JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 31, No. 6, November–December 2008

Uncertainty Propagation for Nonlinear Dynamic Systems Using Gaussian Mixture Models

Gabriel Terejanu,* Puneet Singla,† Tarunraj Singh,‡ and Peter D. Scott§
University at Buffalo, State University of New York, Buffalo, New York 14260

DOI: 10.2514/1.36247

A Gaussian-mixture-model approach is proposed for accurate uncertainty propagation through a general nonlinear system. The transition probability density function is approximated by a finite sum of Gaussian density functions for which the parameters (mean and covariance) are propagated using linear propagation theory. Two different approaches are introduced to update the weights of different components of a Gaussian-mixture model for uncertainty propagation through nonlinear system. The first method updates the weights such that they minimize the integral square difference between the true forecast probability density function and its Gaussian-sum approximation. The second method uses the Fokker–Planck–Kolmogorov equation error as feedback to adapt for the amplitude of different Gaussian components while solving a quadratic programming problem. The proposed methods are applied to a variety of problems in the open literature and are argued to be an excellent candidate for higher-dimensional uncertainty-propagation problems.

Introduction

THE state of a stochastic dynamic system $\mathbf{x}$ is characterized by its time-dependent probability density function (pdf) $p(t, \mathbf{x})$. The random nature of the state may be due to the uncertain initial conditions and/or due to the random excitation that is driving the dynamic system. The knowledge about the time evolution of the pdf of a state of a dynamic system is important to quantify the uncertainty in the state at a future time. Numerous fields of science and engineering present the problem of pdf evolution through nonlinear dynamic systems with stochastic excitation and uncertain initial conditions [1,2]. This is a difficult problem and has received much attention over the past century.

One may be interested in the determination of the response of engineering structures [such as beams, plates, entire buildings beams under random excitation (in structure mechanics [3])], the propagation of initial condition uncertainty of an asteroid for the determination of its probability of collision with a planet (in astrodynamics [4]), the motion of particles under the influence of stochastic force fields (in particle physics [5]), or simply the computation of the prediction step in the design of a Bayes filter (in filtering theory [6]). All these applications require the study of the time evolution of the pdf $p(t, \mathbf{x})$, corresponding to the state $\mathbf{x}$ of the relevant dynamic system.

For nonlinear systems, the exact description of the transition pdf is provided by a linear partial differential equation known as the Fokker–Planck–Kolmogorov equation (FPKE) [2]. Analytical solutions exist only for stationary pdfs and are restricted to a limited class of dynamic systems [1,2]. Thus, researchers are actively looking at numerical approximations to solve the Fokker–Planck equation [7–10], generally using the variational formulation of the problem. The finite difference method and the finite element method have been used successfully for two- and three-dimensional systems. However, these methods are severely handicapped for higher dimensions because the generation of meshes for spaces beyond three dimensions is still impractical. Furthermore, because these techniques rely on the FPKE, they can only be applied to continuous-time dynamic systems. For discrete-time dynamic systems, solving for the exact forecast pdf, which is given by the Chapman–Kolmogorov equation [11], yields the same problems as in the continuous-time case.

Several other approximate techniques exist in the literature to approximate the pdf evolution, the most popular being Monte Carlo methods [12], Gaussian closure [13] (or higher-order moment closure), equivalent linearization [14], and stochastic averaging [15,16]. Monte Carlo methods require extensive computational resources and effort and become increasingly infeasible for high-dimensional dynamic systems. All of these algorithms except Monte Carlo methods are similar in several respects and are suitable only for linear or moderately nonlinear systems, because the effect of higher-order terms can lead to significant errors. Furthermore, all these approaches provide only an approximate description of the uncertainty-propagation problem by restricting the solution to a small number of parameters: for instance, the first N moments of the sought pdf.

The present paper is concerned with improving the approximation to the forecast density function using a finite Gaussian mixture. With a sufficient number of Gaussian components, any pdf may be approximated as closely as desired. Many algorithms [17,18] have been discussed in literature for time evolution of the initial state pdf through a nonlinear dynamic system in discrete time or continuous time. In conventional methods, the weights of different components of a Gaussian mixture are kept constant while propagating the uncertainty through a nonlinear system and are updated only in the presence of measurement data [17,18]. This assumption is valid if the underlying dynamics is linear or the system is marginally nonlinear. The same is not true for the general nonlinear case and new estimates of weights are required for accurate propagation of the state pdf. However, the existing literature provides no means for adaptation of the weights of different Gaussian components in the mixture model during the propagation of state pdf. The lack of adaptive algorithms for the weights of the Gaussian mixture is felt to be a serious disadvantage of existing algorithms and provides the motivation for this paper.

In this paper, two different schemes are introduced to update the weights corresponding to different components of the Gaussian-mixture model during pure propagation. The first method updates the forecast weights by minimizing the integral square difference

Received 17 December 2007; revision received 28 April 2008; accepted for publication 29 April 2008. Copyright © 2008 by Puneet Singla. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0731-5090/08 $10.00 in correspondence with the CCC.

*Ph.D. Student, Department of Computer Science and Engineering; terejanu@buffalo.edu.

†Assistant Professor, Department of Mechanical and Aerospace Engineering; psingla@buffalo.edu.

‡Professor, Department of Mechanical and Aerospace Engineering; tsingh@buffalo.edu.

§Professor, Department of Computer Science and Engineering; peter@buffalo.edu.

between the true forecast pdf and its Gaussian-sum approximation. The second method updates the weights by constraining the Gaussian-sum approximation to satisfy the Fokker–Planck equation for continuous-time dynamic systems. Both methods lead to a convex quadratic programming problem that is computationally efficient.

The structure of the paper is as follows: First, an introduction to conventional Gaussian-sum approximation is presented for the discrete-time dynamic systems, followed by a new scheme for updating the weights of different components of a Gaussian mixture. Next, another novel scheme is presented to update the weights for different components of Gaussian-mixture models for continuous-time dynamic systems. Finally, several numerical examples are considered to illustrate the efficacy of proposed methods.

Update Scheme for Discrete-Time Dynamic Systems

Problem Statement

Consider the following nonlinear discrete-time dynamic system driven by white noise $\boldsymbol{\eta}_k$:

$$\mathbf{x}_{k+1} = \mathbf{f}(k, \mathbf{x}_k) + \boldsymbol{\eta}_k \tag{1}$$

$$\boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k) \tag{2}$$

with the probability density function of the initial conditions given by the following Gaussian sum:

$$p(t_k, \mathbf{x}_k) = \sum_{i=1}^{N} w_k^i \mathcal{N}(\mathbf{x}_k | \boldsymbol{\mu}_k^i, \mathbf{P}_k^i) \tag{3}$$

where

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{P}) = |2\pi\mathbf{P}|^{-1/2} \exp[-\tfrac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \mathbf{P}^{-1}(\mathbf{x}-\boldsymbol{\mu})] \tag{4}$$

We are interested in approximating the probability density function $p(t_{k+1}, \mathbf{x}_{k+1})$ as a Gaussian mixture. The true forecast pdf is given by the Chapman–Kolmogorov equation [11]:

$$p(t_{k+1}, \mathbf{x}_{k+1}) = \int p(t_{k+1}, \mathbf{x}_{k+1} | t_k, \mathbf{x}_k) p(t_k, \mathbf{x}_k) \, \mathrm{d}\mathbf{x}_k \tag{5}$$

where $p(t_{k+1}, \mathbf{x}_{k+1} | t_k, \mathbf{x}_k)$ is the conditional state transition pdf that corresponds to the pdf for the process-noise variable $\boldsymbol{\eta}_k$, generally modeled as Gaussian white noise of covariance $\mathbf{Q}_k$: that is,

$$p(t_{k+1}, \mathbf{x}_{k+1} | t_k, \mathbf{x}_k) = \mathcal{N}(\mathbf{x}_{k+1} | \mathbf{f}(k, \mathbf{x}_k), \mathbf{Q}_k)$$

.

A linear mapping will transform a Gaussian mixture into another Gaussian mixture without changing the weights of different Gaussian components, where the parameters (mean and covariance) of the resulting mixands can be easily computed. But the outcome of a Gaussian that undergoes a nonlinear transformation is generally non-Gaussian. Conventionally, a Gaussian-sum approximation to the forecast density function $p(t_{k+1}, \mathbf{x}_{k+1})$ is obtained by linearizing the nonlinear transformation and assuming weights of different components to be constant:

$$\hat{p}(t_{k+1}, \mathbf{x}_{k+1}) = \sum_{i=1}^{N} w_{k+1}^i \mathcal{N}(\mathbf{x}_{k+1} | \boldsymbol{\mu}_{k+1}^i, \mathbf{P}_{k+1}^i) \tag{6}$$

where the reference values of the parameters of the Gaussian components are given by the prediction step of the extended Kalman filter:

$$w_{k+1}^i = w_k^i \tag{7a}$$

$$\boldsymbol{\mu}_{k+1}^i = \mathbf{f}(k, \boldsymbol{\mu}_k^i) \tag{7b}$$

$$\mathbf{P}_{k+1}^i = \mathbf{A}_k(\boldsymbol{\mu}_k^i) \mathbf{P}_k^i \mathbf{A}_k^T(\boldsymbol{\mu}_k^i) + \mathbf{Q}_k \tag{7c}$$

where

$$\mathbf{A}_k(\mathbf{x}_k) = \frac{\partial \mathbf{f}(k, \mathbf{x}_k)}{\partial \mathbf{x}_k}$$

Evolution schemes other than linearization using Taylor series, such as statistical linearization [19], may be used for obtaining the moments of the Gaussian components. Because they imply linearizations, such approximations are computationally convenient and may be easily used in linear applications.

