & Aerospace Engineering. Email: linares2@buffalo.edu, Student Member AIAA.

†Professor, Department of Mechanical & Aerospace Engineering. Email: johnc@buffalo.edu, Associate Fellow AIAA.

where it is desired to approximate this functional to help describe the probability space of x. Consider the following dynamic system:

$$x_{k+1} = f_{k+1}(x_k) \tag{2}$$

where x_k describes the state of the system at some time t_k, k denoting the time index in x. To completely describe the probability space of x_{k+1} it would be sufficient to determine

$$\mu_n = \int x_{k+1}^n w_{k+1}(x_{k+1}) dx_{k+1} = \int f_{k+1}(x_k) w_k(x_k) dx_k \tag{3}$$

where $n = 1, \ldots, \infty$, w_k and w_{k+1} are the probability density functions at t_k and t_{k+1} respectively. Therefore computing μ_n of $f_{k+1}(x_k)$ with respect to w_k provides the probability space for x_{k+1}. If it is possible to calculate μ_n for all n then an exact representation of the probability law can be found using the insufficient reason principle.[5] The issue with this approach is that the integral in Eq. (3) is not generally solvable, but can be approximated using approximation theory.[6] One approximation, is the well known unscented transformation[2,7–9] in which the integral is approximated by a weight sum of nodes. This approximation is remarkably accurate and has the benefit of giving second order level of accuracy. The UKF concerns itself with just the mean and covariance, modeling the pdf as a Gaussian. The accuracy for this transformation is shown using a Taylor series expansion and it is shown to be accurate up to second order in terms of the covariance calculation.[9] The unscented transformation is based on Gaussian quadrature integration and this theory allows one to approximate integrals with respect to a Gaussian distribution.[4]

Recent work on polynomial chaos expansions has provided accurate solutions to stochastic process estimation problems where a polynomial expansion is considered that is orthogonal with respect to the underlying probability measure.[10] It has been shown that the error in the approximation converges exponentially to the true stochastic process with increasing order of expansion if the orthogonal expansion is developed with respect the underlining probability measure. This result is not unrelated to the theory behind quadrature integration. The integration operator, like any other operator, can be decomposed into its Fourier components and the action of the operator on functions can be represented by the action of the Fourier components on functions. The convergence of this decomposition to the true operator is exponential and if the function has nonlinearity of polynomial order, then the Fourier decomposition converges to the true operator with finite terms.[11]

II. Orthogonal Polynomials

This section discusses orthogonal polynomials where a complete treatment on the subject can be found in reference 12. A collection of functions $\{p_n(x)\}_{n=1}^{\infty}$ is said to form a orthogonal set if and only if

$$\langle p_i(x), p_j(x) \rangle = \begin{cases} \tau_i & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{4}$$

where τ_i is a bounded constant and if $\tau_i = 1$ for all i then the functions $\{p_n(x)\}_{n=1}^{\infty}$ form an orthonormal set. Note that $\langle \cdot, \cdot \rangle$ denotes an inner product. The inner product for continuous variables can be written as

$$\langle p_i(x), p_j(x) \rangle = \int_a^b p_i(x) p_j(x) w(x) dx \tag{5}$$

where $w(x)$ is a weighting function. For the case of uncertainty propagation $w(x)$ is a probability distribution. Then the orthogonal decomposition property of function spaces allows us to write any element in the function space by linear combinations of the basis functions that form a maximal orthonormal set of the function space. So any arbitrary element of the function space can be written as

$$f(x) = \sum_{k=1}^{\infty} a_k p_k(x) \tag{6}$$

To form an exact representation the sum must be taken to infinity. By truncating the sum at a finite number n an approximation for $f(x)$ can be formed and the series forms a basis for a subspace of the full function space.

There are a number of collections of functions that form an orthogonal basis but a particular useful set are orthogonal polynomials. Consider a finite dimensional space, an n^{th} order orthogonal polynomial $p_n(x)$ is orthogonal with respect to the subspace

$$\mathbf{P}_{n-1} = \text{span}\{1, x, ..., x^{n-1}\} \tag{7}$$

of all polynomials of degree less than n. Note that $p_n(x)$ is orthogonal to $\mathbf{P}_{n-1}$ with respect to the weighting function $w(x)$.