The reason that the weights are not changed in Eq. (7a) is because it is assumed that the covariances are small enough and that there is a sufficient number of Gaussian components [11,17] such that the linearizations become representative for the dynamics around the means. To better understand this assumption, let us substitute Eq. (3) into Eq. (5), and further, one can derive the following relationships presented in [11]:

$$p(t_{k+1}, \mathbf{x}_{k+1}) = \sum_{i=1}^{N} w_k^i \int \mathcal{N}(\mathbf{x}_k | \boldsymbol{\mu}_k^i, \mathbf{P}_k^i) \mathcal{N}(\mathbf{x}_{k+1} | \mathbf{f}(k, \mathbf{x}_k), \mathbf{Q}_k) \, \mathrm{d}\mathbf{x}_k \tag{8a}$$

$$= \sum_{i=1}^{N} w_k^i \int \mathcal{N}(\mathbf{x}_k | \boldsymbol{\mu}_k^i, \mathbf{P}_k^i) \mathcal{N}(\mathbf{x}_{k+1} | \mathbf{A}_k(\boldsymbol{\mu}_k^i)(\mathbf{x}_k - \boldsymbol{\mu}_k^i) + \mathbf{f}(k, \boldsymbol{\mu}_k^i), \mathbf{Q}_k) \, \mathrm{d}\mathbf{x}_k + \sum_{i=1}^{N} w_k^i \int \boldsymbol{\epsilon}_k \, \mathrm{d}\mathbf{x}_k \tag{8b}$$

where

$$\begin{aligned} \boldsymbol{\epsilon}_k = {} & \mathcal{N}(\mathbf{x}_k | \boldsymbol{\mu}_k^i, \mathbf{P}_k^i)[\mathcal{N}(\mathbf{x}_{k+1} | \mathbf{f}(k, \mathbf{x}_k), \mathbf{Q}_k) \\ & - \mathcal{N}(\mathbf{x}_{k+1} | \mathbf{A}_k(\boldsymbol{\mu}_k^i)(\mathbf{x}_k - \boldsymbol{\mu}_k^i) + \mathbf{f}(k, \boldsymbol{\mu}_k^i), \mathbf{Q}_k)] \end{aligned} \tag{9}$$

Assuming that all covariances $\mathbf{P}_k^i$ of the Gaussian components are small enough such that the linearization around a mean is representative for the dynamics in the vicinity of the respective mean and that there are sufficient number of Gaussian components, then $\mathbf{P}_k^i \to 0$ implies $\int \boldsymbol{\epsilon}_k \, \mathrm{d}\mathbf{x}_k \to 0$. Using this assumption and the fact that the product of two Gaussian densities yields another Gaussian density function, Eq. (8b) becomes, after some algebraic manipulations,

$$p(t_{k+1}, \mathbf{x}_{k+1}) \approx \sum_{i=1}^{N} w_k^i \int \mathcal{N}(\mathbf{x}_k | \boldsymbol{\mu}_k^i, \mathbf{P}_k^i) \mathcal{N}(\mathbf{x}_{k+1} | \mathbf{A}_k(\boldsymbol{\mu}_k^i)(\mathbf{x}_k - \boldsymbol{\mu}_k^i) + \mathbf{f}(k, \boldsymbol{\mu}_k^i), \mathbf{Q}_k) \, \mathrm{d}\mathbf{x}_k \tag{10a}$$

$$= \sum_{i=1}^{N} w_k^i \mathcal{N}(\mathbf{x}_{k+1} | \mathbf{f}(k, \boldsymbol{\mu}_k^i), \mathbf{A}_k(\boldsymbol{\mu}_k^i) \mathbf{P}_k^i \mathbf{A}_k^T(\boldsymbol{\mu}_k^i) + \mathbf{Q}_k) \tag{10b}$$

$$= \sum_{i=1}^{N} w_k^i \mathcal{N}(\mathbf{x}_{k+1} | \boldsymbol{\mu}_{k+1}^i, \mathbf{P}_{k+1}^i) \tag{10c}$$

$$= \hat{p}(t_{k+1}, \mathbf{x}_{k+1}) \tag{10d}$$

In practice, this assumption

$$\forall\ i \quad \mathbf{P}_k^i \to 0 \quad \text{implies} \int \boldsymbol{\epsilon}_k \mathrm{d}\mathbf{x}_k \to 0 \quad \text{implies } w_{k+1}^i = w_k^i \tag{11}$$

may be easily violated, resulting in a poor approximation of the forecast pdf. Practically, the dynamic system may exhibit strong nonlinearities, and the total number of Gaussian components needed to represent the pdf may be restricted due to computational requirements. The existing literature provides no means for adaption of the weights of different Gaussian components in the mixture model during the propagation of the state pdf. The lack of adaptive means for updating the weights of the Gaussian mixture is felt to be a serious disadvantage of existing algorithms and provides the main motivation for this paper.

Weight Update I

In this section, we present a novel scheme to better approximate the forecast pdf by developing the update laws for the forecast weights. The new weights can be obtained by minimizing the following integral square difference between the true pdf $p(t_{k+1}, \mathbf{x}_{k+1})$ and its approximation $\hat{p}(t_{k+1}, \mathbf{x}_{k+1})$ in the least-squares sense:

$$\min_{w^i_{k+1}} \frac{1}{2} \int |p(t_{k+1}, \mathbf{x}_{k+1}) - \hat{p}(t_{k+1}, \mathbf{x}_{k+1})|^2 \,\mathrm{d}\mathbf{x}_{k+1} \tag{12a}$$

$$\text{subject to } \sum_{i=1}^{N} w^i_{k+1} = 1 \tag{12b}$$

$$w^i_{k+1} \geq 0, \qquad i = 1, \ldots, N \tag{12c}$$

Notice that Eqs. (12b) and (12c) are introduced to account for the normality and positivity constraints for the state pdf. Here, the true density function $p(t_{k+1}, \mathbf{x}_{k+1})$ is given by Eq. (5). By substituting Eq. (6) in Eq. (12a) and expanding and grouping the terms in the cost function with regard to the weights, the new cost function is given by

$$J = \tfrac{1}{2}\mathbf{w}^T_{k+1}\mathbf{M}\mathbf{w}_{k+1} - \mathbf{w}^T_{k+1}\mathbf{y} \tag{13}$$

where $\mathbf{w}_{k+1} = [w^1_{k+1} w^2_{k+1} \cdots w^N_{k+1}]^T$, and $\mathbf{M} \in \mathbb{R}^{N\times N}$ is a symmetric matrix given by

$$\mathbf{M} = \int \mathfrak{M}(\mathbf{x}_{k+1})\mathfrak{M}^T(\mathbf{x}_{k+1}) \,\mathrm{d}\mathbf{x}_{k+1} \tag{14}$$

where $\mathfrak{M}$ is a $N \times 1$ vector that contains all the Gaussian components at time $k+1$:

$$\mathfrak{M}(\mathbf{x}_{k+1}) = [\mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}^1_{k+1}, \mathbf{P}^1_{k+1}) \times \mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}^2_{k+1}, \mathbf{P}^2_{k+1}) \cdots \mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}^N_{k+1}, \mathbf{P}^N_{k+1})]^T \tag{15}$$

Thus, the components of matrix $\mathbf{M}$ are easily given by the product rule of two Gaussian density functions, which yields another Gaussian density function [20]. By integrating the product, we are left only with the normalization constant as illustrated next:

$$m_{ij} = \int \mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1})\mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}^j_{k+1}, \mathbf{P}^j_{k+1}) \,\mathrm{d}\mathbf{x}_{k+1}, \qquad i \neq j \tag{16a}$$

$$= \mathcal{N}(\boldsymbol{\mu}^i_{k+1}|\boldsymbol{\mu}^j_{k+1}, \mathbf{P}^i_{k+1} + \mathbf{P}^j_{k+1}) \int \mathcal{N}(\mathbf{x}_{k+1}|\mathbf{P}^{ij}_{k+1}[(\mathbf{P}^i_{k+1})^{-1}\boldsymbol{\mu}^i_{k+1} + (\mathbf{P}^j_{k+1})^{-1}\boldsymbol{\mu}^j_{k+1}], \mathbf{P}^{ij}_{k+1}) \,\mathrm{d}\mathbf{x}_{k+1} = \mathcal{N}(\boldsymbol{\mu}^i_{k+1}|\boldsymbol{\mu}^j_{k+1}, \mathbf{P}^i_{k+1} + \mathbf{P}^j_{k+1}) \tag{16b}$$

$$= |2\pi(\mathbf{P}^i_{k+1} + \mathbf{P}^j_{k+1})|^{-1/2} \exp[-\tfrac{1}{2}(\boldsymbol{\mu}^i_{k+1} - \boldsymbol{\mu}^j_{k+1})^T \times (\mathbf{P}^i_{k+1} + \mathbf{P}^j_{k+1})^{-1}(\boldsymbol{\mu}^i_{k+1} - \boldsymbol{\mu}^j_{k+1})] \tag{16c}$$

$$m_{ii} = \mathcal{N}(\boldsymbol{\mu}^i_{k+1}|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1} + \mathbf{P}^i_{k+1}) = |4\pi\mathbf{P}^i_{k+1}|^{-1/2} \tag{16d}$$

where

$$\mathbf{P}^{ij}_{k+1} = [(\mathbf{P}^i_{k+1})^{-1} + (\mathbf{P}^j_{k+1})^{-1}]^{-1}$$

The components of the vector $\mathbf{y} \in \mathbb{R}^{N\times 1}$ are given by

$$y_i = \int p(t_{k+1}, \mathbf{x}_{k+1})\mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1}) \,\mathrm{d}\mathbf{x}_{k+1}, \qquad i = 1, \ldots, N \tag{17}$$