A Gram-Schmidt orthogonalization can be applied to form an orthonormal basis with respect to an inner product. The Gram-Schmidt process is general and the same recursive relationship can be written for both continuous and discrete inner products. Once the inner product has been defined the Gram-Schmidt process can be written as

$$p_{n+1}(x) = \left[x^n - \sum_{j=1}^{n-1} \frac{\langle x, p_j(x) \rangle}{\langle p_j(x), p_j(x) \rangle} p_j(x) \right] \tag{8}$$

The Gram-Schmidt process gives a recursive mechanism to generate orthogonal polynomials with respect to a predestined weighting function $w(x)$.

Given that an orthogonal basis for any inner product space can be constructed, we can now form an approximation of the function f by a set of orthogonal polynomials generated with respect the the inner product $\langle \cdot, \cdot \rangle$ up to a resolution n, where here n is the maximum degree of the orthogonal polynomials used in the basis. An approximation of a function $f(x)$ can be written as

$$\hat{f}(x) = \sum_{k=1}^{n} a_k p_k(x) \tag{9}$$

The constants a_k can be found by minimizing the error $\hat{f}(x) - f(x)$. Then using the projection property the approximation that minimizes the error can be written as

$$\hat{f}(x) = \sum_{k=1}^{n} \langle f(x), p_k(x) \rangle p_k(x) \tag{10}$$

The set of orthogonal polynomials with respect to a weighting function $w(x)$ form an eigenspace for integration with respect to $w(x)$ and therefore are the best approximation of $f(x)$. In the next section an algorithm for calculating the roots of the n^{th} order orthogonal polynomial is discussed. These roots are then used as the quadrature points and numerical results are shown for approximating an integral using these quadrature points for a Gaussian mixture model.

The orthogonal polynomials can be calculated using the moment relationship where the n^{th} order orthogonal polynomial is calculated as

$$p_k(x) = c_n \det \begin{vmatrix} \mu_0 & \mu_1 & \dots & \mu_n \\ \mu_1 & \mu_2 & \dots & \mu_{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{n-1} & \mu_n & \dots & \mu_{2n+1} \\ 1 & x & \dots & x^n \end{vmatrix} \tag{11}$$

where c_n is a normalization constant and μ_i are given by

$$\mu_i = \int_a^b x^i w(x) dx \tag{12}$$

III. Unscented Transformation

The UKF, introduced by Julier and Uhlmann,[8] uses a nonlinear transformation called the scaled unscented transformation, in which the state probability distribution is represented by a set of weighted sigma

points, which are used to parameterize the true mean and covariance of the state distribution. When the sigma points are propagated through the nonlinear system, the posterior mean and covariance is obtained up to second order for any nonlinearity.

Consider the following nonlinear system and measurement model:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{z}_k \tag{13a}$$

$$\tilde{\mathbf{y}}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \tag{13b}$$

where $\mathbf{z}_k$ and $\mathbf{v}_k$ are zero-mean Gaussian noise processes with covariances $\mathbf{Q}$ and $\mathbf{R}$, respectively. The state vector is redefined in the UKF approach by augmenting the state vector to include noise variables, where the augmented state vector is defined by $\mathbf{x}_k^a = [\mathbf{x}_k^T \ \mathbf{z}_k^T \ \mathbf{v}_k^T]^T$ and the augmented state vector has dimension $N_a = N + q + l$. All random variables in the UKF are assumed to be Gaussian random variables. Therefore one can think of a joint distribution for the random variables, equivalent to the distribution of $\mathbf{x}_k^a$, defining a multivariate Gaussian distribution given by $w(\mathbf{x}_k^a) = w(\mathbf{x}_k, \mathbf{z}_k, \mathbf{v}_k)$. Then the joint distribution is approximated by $w(\mathbf{x}_k, \mathbf{z}_k, \mathbf{v}_k) \sim \mathcal{N}(\mathbf{x}_k^a, \mathbf{P}^a)$. The mean augmented vector $\mathbf{x}_k^a$ can written as $\mu^a = [\boldsymbol{\mu}^T \ \mathbf{0}_{l\times 1}^T \ \mathbf{0}_{q\times 1}^T]^T$, where $\boldsymbol{\mu}$ is the state estimate. The covariance matrix, $\mathbf{P}^a$, for the joint distribution can be written as

$$\mathbf{P}^a = \begin{bmatrix} \mathbf{P} & \mathbf{P}^{x\,z} & \mathbf{P}^{x\,v} \\ \mathbf{P}^{z\,x} & \mathbf{Q} & \mathbf{P}^{z\,v} \\ \mathbf{P}^{v\,x} & \mathbf{P}^{v\,z} & \mathbf{R} \end{bmatrix} \tag{14}$$

Then the distribution is approximated as a set of symmetric selected scaled sigma points. The sigma points are selected such that they are zero-mean, but if the distribution has mean $\boldsymbol{\mu}$, then simply adding $\boldsymbol{\mu}$ to each of the points yields a symmetric set of $2N_a$ points having the same covariance as the initial Gaussian distribution.[8] The sigma points are selected to be along the principal axis direction of the Gaussian distribution $w(\mathbf{x}_k^a)$ or along the eigenvector directions of $\mathbf{P}_k^a$. Then the augmented state vector and covariance matrix is constructed by using the following sigma points

$$\boldsymbol{\sigma}_k \leftarrow 2N_a \text{ columns from } \sqrt{(n_a + \lambda)\mathbf{P}_k^a} \tag{15a}$$

$$\boldsymbol{\chi}_k^a(0) = \boldsymbol{\mu}_k \tag{15b}$$

$$\boldsymbol{\chi}_k^a(i) = \boldsymbol{\sigma}_k(i) + \boldsymbol{\mu}_k \tag{15c}$$

Therefore given an $N_a \times N_a$ covariance matrix $\mathbf{P}_k^a$, a set of $2N_a$ sigma points can be generated from the columns of the matrices $\sqrt{(N_a + \lambda)\mathbf{P}_k^a}$, where $\sqrt{M}$ is shorthand notation for a matrix Z such that $M = Z\,Z^T$. Using the notation of the augmented state vector the sigma point vector can be written as

$$\boldsymbol{\chi}_k^a(i) = \begin{bmatrix} \boldsymbol{\chi}^x(i) \\ \boldsymbol{\chi}^z(i) \\ \boldsymbol{\chi}^v(i) \end{bmatrix} \tag{16}$$

Then, given that these points are selected to represent the distribution of the augmented state vector, each sigma point is given a weight that preserves the information contained in the initial distribution:

$$W_0^{mean} = \frac{\lambda}{N + \lambda} \tag{17a}$$

$$W_0^{cov} = \frac{\lambda}{N + \lambda} + (1 - \gamma^2 + \xi) \tag{17b}$$

$$W_i^{mean} = W_i^{cov} = \frac{\lambda}{2(N + \lambda)} \tag{17c}$$

where $\lambda = \gamma^2 N_a + \kappa - N_a$ includes scaling parameters. The constant parameter controls the size of the sigma point distribution and should be a small number $0 \leq \gamma \leq 1$, and κ provides an extra degree of freedom that is used to fine-tune the higher-order moments; $\kappa = 3 - N_a$ for a Gaussian distribution, also ξ is a third parameter that further incorporates higher-order effects by adding the weighting of the zeroth sigma point to the calculation of the covariance; note $\xi = 2$ is the optimal value for Gaussian distributions.

This paper will focus on the one dimensional unscented transformation, where $\lambda = 2$ and only uncertainty in the state variable is considered. For this case the sigma points are given by

$$\chi_k(0) = \bar{x} \tag{18a}$$

$$\chi(1) = \bar{x} - \sqrt{3}\sigma \tag{18b}$$

$$\chi(2) = \bar{x} + \sqrt{3}\sigma \tag{18c}$$

here $\bar{x}$ is the mean of the initial one dimensional Gaussian distribution.

IV. Quadrature Rule Integration

Gauss quadrature rules form an approximation of integrals with respect to a given weighting function. For example consider the following integral of $f(x)$ over the interval (x_l, x_u):

$$I(f) = \int_{x_l}^{x_u} w(x) f(x) dx \tag{19}$$

where $w(x)$ is a positive weighting function. Then a n-point numerical quadrature integration approximation of the integral in Eq. 19 can be written in the following form

$$I(f) \approx \sum_{i=1}^{n} w_i f(x_i) \tag{20}$$

Here, the x_i are the quadrature points and w_i are the associated weights. Given n distinct quadrature points, the weights w_i can be determined by first computing the moments μ_i using the following expression

$$\mu_i = \int_{x_l}^{x_u} x^i w(x) dx \tag{21}$$

and then solving the following Vandermonde system of equations:[13]

$$\begin{pmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \\ \vdots & \vdots & \ddots & \vdots \\ x_1^n & x_2^n & \dots & x_n^n \end{pmatrix} \begin{pmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_n \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{pmatrix} \tag{22}$$