Now, making use of the Chapman–Kolmogorov equation given by Eq. (5) and the assumption that the process noise is additive and modeled by a Gaussian white noise process, Eq. (17) reduces to

$$y_i = \iint p(t_{k+1}, \mathbf{x}_{k+1}|t_k, \mathbf{x}_k)p(t_k, \mathbf{x}_k)\mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1}) \,\mathrm{d}\mathbf{x}_{k+1} \,\mathrm{d}\mathbf{x}_k \tag{18a}$$

$$= \int\left[\int \mathcal{N}(\mathbf{x}_{k+1}|\mathbf{f}(k, \mathbf{x}_k), \mathbf{Q}_k)\mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1}) \,\mathrm{d}\mathbf{x}_{k+1}\right] \times p(t_k, \mathbf{x}_k) \,\mathrm{d}\mathbf{x}_k \tag{18b}$$

Further, by applying the integration rule of Eq. (16b) of a product of Gaussian densities for the inner integral in Eq. (18b), the new y_i is given by

$$y_i = \int \underbrace{\left[\int \mathcal{N}(\mathbf{x}_{k+1}|\mathbf{f}(k, \mathbf{x}_k), \mathbf{Q}_k)\mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1}) \,\mathrm{d}\mathbf{x}_{k+1}\right]}_{\mathcal{N}(\mathbf{f}(k,\mathbf{x}_k)|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1}+\mathbf{Q}_k)} \times p(t_k, \mathbf{x}_k) \,\mathrm{d}\mathbf{x}_k \tag{19a}$$

$$= \int p(t_k, \mathbf{x}_k)\mathcal{N}(\mathbf{f}(k, \mathbf{x}_k)|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1} + \mathbf{Q}_k) \,\mathrm{d}\mathbf{x}_k \tag{19b}$$

$$\approx \int \hat{p}(t_k, \mathbf{x}_k)\mathcal{N}(\mathbf{f}(k, \mathbf{x}_k)|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1} + \mathbf{Q}_k) \,\mathrm{d}\mathbf{x}_k \tag{19c}$$

Finally, making use of the Gaussian-sum representation of $\hat{p}(t_k, \mathbf{x}_k)$, we get

$$y_i = \sum_{j=1}^{N} w^j_k \int \mathcal{N}(\mathbf{x}_k|\boldsymbol{\mu}^j_k, \mathbf{P}^j_k)\mathcal{N}(\mathbf{f}(k, \mathbf{x}_k)|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1} + \mathbf{Q}_k) \,\mathrm{d}\mathbf{x}_k = \sum_{j=1}^{N} w^j_k N_{ij} \tag{20}$$

where

$$N_{ij} = \int \mathcal{N}(\mathbf{f}(k, \mathbf{x}_k)|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1} + \mathbf{Q}_k)\mathcal{N}(\mathbf{x}_k|\boldsymbol{\mu}^j_k, \mathbf{P}^j_k) \,\mathrm{d}\mathbf{x}_k \tag{21}$$

The result is a sum of expectations of composite functions. Notice that we assume that we have a good Gaussian-sum approximation at time t_k, such that $\hat{p}(t_k, \mathbf{x}_k) \approx p(t_k, \mathbf{x}_k)$.

Further, substitution of Eq. (20) into Eq. (13) leads to the following expression for the cost function:

$$J = \tfrac{1}{2}\mathbf{w}^T_{k+1}\mathbf{M}\mathbf{w}_{k+1} - \mathbf{w}^T_{k+1}\mathbf{N}\mathbf{w}_k \tag{22}$$

where $\mathbf{w}_k = [w^1_k w^2_k \cdots w^N_k]^T$ is the prior weight vector, and the matrix $\mathbf{N} \in \mathbb{R}^{N\times N}$ has the following components:

$$N_{ij} = \int \mathcal{N}(\mathbf{f}(k, \mathbf{x}_k)|\boldsymbol{\mu}^i_{k+1}, \mathbf{P}^i_{k+1} + \mathbf{Q}_k)\mathcal{N}(\mathbf{x}_k|\boldsymbol{\mu}^j_k, \mathbf{P}^j_k) \,\mathrm{d}\mathbf{x}_k \tag{23a}$$

$$= E_{\mathcal{N}(\mathbf{x}_k|\boldsymbol{\mu}_k^j,\mathbf{P}_k^j)}[\mathcal{N}(\mathbf{f}(k,\mathbf{x}_k)|\boldsymbol{\mu}_{k+1}^i,\mathbf{P}_{k+1}^i+\mathbf{Q}_k)] \tag{23b}$$

The expectations (23b) may be computed using Gaussian quadrature, Monte Carlo integration, or unscented transformation [21]. Although the unscented transformation is mostly equivalent with Gaussian quadrature in lower dimensions, the unscented transformation is computationally more appealing in evaluating integrals in higher dimensions, because the number of points grows only linearly with the number of dimensions. However, this comes with a loss in accuracy [22], hence the need for a larger number of points [23] to capture additional information.

In the case of linear transformation, $\mathbf{f}(k,\cdot)=\mathbf{F}_k$, using the integration rule of two Gaussian densities in Eq. (21), and after some algebraic manipulations, the elements $M_{ij}=N_{ij}$; therefore, $\mathbf{M}=\mathbf{N}$. Hence, as expected, the weight vector will be kept constant. However, the numerical approximations made in computing the expectation integrals involved in Eq. (23b) may result in some change in the value of the weight vector.

The final formulation of optimization (12a) can be posed in the quadratic programming framework and solved using readily available solvers:

$$\min_{\mathbf{w}_{k+1}} \; J=\tfrac{1}{2}\mathbf{w}_{k+1}^T\mathbf{M}\mathbf{w}_{k+1}-\mathbf{w}_{k+1}^T\mathbf{N}\mathbf{w}_k$$
$$\text{subject to } \mathbf{1}^T\mathbf{w}_{k+1}=1 \qquad \mathbf{w}_{k+1}\geq\mathbf{0} \tag{24}$$

where $\mathbf{1}\in\mathbb{R}^{N\times 1}$ is a vector of ones and $\mathbf{0}\in\mathbb{R}^{N\times 1}$ is a vector of zeros. If we show that the matrix $\mathbf{M}$ is a positive semidefinite and the cost function J is lower bounded, then the aforementioned optimization problem can be posed as a convex optimization problem and we are guaranteed to have a unique solution [24]. Notice that the integrand $\mathfrak{M}(\mathbf{x}_{k+1}^i)\mathfrak{M}^T(\mathbf{x}_{k+1}^i)$ is a rank-1 symmetric positive semidefinite matrix with nonzero eigenvalue equal to $\mathfrak{M}^T(\mathbf{x}_{k+1}^i)\mathfrak{M}(\mathbf{x}_{k+1}^i)$. Further, we write the integral (14) as an infinite sum as follows:

$$\mathbf{M}=\sum_i\mathfrak{M}(\mathbf{x}_{k+1}^i)\mathfrak{M}^T(\mathbf{x}_{k+1}^i) \tag{25}$$

Making use of the fact that the summation of two positive semidefinite matrix is also a positive semidefinite matrix, we conclude that matrix $\mathbf{M}$ is a positive semidefinite matrix because $\mathbf{M}$ is a sum of positive semidefinite matrices. Further, notice that the linear term $\mathbf{w}_{k+1}^T\mathbf{N}\mathbf{w}_k$ in cost function J is lower bounded because $\mathbf{N}\geq 0$, and both $\mathbf{w}_{k+1}$ and $\mathbf{w}_k$ have to lie between 0 and 1. Hence, we can conclude that the cost function J is always bounded below and we are guaranteed to have a unique solution.

Update Scheme for Continuous-Time Dynamic Systems

Problem Statement

Consider a general n-dimensional noise-driven nonlinear dynamic system with uncertain initial conditions, given by the equation

$$\dot{\mathbf{x}}=\mathbf{f}(t,\mathbf{x})+\mathbf{g}(t,\mathbf{x})\Gamma(t) \tag{26}$$

where $\Gamma(t)$ represents a Gaussian white noise process with the correlation function $\mathbf{Q}\delta(t_1-t_2)$, and the initial state uncertainty is captured by the pdf $p(t_0,\mathbf{x})$. Then the time evolution of $p(t_0,\mathbf{x})$ is described by the following FPKE, which is a second-order partial differential equation in $p(t,\mathbf{x})$:

$$\frac{\partial}{\partial t}p(t,\mathbf{x})=\mathcal{L}_{\mathcal{FP}}[p(t,\mathbf{x})] \tag{27}$$

where

$$\mathcal{L}_{\mathcal{FP}}=\left[-\sum_{i=1}^n\frac{\partial}{\partial x_i}D_i^{(1)}(t,\mathbf{x})[\cdot]+\sum_{i=1}^n\sum_{j=1}^n\frac{\partial^2}{\partial x_i\partial x_j}D_{ij}^{(2)}(t,\mathbf{x})[\cdot]\right] \tag{28a}$$

$$D^{(1)}(t,\mathbf{x})=\mathbf{f}(t,\mathbf{x})+\frac{1}{2}\frac{\partial\mathbf{g}(t,\mathbf{x})}{\partial\mathbf{x}}\mathbf{Q}\mathbf{g}(t,\mathbf{x}) \tag{28b}$$

$$D^{(2)}(t,\mathbf{x})=\tfrac{1}{2}\mathbf{g}(t,\mathbf{x})\mathbf{Q}\mathbf{g}^T(t,\mathbf{x}) \tag{28c}$$

We mention that Eqs. (27) and (28a) represent the Stratonovich interpretation for the FPKE, which is more popular among engineers and physicists. An alternative interpretation (Itō interpretation) for the FPKE is also commonly used among mathematicians and leads to the same equation if $\mathbf{g}(\cdot)$ is constant in Eq. (26). In this paper, we use the Stratonovich interpretation for the FPKE to illustrate the main idea of updating the weights corresponding to different component of the Gaussian-mixture model. However, one can use the Itō interpretation for the FPKE equation without compromising the formulation presented in the next section. Irrespective of the form (Stratonovich or Itō), the FPKE is a formidable problem to solve, because of the following issues: 1) positivity of the pdf

$$p(t,\mathbf{x})\geq 0 \quad \forall\; t,x$$

2) normalization constraint of the pdf

$$\int_{\mathbb{R}^n}p(t,\mathbf{x})\,\mathrm{d}\mathbf{x}=1$$

and 3) no fixed solution domain (how to impose boundary conditions in a finite region and restrict numerical computation to regions in which $p>\sim 10^{-9}$).