The system of equations in Eq. 22 is called the system of moment equations. Orthogonal polynomials have the property that they follow a three-term recurrence relationship by[14]

$$x p_n(x) = \alpha_n p_{n+1}(x) - \beta_n p_n(x) + \gamma_n p_{n-1}(x) \tag{23}$$

This equation differs from the traditional form of the three-term recurrence since we have solved for the term $x p_n(x)$. This relationship can be written out for all polynomials in the series

$$\begin{aligned} x p_0(x) &= \alpha_1 p_1(x) - \beta_1 p_0(x) \\ x p_1(x) &= \alpha_1 p_2(x) - \beta_1 p_1(x) - \gamma_1 p_0(x) \\ x p_2(x) &= \alpha_2 p_3(x) - \beta_2 p_2(x) - \gamma_2 p_1(x) \\ &\vdots \end{aligned} \tag{24}$$

This series of equations can be written in matrix equation form

$$x \begin{bmatrix} p_0(x) \\ p_1(x) \\ \vdots \\ p_{n-2}(x) \\ p_{n-1}(x) \end{bmatrix} = \begin{bmatrix} \beta_1 & \alpha_1 & 0 & \dots & 0 \\ \gamma_2 & \beta_2 & \alpha_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \beta_{n-2} & \alpha_{n-2} \\ 0 & 0 & \dots & \gamma_{n-1} & \beta_{n-1} \end{bmatrix} \begin{bmatrix} p_0(x) \\ p_1(x) \\ \vdots \\ p_{n-2}(x) \\ p_{n-1}(x) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ p_n(x) \end{bmatrix} \tag{25}$$

Then the three-term recursion in vector matrix form can be written as

$$x\mathbf{p}(x) = \mathbf{J}\mathbf{p}(x) + \mathbf{e}p_n(x) \tag{26}$$

where J is a tri-diagonal matrix or the Jacobi matrix, $\mathbf{e} = [0, \ldots, 1]$, and the vector $\mathbf{p}(x)$ is given by $\mathbf{p}(x) = [p_0(x), \ldots, p_{n-1}(x)]$. If the polynomials are orthogonal then the Jacobi matrix is systematic, we have $\gamma_n = \alpha_{n-1}$. If $\mathbf{J}$ is not symmetric, then we may perform a diagonal similarity transformation, yielding a symmetric Jacobi matrix $\mathbf{J}$.[12] This transformation yields the following expression

$$\mathbf{J} = \begin{bmatrix} a_0 & \sqrt{b_1} & 0 & \ldots & 0 \\ \sqrt{b_1} & a_1 & \sqrt{b_2} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & a_{n-2} & \sqrt{b_{n-1}} \\ 0 & 0 & \ldots & \sqrt{b_{n-1}} & a_{n-1} \end{bmatrix} \tag{27}$$

where a_n and b_n are related to β_n, α_n, and γ_n. The coefficients a_n and b_n are also related to the following three-term relationship

$$p_{n+1}(x) = (x - a_n)p_n(x) - b_n p_{n-1}(x) \tag{28}$$

where the equation above assumes that the leading coefficients are equal to one. To determine the optimal n-point Gauss quadrature rule for the weighting function that corresponds to the orthogonal series above we must first find the roots of the polynomial p_n. This can be done by setting $p_n(x_i) = 0$ for roots x_i and then solving for the x_i. Therefore we set $p_n(x_i) = 0$ and set all $x = x_i$ in Eq. 26 resulting in the following equation

$$x_i\mathbf{p}(x_i) = \mathbf{J}\mathbf{p}(x_i) \tag{29}$$

This is an eigenvalue/eigenvector problem where the roots x_i are the eigenvalues of $\mathbf{J}$ and $\mathbf{p}(x_i)$ are the eigenvectors of $\mathbf{J}$. For classical orthogonal polynomials, such as Hermite, Laguerre, and Jacobi, there exists a simple three-term recursion relationships. For more complex weighting functions this may not be the case and therefore the Gram-Schmidt process is used in this paper for Gaussian mixture weighting functions. Orthogonal polynomials can be formed using the Gram-Schmidt process shown previously. For a single Gaussian weighting function a closed-form expression for the three-term relation relationship exists and therefore $\mathbf{J}$ is given in terms of Gaussian parameters μ and σ but in the case for Gaussian mixtures the three-term relation relationship is not straightforward.

The terms a_i and b_i can be found from the following relationships:

$$b_i = \frac{\langle xp_i(x), p_i(x)\rangle}{\langle p_i(x), p_i(x)\rangle} \tag{30}$$

$$a_i = \frac{\langle p_i(x), p_i(x)\rangle}{\langle p_{i-1}(x), p_{i-1}(x)\rangle} \tag{31}$$

$$b_o = \langle p_0(x), p_0(x)\rangle \tag{32}$$

These expressions are used to determine a_i and b_i, which are then used to compute $\mathbf{J}$ using Eq. 27. Following this a eigenvalue/eigenvector decomposition is performed on $\mathbf{J}$. The nodes are selected to be the eigenvalues. Then the weights associated with x_i can be determined using the following relationship:[12]

$$w_i\mathbf{p}(x_i)^T\mathbf{p}(x_i) = 1 \tag{33}$$

For orthogonal polynomials the weights are given by

$$w_i = \mathbf{p}(x_i)\{1\}\int_a^b w(x)dx \tag{34}$$

where $\mathbf{p}(x)\{j\}$ denotes the j^{th} component of the vector $\mathbf{p}(x) = [p_0(x), \ldots, p_{n-1}(x)]$. In the case discussed in this paper $w(x)$ is a pdf, $\int_a^b w(x)dx = 1$, therefore the weights are simply $w_i = \alpha_0\mathbf{p}(x_i)\{1\} = \mathbf{p}(x_i)\{1\}$.

In the Gauss-Hermite quadrature rule, the weighting function is chosen to be the Gaussian density with zero mean and unit variance $\mathcal{N}(x;0,1)$. The interval of integration is chosen to be $(-\infty,\infty)$. In the fundamental theorem of Gauss-Hermite quadrature, the quadrature points are chosen to be the zeros of the n^{th} order Hermite polynomial. The zeros of the Hermite polynomials are distinct. Hence the solution vector $\{w_1,...,w_n\}$ is unique. The Hermit polynomials are chosen since they form the orthogonal set for the Gaussian distribution. Since $2n$ unknown parameters are estimated in the Gauss-Hermite quadrature rule (n quadrature points and n weights), the resulting quadrature rule is exact for all polynomials of degree $\leq (2n-1)$. In fact a quadrature rule using the general procedure discussed in this section will be exact for all polynomials of degree $\leq (2n-1)$. Therefore a n^{th} order quadrature rule for Gaussian mixture distributions formed using eigenvalues/eigenvectors of $\mathbf{J}$ is exact for all polynomials of degree $\leq (2n-1)$.

V. Quadrature Point Calculation

In this section quadrature points are calculated using the methods described in section §IV. Four different cases are considered: one, two, four and ten Gaussian components. The components are randomly generated and for each case the quadrature points are found using the methods discussed in section §IV. The Unscented Transformation (UT) is formed for each component (3 points per component) and the Gaussian Mixture Quadrature (GMQ) is formed for the combined distribution in each case using 10 points for all cases. It is seen from Figure 1(a) that for the one Gaussian component case the UT and GMQ approach are both symmetric but since the GMQ has more points (10 points for GMQ vs 3 point for UT) it covers more of the tail of the distribution.

For the case with two components, Figure 1(b), it is seen that the UT approach is centered around each component and is symmetric about each component. On the other hand the GMQ approach is taking into account the overlapping of component within distribution and spreading the points out more and covering more of the tail of the distribution. For the four component case, Figure 1(c), we can start to see that the UT approach is beginning to appear unstructured but the GMQ approach still takes into account the overlapping information between components. Also, the UT has 12 total points (3 for each component) but the GMQ approach only has 10 points and is more spread out covering more of the tail. Finally the case with 10 components, Figure 1(b), really shows the power of the GMQ approach. Here the UT is very unstructured but the GMQ approach still shows an optimal placement of points.

Figure V shows two Gaussian components that are slowly moved closer to each other over the index k. The individual UT points for each distribution are shown in red and blue and the GMQ approach is in black. It can be seen when the components are far apart the GMQ approach coincides with the UT since the distribution are not overlapping and do not share much information. When the components become closer the GMQ approach spreads out more to provide a more optimal coverage.