Let us assume that underlying pdf can be approximated by a finite sum of Gaussian pdfs:

$$\hat{p}(t,\mathbf{x})=\sum_{i=1}^N w_i p_{g_i} \tag{29}$$

where

$$p_{g_i}=\mathcal{N}(\mathbf{x}(t)|\boldsymbol{\mu}_i,\mathbf{P}_i)$$

where $\boldsymbol{\mu}_i$ and $\mathbf{P}_i$ represent the mean and covariance of the ith component of the Gaussian pdf, respectively, and w_i denotes the amplitude of ith Gaussian in the mixture. The positivity and normalization constraint on the mixture pdf $\hat{p}(t,\mathbf{x})$ leads to following constraints on the amplitude vector:

$$\sum_{i=1}^N w_i=1, \qquad w_i\geq 0 \tag{30}$$

All the components of the mixture pdf (29) are Gaussian and thus only estimates of their mean and covariance need to be maintained to obtain the optimal state estimates that can be propagated using the extended-Kalman-filter time-update equations:

$$\dot{\boldsymbol{\mu}}_i=\mathbf{f}(\boldsymbol{\mu}_i) \tag{31}$$

$$\dot{\mathbf{P}}_i=\mathbf{A}_i\mathbf{P}_i+\mathbf{P}_\mathbf{i}\mathbf{A}_i^T+\mathbf{g}(t,\boldsymbol{\mu}_i)\mathbf{Q}\mathbf{g}^T(t,\boldsymbol{\mu}_i) \tag{32}$$

where

$$\mathbf{A}_i=\left.\frac{\partial\mathbf{f}(t,\mathbf{x})}{\partial\mathbf{x}}\right|_{x=\boldsymbol{\mu}_i} \tag{33}$$

Notice that the weights w_i corresponding to each Gaussian component are unknown. In a conventional Gaussian-sum filter, the weights are initialized such that the initial mixture pdf approximates the given initial pdf and it is assumed that the weight does not change over time. This assumption is valid if the underlying dynamics are linear or the system is marginally nonlinear. The same is not true for the general nonlinear case, and new estimates of weights are required for accurate propagation of the state pdf.

Weight Update II

In this section, a novel method is described to update the weights of different components of the Gaussian mixture of Eq. (29). The main idea is that the mixture pdf of Eq. (29), $\hat{p}(t,\mathbf{x})$, should satisfy the Fokker–Planck equation (28a), and the Fokker–Planck-equation error can be used as a feedback to update the weights of different Gaussian components in the mixture pdf [25]. In other words, we seek to minimize the Fokker–Planck-equation error under the assumption of Eqs. (29), (31), and (32).

Substituting Eq. (29) in Eq. (28a) leads to

$$e(t,\mathbf{x}) = \frac{\partial \hat{p}(t,\mathbf{x})}{\partial t} - \mathcal{L}_{\mathcal{FP}}[\hat{p}(t,\mathbf{x})] \tag{34}$$

$$= \frac{\partial \hat{p}(t,\mathbf{x})}{\partial t} - \left[-\sum_{i=1}^{n} \frac{\partial D_i^{(1)}(t,\mathbf{x})}{\partial x_i} + \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2 D_{ij}^{(2)}(t,\mathbf{x})}{\partial x_i \partial x_j}\right][\hat{p}(t,\mathbf{x})] \tag{35}$$

where $\mathcal{L}_{\mathcal{FP}}(\cdot)$ is the so-called Fokker–Planck operator, and

$$\frac{\partial \hat{p}(t,\mathbf{x})}{\partial t} = \sum_{i=1}^{N} w_i \left[\frac{\partial p_{g_i}^T}{\partial \boldsymbol{\mu}_i} \dot{\boldsymbol{\mu}}_i + \sum_{j=1}^{n}\sum_{k=1}^{n} \frac{\partial p_{g_i}}{\partial P_{i_{jk}}} \dot{P}_{i_{jk}}\right] \tag{36}$$

where $P_{i_{jk}}$ is the jkth element of the ith covariance matrix $\mathbf{P}_i$. Further, substitution of Eqs. (28b) and (28c) along with Eq. (36) into Eq. (35) leads to

$$e(t,\mathbf{x}) = \sum_{i=1}^{N} w_i \mathfrak{L}_i(t,\mathbf{x}) = \mathfrak{L}^T \mathbf{w} \tag{37}$$

where $\mathbf{w}$ is a $N \times 1$ vector of Gaussian weights, and $\mathfrak{L}_i$ is given by

$$\begin{aligned}\mathfrak{L}_i(t,\mathbf{x}) = \Bigg[&\frac{\partial p_{g_i}^T}{\partial \boldsymbol{\mu}_i} \mathbf{f}(t,\boldsymbol{\mu}_i) + \sum_{j=1}^{n}\sum_{k=1}^{n} \frac{\partial p_{g_i}}{\partial P_{i_{jk}}} \dot{P}_{i_{jk}} \\ &+ \sum_{j=1}^{n}\Bigg(f_j(t,\mathbf{x}) \frac{\partial p_{g_i}}{\partial x_j} + p_{g_i} \frac{\partial f_j(t,\mathbf{x})}{\partial x_j} \\ &+ \frac{1}{2}\frac{\partial d_j^{(1)}(t,\mathbf{x}) p_{g_i}}{\partial x_j} - \frac{1}{2}\sum_{k=1}^{n} \frac{\partial^2 d_{jk}^{(2)}(t,\mathbf{x}) p_{g_i}}{x_j x_k}\Bigg)\Bigg]\end{aligned} \tag{38a}$$

where $d^{(1)}(t,\mathbf{x})$ and $d^{(2)}(t,\mathbf{x})$ are given as

$$d^{(1)}(t,\mathbf{x}) = \frac{1}{2}\frac{\partial \mathbf{g}(t,\mathbf{x})}{\partial \mathbf{x}} \mathbf{Q}\mathbf{g}(t,\mathbf{x}) \tag{39a}$$

$$d^{(2)}(t,\mathbf{x}) = \tfrac{1}{2}\mathbf{g}(t,\mathbf{x})\mathbf{Q}\mathbf{g}^T(t,\mathbf{x}) \tag{39b}$$

Further, different derivatives in the preceding equation can be computed using the following analytical formulas:

$$\frac{\partial p_{g_i}}{\partial \boldsymbol{\mu}_i} = \mathbf{P}_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i) p_{g_i} \tag{40a}$$

$$\begin{aligned}\frac{\partial p_{g_i}}{\partial \mathbf{P}_i} &= -\frac{p_{g_i}}{2|\mathbf{P}_i|}\frac{\partial |\mathbf{P}_i|}{\partial \mathbf{P}_i} - \frac{p_{g_i}}{2}\frac{\partial (\mathbf{x} - \boldsymbol{\mu}_i)^T \mathbf{P}_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)}{\partial \mathbf{P}_i} \\ &= -\frac{p_{g_i}}{2}\mathbf{P}_i^{-1} - \frac{p_{g_i}}{2}\frac{\partial (\mathbf{x} - \boldsymbol{\mu}_i)^T \mathbf{P}_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)}{\partial \mathbf{P}_i} \\ &= -\frac{p_{g_i}}{2}\mathbf{P}_i^{-1} + \frac{p_{g_i}}{2}\mathbf{P}_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^T \mathbf{P}_i^{-1}\end{aligned} \tag{40b}$$

$$\frac{\partial p_{g_i}}{\partial \mathbf{x}} = -\mathbf{P}_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i) p_{g_i} \tag{40c}$$

$$\frac{\partial^2 p_{g_i}}{\partial \mathbf{x}\mathbf{x}^T} = -\mathbf{P}_i^{-1}\left[\mathbf{I} + (\mathbf{x} - \boldsymbol{\mu}_i)\frac{\partial p_{g_i}^T}{\partial \mathbf{x}}\right] p_{g_i} \tag{40d}$$

At a given time instant, after propagating the mean $\boldsymbol{\mu}_i$ and the covariance $\mathbf{P}_i$ of individual Gaussian elements using Eqs. (31) and (32), we seek to update weights by minimizing the FPE equation error over some volume of interest V:

$$\min_{w_i} \frac{1}{2}\int_{V \subset \mathbb{R}^n} e^2(t,\mathbf{x})\,\mathrm{d}\mathbf{x}$$

$$\text{subject to } \sum_{i=1}^{N} w_i = 1 \qquad w_i \geq 0, \qquad i = 1,\ldots,N \tag{41}$$