VI. Numerical Integration Example

In this section the Gaussian mixture quadrature is tested with polynomial integrand functions. The performance of the proposed approach is shown for three represented functions. A simulation scenario is considered where a Gaussian mixture model is used, given by

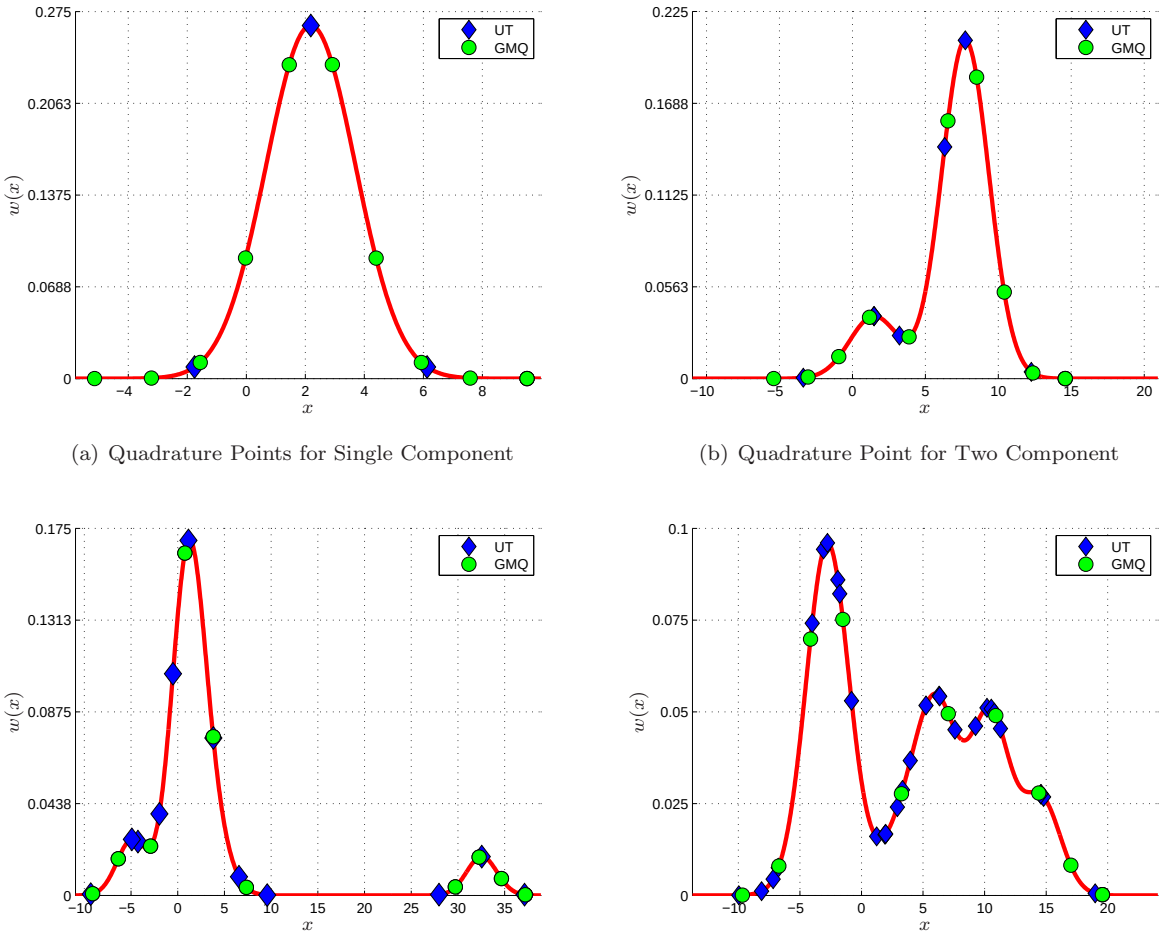
$$w(x) = 0.25w_1(x) + 0.75w_2(x) \tag{35}$$

where $w_i(x)$ is a Gaussian Kernels given by

$$w_i(x) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}} \tag{36}$$

where $\sigma_1 = \sigma_2 = 0.5$, $\mu_1 = -0.7$ and $\mu_2 = 0.7$. The roots of each Gaussian component up to $n = 3$ are calculated which coincide with the unscented transformation. Therefore the unscented transformation for 1-D weighting functions are equivalent to the Gaussian quadrature for up to $n = 3$. For the case of the Gaussian mixture distribution from Eq. (35), the Gaussian quadrature points are found from the polynomial roots of the n^{th} order orthogonal polynomial with respect to the mixture $w(x)$.

The GMQ is calculated for the case where $n = 6$ and therefore the integral should be exact for all 6^{th}



(a) Quadrature Points for Single Component

(b) Quadrature Point for Two Component

(c) Quadrature Points for Four Component

(d) Quadrature Points for Ten Component

Figure 1. Quadrature Point Calculation

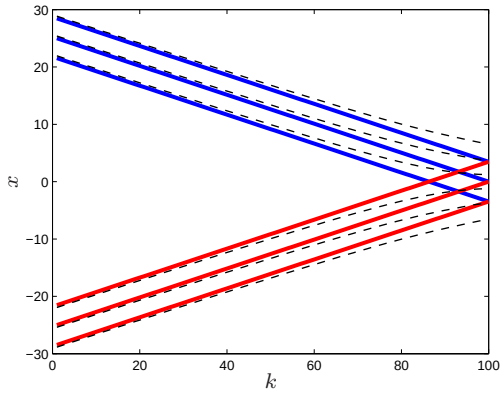


Figure 2. Trajectory of Components and Gaussian Mixture Quadrature

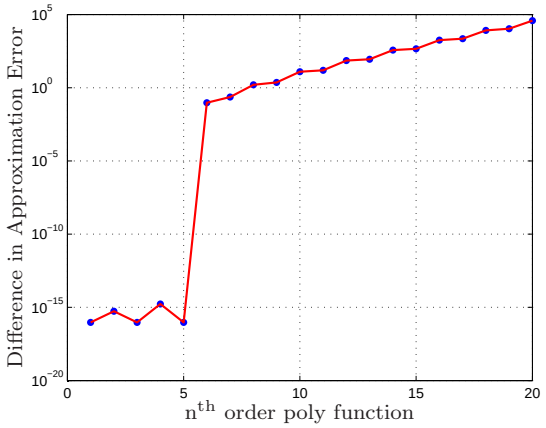
order polynomial functions. The test scenario chosen is given by

$$I(x^p) = \int_{-\infty}^{\infty} x^p w(x) dx \tag{37}$$

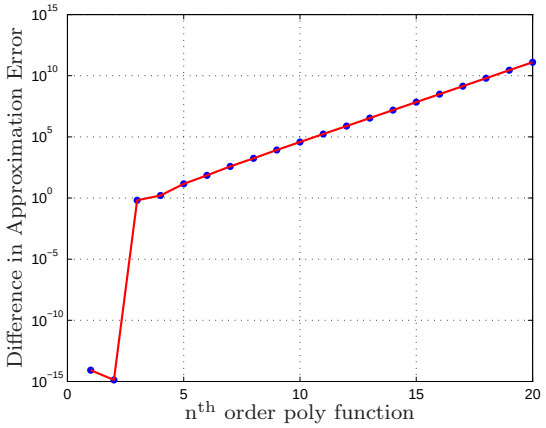
This integral is the first moment of the function $f(x) = x^p$ and the second moment of $f(x)$ is calculated as

$$I(x^p) = \int_{-\infty}^{\infty} \left(x^p - I(x^p)\right)^2 w(x) dx \tag{38}$$

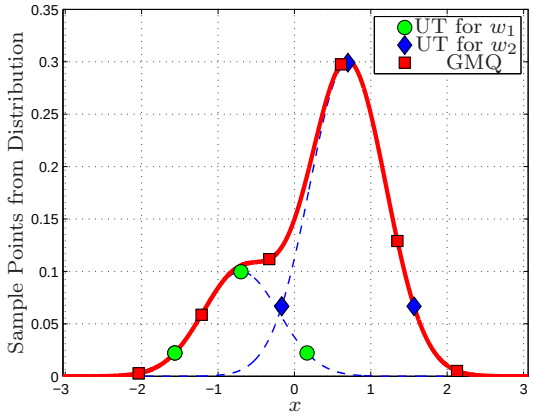
where $I(x^p)$ is the first moment calculated using the expression in Eq. 37.



(a) Error Difference in Mean as a Function of Order of Polynomial



(b) Error Difference in Variance as a Function of Order of Polynomial



(c) Quadrature Points

Figure 3. Numerical Integration Example Simulation Results

The results for the integration example are shown in Figures 3(a), 3(c), and 3(b). Figure 3(a) shows the difference between the error in calculating Eq. 37 using the UT and the error in calculating Eq. 37 using the GMQ, i.e. (UT True) − (GMQ True). Figure 3(c) shows the difference between the error in calculating Eq. 38 using the UT and the error in calculating Eq. 38 using the GMQ. Figure 3(b) shows the UT points and the GMQ points. From Figure 3(b) it is seen that the UT does not account for the shared information between the two Gaussian elements and therefore will not provide the maximum accuracy for the given number of points (6 points, UT uses 3 points for each Gaussian). The GMQ approach will provide a locally and globally optimal approximation and therefore guarantees a higher order accuracy than that of the UT approach for the same number of points.