The Fokker–Planck-equation error of Eq. (37) is linear in Gaussian weights w_i, and hence the aforementioned problem can be written as a quadratic programming problem:

$$\min_{\mathbf{w}} \tfrac{1}{2}\mathbf{w}^T \mathbf{L}\mathbf{w} \qquad \text{subject to } \mathbf{1}^T\mathbf{w} = 1 \qquad \mathbf{w} \geq \mathbf{0} \tag{42}$$

where $\mathbf{1} \in \mathbb{R}^{N\times 1}$ is a vector of ones, $\mathbf{0} \in \mathbb{R}^{N\times 1}$ is a vector of zeros, and $\mathbf{L}$ is given by

$$\begin{aligned}\mathbf{L} &= \int_{V \subset \mathbb{R}^n} \mathfrak{L}(\mathbf{x})\mathfrak{L}^T(\mathbf{x})\,\mathrm{d}V \\ &= \begin{bmatrix} \int_{V\subset\mathbb{R}^n} \mathfrak{L}_1\mathfrak{L}_1\,\mathrm{d}\mathbf{x} & \int_{V\subset\mathbb{R}^n} \mathfrak{L}_2\mathfrak{L}_1\,\mathrm{d}\mathbf{x} & \cdots & \int_{V\subset\mathbb{R}^n} \mathfrak{L}_N\mathfrak{L}_1\,\mathrm{d}\mathbf{x} \\ \int_{V\subset\mathbb{R}^n} \mathfrak{L}_1\mathfrak{L}_2\,\mathrm{d}\mathbf{x} & \int_{V\subset\mathbb{R}^n} \mathfrak{L}_2\mathfrak{L}_2\,\mathrm{d}\mathbf{x} & \cdots & \int_{V\subset\mathbb{R}^n} \mathfrak{L}_N\mathfrak{L}_2\,\mathrm{d}\mathbf{x} \\ \vdots & & \ddots & \vdots \\ \int_{V\subset\mathbb{R}^n} \mathfrak{L}_1\mathfrak{L}_N\,\mathrm{d}\mathbf{x} & \int_{V\subset\mathbb{R}^n} \mathfrak{L}_2\mathfrak{L}_N\,\mathrm{d}\mathbf{x} & \cdots & \int_{V\subset\mathbb{R}^n} \mathfrak{L}_N\mathfrak{L}_N\,\mathrm{d}\mathbf{x} \end{bmatrix}\end{aligned} \tag{43}$$

If we show that the matrix $\mathbf{L}$ is a positive semidefinite, then the aforementioned optimization problem can be posed as a convex optimization problem and we are guaranteed to have a unique solution. Notice that the integrand $\mathfrak{L}(\mathbf{x})\mathfrak{L}^T(\mathbf{x})$ is a rank-1 symmetric positive semidefinite matrix with a nonzero eigenvalue equal to $\mathfrak{L}^T(\mathbf{x})\mathfrak{L}(\mathbf{x})$. Further, the integral in the definition of matrix L can be approximated by an infinite sum of the positive semidefinite matrix:

$$\int_{V\subset\mathbb{R}^n} \mathfrak{L}(\mathbf{x})\mathfrak{L}^T(\mathbf{x})\,\mathrm{d}V = \sum_i \mathfrak{L}(\mathbf{x}_i)\mathfrak{L}^T(\mathbf{x}_i) \tag{44}$$

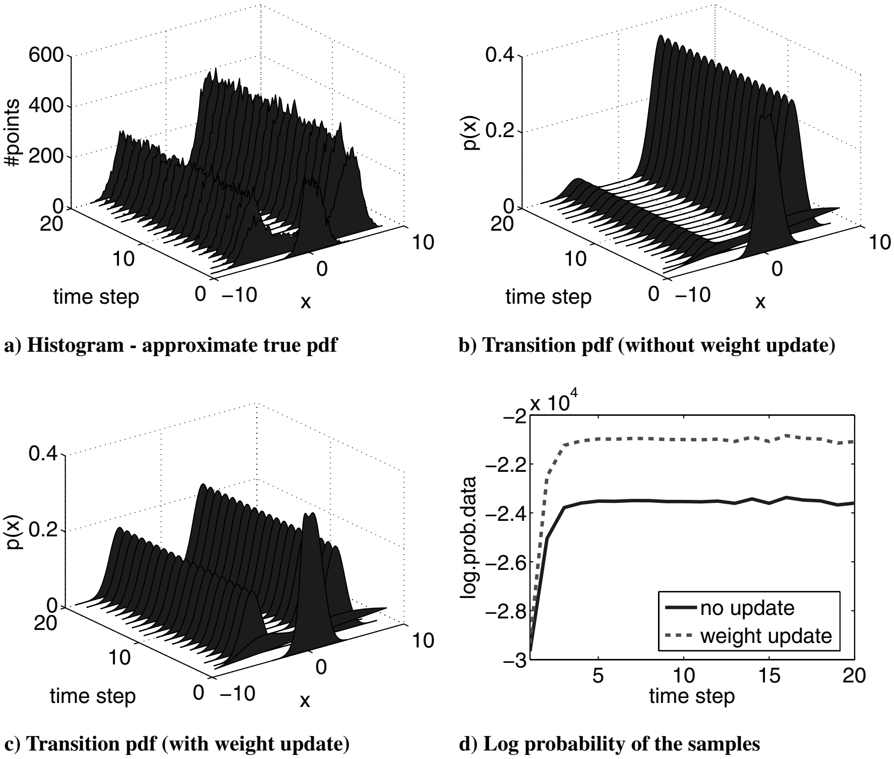
Making use of the fact that summation of two positive semidefinite matrices is also a positive semidefinite matrix, we conclude that matrix $\mathbf{L}$ is a positive semidefinite matrix and thus the optimization problem of Eq. (42) has a unique solution.

Notice that to carry out this minimization, we need to evaluate integrals involving Gaussian pdfs over volume V, which can be computed exactly for polynomial nonlinearity and, in general, can be approximated by the Gaussian quadrature method.

Numerical Results

The two update schemes presented in this paper were tested on a variety of benchmark problems studied extensively in the literature [7–9]. In this section, we present results from these studies. For a discrete-time-update scheme, we represent the system dynamics by converting the continuous-time differential equations to discrete-time difference equations. The selection of the examples presented in assessing the performance of the proposed methods was made such that they cover a wide spectrum of scenarios, including 1-D and 2-D dynamic systems and different initial conditions for the Gaussian components: equally weighted, randomly weighted, and having particular weight assignments. The two update schemes are compared with the usual procedure of not updating the weights on several dynamic systems in which either a closed-form solution for the stationary pdf is available or a Monte Carlo simulation is carried out. For the examples in which an analytical solution is available, the following integral of the absolute error was used to quantitatively compare the methods:

$$E = \int |p(t,\mathbf{x}) - \hat{p}(t,\mathbf{x})|\,\mathrm{d}\mathbf{x} \tag{45}$$



a) Histogram - approximate true pdf

b) Transition pdf (without weight update)

c) Transition pdf (with weight update)

d) Log probability of the samples

Fig. 1 Numerical results example 1.

The different expectation integrals that appear in the cost function of the two methods have compact support and were numerically approximated using the Gaussian quadrature method.

Example 1

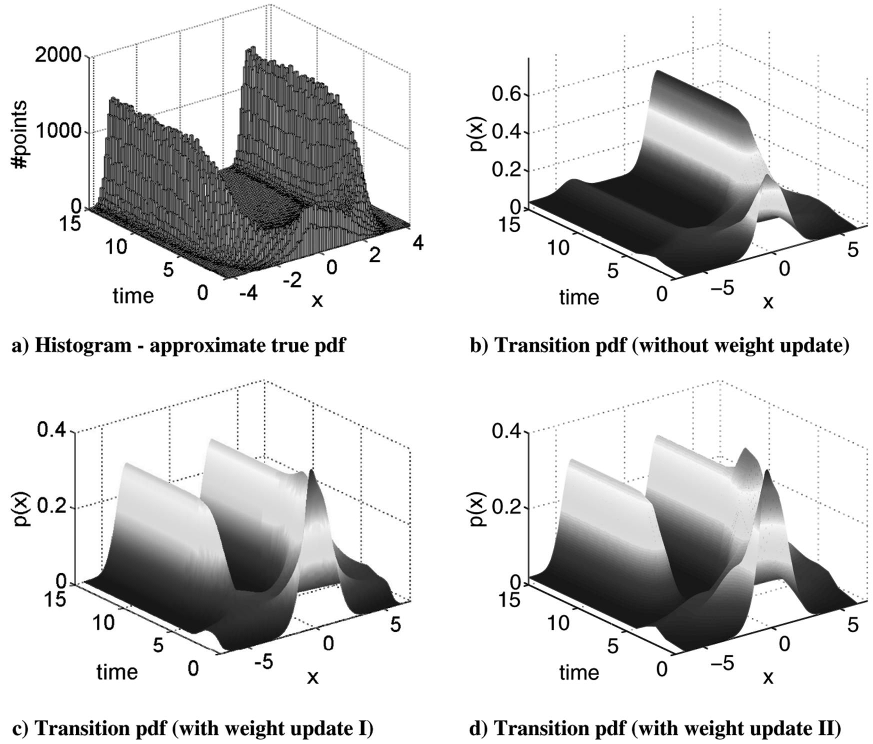
The first update method was applied on the following nonlinear discrete-time dynamic system with uncertain initial condition given by Eq. (47):

$$x_{k+1} = \frac{1}{2}x_k + 10\frac{x_k}{1 + x_k^2} + \eta_k \tag{46}$$

where $\eta_k \sim \mathcal{N}(0, 1)$

$$x_0 \sim 0.1\mathcal{N}(-0.5, 0.1) + 0.9\mathcal{N}(0.5, 1) \tag{47}$$

The moments of the two Gaussian components are propagated for 20 time steps using Eqs. (7b) and (7c). Because there is no analytical



a) Histogram - approximate true pdf

b) Transition pdf (without weight update)

c) Transition pdf (with weight update I)

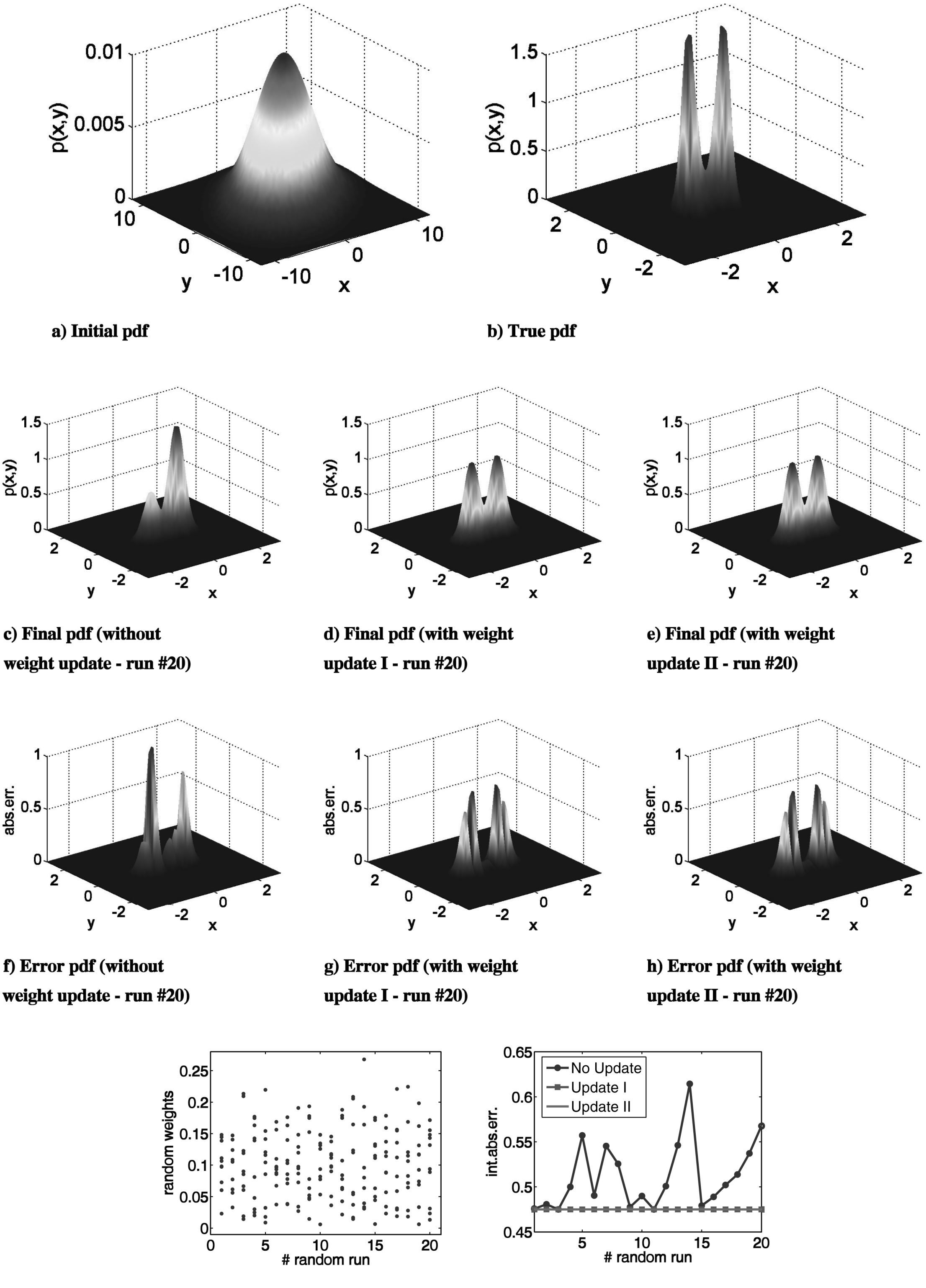
d) Transition pdf (with weight update II)

Fig. 2 Numerical results example 2.

solution for the forecast pdf, we run a Monte Carlo simulation using 10,000 samples and compute the histogram of the samples at each time step, as shown in Fig. 1a. Figure 1b shows the pdf approximation without updating the weights and Fig. 1c plots the pdf approximation with updated weights. By updating the weights, we are able to better capture the two modes presented in Fig. 1a. In addition to this, log probability of the Monte Carlo sample points was computed according to the following relationship:

$$L = \sum_{j=1}^{M} \log \sum_{i=1}^{N} w_i \mathcal{N}(\mathbf{x}^j | \boldsymbol{\mu}_i, \mathbf{P}_i) \tag{48}$$

where M is the total number of samples used in the Monte Carlo approximation, and the higher value of L, the better the pdf approximation. Figure 1d shows the plot of log probability of the samples with and without updating the weights of the Gaussian



a) Initial pdf

b) True pdf

c) Final pdf (without weight update - run #20)

d) Final pdf (with weight update I - run #20)

e) Final pdf (with weight update II - run #20)

f) Error pdf (without weight update - run #20)

g) Error pdf (with weight update I - run #20)

h) Error pdf (with weight update II - run #20)

i) Random weights run

j) Integral absolute error run

Fig. 3 Numerical results example 3.

mixture. As expected, updating the weights of the Gaussian mixture leads to higher log probability of the samples. Hence, we conclude that the adaptation of the weights during propagation leads to more accurate pdf approximation than without weight updates.

Example 2

For the second example, we consider the following continuous-time dynamic system with an uncertain initial condition given by Eq. (50):

$$\dot{x} = \sin(x) + \mathcal{G}(t) \tag{49}$$

where $Q = 1$

$$x_0 \sim 0.1\mathcal{N}(-0.5, 0.1) + 0.9\mathcal{N}(0.2, 1) \tag{50}$$

The moments of the two Gaussian components are propagated for 15 s using Eqs. (31) and (32). Again, because there is no analytical solution for the transition pdf, we run a Monte Carlo simulation using 10,000 samples and compute the histogram of the samples at each time step ($\Delta t = 0.1$ s.), as shown in Fig. 2a. Figure 2b shows the pdf approximation without updating the weights and Figs. 2c and 2d show the plot of the approximated pdf using the two methods introduced in this paper. We mention that weight-update scheme I is implemented by approximating the continuous-time dynamic system by a discrete-time difference equation. It is clear from these plots that we are able to better capture the transition pdf with the incorporation of a weight-update scheme. Also, the shape of the approximated pdf is in accordance with the histograms generated by Monte Carlo simulations and is consistent with the behavior of the dynamic system, which has two attractors at $-\pi$ and π.

Example 3

We consider the following 2-D nonlinear dynamic system for analysis:

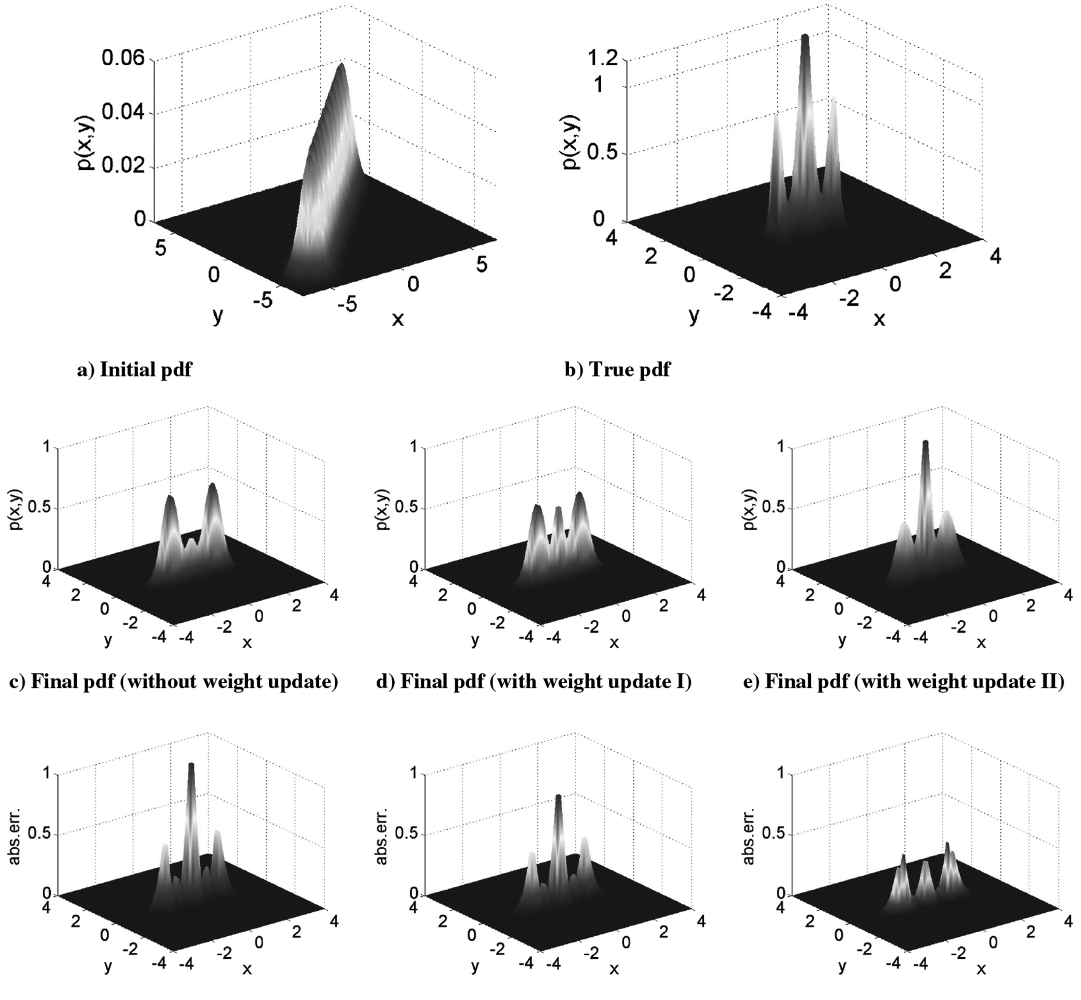
$$\ddot{x} + \eta\dot{x} + \alpha x + \beta x^3 = g(t)\mathcal{G}(t) \tag{51}$$

Equation (51) represents a noise-driven duffing oscillator with a soft spring ($\alpha\beta < 0$, $\eta > 0$) and included damping (to ensure the presence of a stationary solution: a bimodal pdf). For simulation purposes, we use $g(t) = 1$, $Q = 1$, $\eta = 10$, $\alpha = -1$, and $\beta = 3$, and the stationary probability density function given by Eq. (52) is plotted in Fig. 3b.

$$p(x, \dot{x}) \propto \exp\left(-2\frac{\eta}{g^2 Q}\left(\frac{\alpha}{2}x^2 + \frac{\beta}{4}x^4 + \frac{1}{2}\dot{x}^2\right)\right) \tag{52}$$

Notice that the stationary pdf is an exponential function of the steady-state system Hamiltonian, scaled by the parameter $-2(\eta/g^2 Q)$ [1].