In Figure 3(a) and 3(c) the difference between the UT and the GMQ is shown as a function of the order of the integrand. The integrands are chosen to be polynomials of varying order from 1^{st} order to 20^{th} order. In Figure 3(a) it is clear that for orders less than 5 both methods result in an exact integration with practically zero error (machine precision) but for higher orders the GMQ method gives a better approximation. In

Figure 3(c) both methods are exact for orders less than 2 but in this case the GMQ still out preforms the UT. The UT uses 3 points for each component and is exact for all polynomials of degree $\leq (2n-1) = 5$ and therefore for the first moment it is accurate up to 5 order functions and for the second moment it is accurate up to 2 order functions (since the second moment involves an additional square term besides the $f(x)$ function's order). This is seen is both Figures 3(a) and 3(c).

VII. Conclusion

Quadrature rule integration was discussed and a general method for calculating the quadrature point was outlined. This method was applied to a numerical example where a Gaussian mixture model was considered and two integration rules were compared. The first approach was based on performing the unscented transformation for each Gaussian component to compute the integral. The second approach was based on the methods discussed in the paper which develop sigma points with respect to the Gaussian mixture. It was shown that the method discussed in the paper outperformed the unscented transformation and provides a means for developing an accurate sigma point transformation for Gaussian mixture models. The method discussed in this paper may be useful for nonlinear filtering problems.

References

[1]Jazwinski, A. H., *Stochastic Processes and Filtering Theory*, chap. 8, Academic Press, San Diego, CA, 1970.

[2]Julier, S. J., Uhlmann, J. K., and Durrant-Whyte, H. F., "A New Approach for Filtering Nonlinear Systems," *Proceedings of the American Control Conference*, Seattle, WA, June 1995, pp. 1628–1632.

[3]Gordon, N. J., Salmond, D. J., and Smith, A. F. M., "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," *IEE Proceedings, Part F - Communications, Radar and Signal Processing*, Vol. 140, No. 2, Seattle, WA, April 1993, pp. 107–113.

[4]Ito, K. and Xiong, K., "Gaussian Filters for Nonlinear Filtering Problems," *IEEE Transactions on Automatic Control*, Vol. 45, No. 5, 2000, pp. 910–927.

[5]Papoulis, A., *Probability, Random Variables, and Stochastic Processes*, 3rd ed., 1991.

[6]Stroud, A. H., *Approximate Calculation of Multiple Integrals*, Englewood Chiffs, New Jersey: Prentice-Hall, 1971.

[7]Julier, S. J. and Uhlmann, J. K., "A New Extension of the Kalman Filter to Nonlinear Systems," *Proceedings of the SPIE, Volume 3068, Signal Processing, Sensor Fusion, and Target Recognition VI*, edited by I. Kadar, Orlando, Florida, April 1997, pp. 182–193.

[8]Julier, S. J., Uhlmann, J. K., and Durrant-Whyte, H. F., "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators," *IEEE Transactions on Automatic Control*, Vol. AC-45, No. 3, March 2000, pp. 477–482.

[9]Julier, S. J., "The Scaled Unscented Transformation," *Proceedings of the American Control Conference*, Anchorage, AK, May 2002, pp. 1108–1114.

[10]Debusschere, B. J., Najm, H. N., Pébay, P. P., Knio, O. M., Ghanem, R. G., and Maître, O. P. L., "Numerical Challenges in the Use of Polynomial Chaos Representations for Stochastic Processes," *SIAM J. Sci. Comput.*, Vol. 26, No. 2, 2005, pp. 698–719.

[11]Naylor, A. and Sell, G., *Linear Operator Theory in Engineering and Science*, Applied Mathematical Sciences, Springer, 2000.

[12]Szegő, G., *Orthogonal polynomials*, No. v. 23, Colloquium Publications - American Mathematical Society, 1939.

[13]Golub, G. and Van Loan, C., *Matrix Computations*, Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, 1996.

[14]Mastroianni, G. and Milovanović, G. V., *Interpolation Processes. Basic Theory and Applications.*, Springer Monographs in Mathematics, Berlin: Springer, 2008.