To approximate the stationary pdf given by Eq. (52), we assume the initial pdf to be a mixture of 10 Gaussian pdfs with centers uniformly distributed between (−5, −5) and (5, 5). The covariance matrix for each Gaussian component is chosen to be $\mathbf{P} = 10 \times \mathbf{I}$, where $\mathbf{I}$ is the 2×2 identity matrix, and the initial weights of the Gaussian components are randomly assigned a number between 0 and 1. The weights were randomly generated in 20 different runs (see



a) Initial pdf

b) True pdf

c) Final pdf (without weight update)

d) Final pdf (with weight update I)

e) Final pdf (with weight update II)

f) Error pdf (without weight update)

g) Error pdf (with weight update I)

h) Error pdf (with weight update II)

Fig. 4 Numerical results example 4.

Fig. 3i for weights/run), and the initial pdf for the last run is plotted in Fig. 3a.

The center and corresponding covariance matrices of the Gaussian components are linearly propagated for 100 s. Figure 3c shows the plot of the approximated pdf without updating the weights of the Gaussian mixture and Fig. 3f shows the corresponding approximation-error plot for the weights of the last run. As expected, the final pdf is biased toward one of the modes of the true pdf, due to the fact that the initial weights of those components were large.

It is clear that although the approximated pdf did capture some non-Gaussian behavior, it failed to capture both modes accurately. Further, Figs. 3d and 3h show the plot of the approximated stationary pdfs after updating the weights of different components of the Gaussian mixture using weight-update schemes I and II, respectively. It is clear that both methods (update I and update II) yield approximately the same pdf with the corresponding approximation error given in Figs. 3g and 3h. The integral absolute error of Eq. (45) for the updated Gaussian sum is lower than the one without weight update. This result is consistent in all 20 runs, as shown in Fig. 3j (in which, for different initial weights, both methods are able to yield approximately the same approximation error), and lower than without updating the weights.

The first method was applied recursively every second to update the weights, and the second method was applied only once at the end of the simulation. The results of the 20 runs were averaged and tabulated at the end of the section in Table 1. As expected, with the adaptation of the weights, the approximation error was reduced.

Example 4

For the fourth example, we have considered the following the 2-D noise-driven quintic oscillator given by Eq. (53):

Table 1 Numerical approximation of the integral of the absolute error

	No update	Update I	Update II
Example 3 (avg 20 runs)	0.50 ± 0.02	0.47 ± 10^{-15}	0.47 ± 10^{-15}
Example 4 (avg 20 runs)	0.86 ± 0.07	0.74 ± 10^{-14}	0.50 ± 10^{-15}
Example 5 (avg 20 runs)	0.75 ± 0.01	0.19 ± 10^{-15}	0.26 ± 10^{-15}

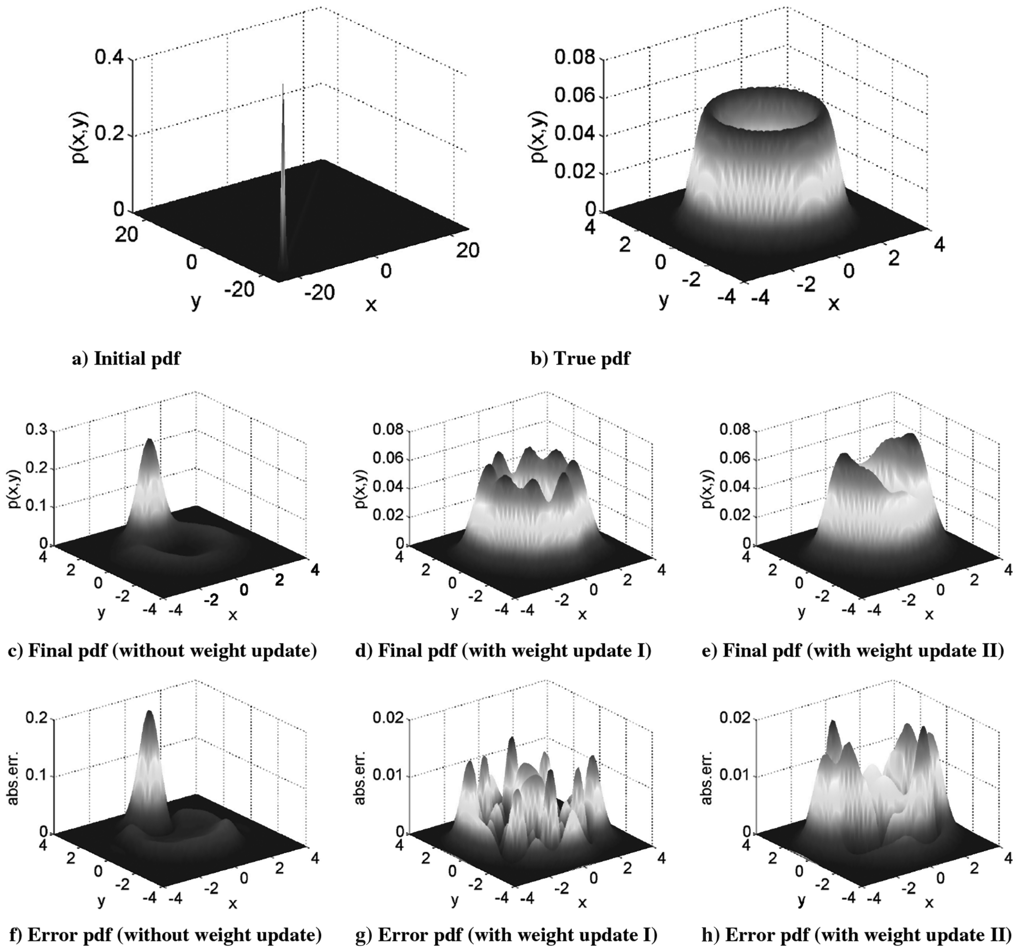
$$\ddot{x} + \eta\dot{x} + x(\epsilon_1 + \epsilon_2 x^2 + \epsilon_3 x^4) = g(t)\mathcal{G}(t) \tag{53}$$

For simulation purposes, we choose the following parameters: $g(t) = 1$, $Q = 1$, $\eta = 10$, $\epsilon_1 = 1$, $\epsilon_2 = -3.065$, and $\epsilon_3 = 1.825$. The true stationary pdf [1] given by Eq. (54), plotted in Fig. 4b, is trimodal in this case, with each of the mode centered at three equilibrium points of the oscillator:

$$p(x,\dot{x}) \propto \exp\left(-2\frac{\eta}{g^2 Q}\left(\frac{\epsilon_1}{2}x^2 + \frac{\epsilon_2}{4}x^4 + \frac{\epsilon_3}{6}x^6 + \frac{1}{2}\dot{x}^2\right)\right) \tag{54}$$

To approximate the stationary pdf, we assume the initial pdf to be a mixture of 11 Gaussian pdfs with centers uniformly distributed between $(-5, -5)$ and $(5, 5)$. The covariance matrix for each Gaussian component is chosen to be $\mathbf{P} = 0.33 \times \mathbf{I}$, and initially, all Gaussian components are assumed to be equally weighted, as shown in Fig. 4a.

The center and corresponding covariance matrices of the Gaussian components are propagated for 1000 s in accordance with Eqs. (31) and (32). Figure 4c shows the plot of the approximated pdf without updating the weights of the Gaussian mixture, and Fig. 4f shows the corresponding approximation-error plot. As expected, the final pdf is not able to capture the mode centered at the unstable equilibrium point (0, 0). It is clear that although the approximated pdf did capture



a) Initial pdf

b) True pdf

c) Final pdf (without weight update)

d) Final pdf (with weight update I)

e) Final pdf (with weight update II)

f) Error pdf (without weight update)

g) Error pdf (with weight update I)

h) Error pdf (with weight update II)

Fig. 5 Numerical results example 5.

some non-Gaussian behavior, it fails to capture all three modes accurately.

Figures 4d and 4e show the plot of the approximated stationary pdf after updating the weights of different components of the Gaussian-mixture model using the first and the second methods, respectively. Figures 4g and 4h show the approximation error of the two methods. The integral absolute errors corresponding to 20 different runs with random assignment of Gaussian weights is tabulated in Table 1. It is clear that the average approximation error and corresponding standard deviation was reduced with the adaptation of the weights. The first method was applied recursively every 0.5 s to update the weights, and the second method was applied only once at the end of the simulation.

Example 5

The last example involves the state pdf propagation through the noise-driven energy-dependent damping oscillator given by the following equation:

$$\ddot{x} + \beta\dot{x} + x + \alpha(x^2 + \dot{x}^2)\dot{x} = g(t)\mathcal{G}(t) \tag{55}$$

For simulation purposes, we choose the following parameters: $g(t) = 1$, $Q = 1/\pi$, $\alpha = 0.125$, and $\beta = -0.5$, and the stationary probability density function [9] is given by

$$p(x, \dot{x}) \propto \exp\left(-\frac{\eta}{2g^2}\left(\beta(x^2 + \dot{x}^2) + \frac{\alpha}{2}(x^2 + \dot{x}^2)^2\right)\right) \tag{56}$$

Figure 5b shows the plot of the true stationary pdf with most of the probability mass centered at the boundary of the stable limit cycle in this case.

To approximate the stationary pdf, we assume the initial pdf to be a mixture of 100 Gaussian pdfs with centers uniformly distributed between $(-20, -20)$ and $(20, 20)$. The covariance matrix for each Gaussian component is chosen to be $\mathbf{P} = 0.1347 \times \mathbf{I}$. Initially, the weight of the first Gaussian component is $w_1 = 0.5$ and the rest of the 99 Gaussian components have equal weights: $w_{2\cdots100} = 0.0051$.

The center and corresponding covariance matrices of the Gaussian components are propagated for 1000 s using Eqs. (31) and (32). Figure 5c shows the plot of the approximated pdf without updating the weights of the Gaussian mixture, and Fig. 5f shows the corresponding approximation-error plot. As expected, the final pdf is not able to capture the non-Gaussian behavior in this case.

Figures 5d and 5e show the plot of the approximated stationary pdf after updating the weights of different components of the Gaussian-mixture model using the first and the second methods, respectively. Figures 5g and 5h show the approximation error of the two methods. The integral absolute errors corresponding to 20 different runs with random assignment of Gaussian weights is tabulated in Table 1. It is clear that the average approximation error and corresponding standard deviation were reduced with the adaptation of the weights. The first method was applied recursively every 0.5 s to update the weights, and the second method was applied only once at the end of the simulation.

For continuous-time dynamic systems, the overall computational complexity in applying the second method is small because it is used to update the weights only when an estimate or approximation of the conditional pdf has to be computed. In the continuous-time case, the first method has to be applied sequentially on the discretized equations of the dynamic system, making the method more computationally expensive than the second one. Therefore, discretization errors may be propagated in the update formulation of the weights.

Finally, we mention that the numerical results presented in this section reiterates our observation and supports the conclusion that the better performance of proposed algorithms can be attributed to the adaptation of amplitude corresponding to different components of the Gaussian mixture. These dramatic advantages from five different problems provide compelling evidence for the merits of the proposed algorithms.

Conclusions

Two update schemes for the forecast weights are presented to obtain a better Gaussian-sum approximation to the forecast pdf. The difference between the two methods comes from the particularities of their derivations. The first method updates the weights such that they minimize the integral square difference between the true probability density function and its approximation. The derivation of the first method indicates that it is more appropriate for discrete-time systems. The second method is derived such that the Gaussian sum satisfies the FPKE, which indicates that it is more appropriate for continuous-time systems. The weights are updated such that they minimize the FPKE error. Both method leads to two different convex quadratic minimization problems that are guaranteed to have a unique solution.

Several benchmark problems are provided to compare the two methods with the usual procedure of not updating the weights in pure propagation settings. In all of these diverse test problems, the proposed two algorithms are found to produce considerably smaller errors than with existing methods. The results presented in this paper serve to illustrate the usefulness of adaptation of weights corresponding to different components of the Gaussian-sum model. We anticipate that major advances are possible in numerous engineering applications such as studying the biochemical-plume propagation using uncertain meteorological data, orbit determination using uncertain measurements, and pricing problems in finance by virtue of the proposed ideas. In filtering settings, in which observations are available, the study of the impact of such an update scheme for Gaussian-sum filtering is set as future work. Finally, we fully appreciate the truth that results from any test are difficult to extrapolate; however, testing the new algorithm on five benchmark problems does provide compelling evidence and a basis for optimism.

Acknowledgment

This work was supported by the Defense Threat Reduction Agency (DTRA) under contract no. W911NF-06-C-0162. The authors gratefully acknowledge the support and constructive suggestions of John Hannan of DTRA.

References

[1] Fuller, A. T., "Analysis of Nonlinear Stochastic Systems by Means of the Fokker–Planck Equation," *International Journal of Control*, Vol. 9, No. 6, June 1969, pp. 603–655. doi:10.1080/00207176908905786

[2] Risken, H., *The Fokker–Planck Equation: Methods of Solution and Applications*, Springer, New York, 1989, pp. 32–62.

[3] Polidori, D. C., and Beck, J. L., "Approximate Solutions for Nonlinear Vibration Problems," *Probabilistic Engineering Mechanics*, Vol. 11, No. 3, July 1996, pp. 179–185. doi:10.1016/0266-8920(96)00011-2

[4] Chodas, P. W., and Yeomans, D. K., "Orbit Determination and Estimation of Impact Probability for Near Earth Objects," *Guidance and Control 1999*, Advances in the Astronautical Sciences, Vol. 101, Univelt, San Diego, CA, 1999, pp. 21–40.

[5] Wang, M. C., and Uhlenbeck, G., "On the Theory of Brownian Motion 2," *Reviews of Modern Physics*, Vol. 17, Nos. 2–3, 1945, pp. 323–342. doi:10.1103/RevModPhys.17.323

[6] Chakravorty, S., Kumar, M., and Singla, P., "A Quasi-Gaussian Kalman Filter," *2006 American Control Conference*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, 2006.

[7] Kumar, M., Singla, P., Chakravorty, S., and Junkins, J. L., "A Multi-Resolution Approach for Steady State Uncertainty Determination in Nonlinear Dynamic Systems," *38th Southeastern Symposium on System Theory*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, 2006, pp. 344–348.

[8] Kumar, M., Singla, P., Chakravorty, S., and Junkins, J. L., "The Partition of Unity Finite Element Approach to the Stationary Fokker–Planck Equation," AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Keystone, CO, AIAA Paper 2006-6285, Aug. 2006.

[9] Muscolino, G., Ricciardi, G., and Vasta, M., "Stationary and Nonstationary Probability Density Function for Nonlinear Oscillators," *International Journal of Non-Linear Mechanics*, Vol. 32, No. 6,

Nov. 1997, pp. 1051–1064.
doi:10.1016/S0020-7462(96)00134-5

[10] Paola, M. D., and Sofi, A., "Approximate Solution of the Fokker–Planck-Kolmogorov Equation," *Probabilistic Engineering Mechanics*, Vol. 17, No. 4, Oct. 2002, pp. 369–384.
doi:10.1016/S0266-8920(02)00034-6

[11] Anderson, B. D., and Moore, J. B., *Optimal Filtering*, Prentice–Hall, Upper Saddle River, NJ, 1979, pp. 193–222.

[12] Doucet, A., de Freitas, N., and Gordon, N., *Sequential Monte Carlo Methods in Practice*, Springer–Verlag, New York, Apr. 2001, pp. 6–14.

[13] Iyengar, R. N., and Dash, P. K., "Study of the Random Vibration of Nonlinear Systems by the Gaussian Closure Technique," *Journal of Applied Mechanics*, Vol. 45, June 1978, pp. 393–399.

[14] Roberts, J. B., and Spanos, P. D., *Random Vibration and Statistical Linearization*, Wiley, New York, 1990, pp. 122–176.

[15] Lefebvre, T., Bruyninckx, H., and Schutter, J. D., "Kalman Filters of Non-Linear Systems: A Comparison of Performance," *International Journal of Control*, Vol. 77, No. 7, 2004, pp. 639–653.
doi:10.1080/00207170410001704998

[16] Lefebvre, T., Bruyninckx, H., and Schutter, J. D., "Comment on A New Method for the Nonlinear Transformations of Means and Covariances in Filters and Estimators," *IEEE Transactions on Automatic Control*, Vol. 47, No. 8, 2002, pp. 1406–1409.

[17] Alspach, D., and Sorenson, H., "Nonlinear Bayesian Estimation Using Gaussian Sum Approximations," *IEEE Transactions on Automatic Control*, Vol. 17, No. 4, 1972, pp. 439–448.
doi:10.1109/TAC.1972.1100034

[18] Ito, K., and Xiong, K., "Gaussian Filters for Nonlinear Filtering Problems," *IEEE Transactions on Automatic Control*, Vol. 45, No. 5, 2000, pp. 910–927.
doi:10.1109/9.855552

[19] Gelb, A., *Applied Optimal Estimation*, MIT Press, Cambridge, MA, 1974, 203–214.

[20] Petersen, K. B., and Pedersen, M. S., *The Matrix Cookbook*, http://matrixcookbook.com/ [retrieved Dec. 2007].

[21] Julier, S., and Uhlmann, J., "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE*, Vol. 92, No. 3, 2004, pp. 401–422.
doi:10.1109/JPROC.2003.823141

[22] Honkela, A., "Approximating Nonlinear Transformations of Probability Distributions for Nonlinear Independent Component Analysis," *IEEE International Joint Conference on Neural Networks*, Vol. 3, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, 2004, pp. 2169–2174.

[23] Wu, Y., Wu, M., Hu, D., and Hu, X., "An Improvement to Unscented Transformation," *17th Australian Joint Conference on Artificial Intelligence*, Springer, Berlin, 2004, pp. 1024–1029.

[24] Boyd, S., and Vandenberghe, L., *Convex Optimization*, Cambridge Univ. Press, New York, 2004, Vol. 67–78, pp. 152–159.

[25] Singla, P., and Singh, T., "A Gaussian Function Network for Uncertainty Propagation Through Nonlinear Dynamic System," *18th AAS/AIAA Spaceflight Mechanics Meeting*, American Astronautical Society, Springfield, VA, 27–31 Jan. 2008, pp. 851–864.