A Decision-Centric Framework for Density Forecasting

GABRIEL TEREJANU
PUNEET SINGLA
TARUNRAJ SINGH
PETER D. SCOTT

In general, the uncertainty propagation problem, in which the uncertain initial condition evolves through a dynamic system driven by noise, is seen strictly from the producer's perspective. This means that uncertainty propagation algorithms are derived and evaluated based on statistical measures independent of the user's decision needs. However accurate the uncertainty evolution given by a particular method, it may be less than optimal to the user or the decision maker, who takes decisions based on an implicit or explicit utility function. While in a static environment, one may be able to select an appropriate method for uncertainty propagation, in a dynamic environment with an ever-changing utility function this becomes a challenging task.

The goal of the present work is to reconcile the two views into a decision-centric framework which provides both a more accurate approximation to the relevant probability density function and a more precise expected utility value for the decision maker. A numerical example using a puff-based dispersion model, for forecasting downwind concentrations of toxic materials, demonstrates the capacity of this approach to focus computational resources on regions of particular interest such as high population density. A second example shows improvement over alternative methods as measured by a variety of utility-weighted metrics.

Manuscript received October 6, 2009; revised February 3, 2010, May 7, 2010; released for publication May 11, 2010.

Refereeing of this contribution was handled by Peter Willett.

Authors' address: G. Terejanu and P. D. Scott, Department of Computer Science & Engineering, University at Buffalo, Buffalo, NY 14260, E-mail: (terejanu@buffalo.edu, peter@buffalo.edu); P. Singla and T. Singh, Department of Mechanical & Aerospace Engineering, University at Buffalo, Buffalo, NY 14260, E-mail: (psingla@buffalo.edu, tsingh@buffalo.edu).

1. INTRODUCTION

Decision makers increasingly rely on mathematical models in choosing the right set of actions in critical situations. The accuracy of mathematical models in predicting the physical state of the system directly affects the accuracy of the decision making process. Such situations are often encountered in deployment of emergency responders in response to extreme events such as covert release of hazardous material, storm surge due to a hurricane, wild fire, etc. Disaster response managers routinely use numerical modeling to assist in hazard response and mitigation. However, any numerical model used to forecast physical state variables and assist in decision making is a reflection of numerous assumptions and simplifications to permit the determination of a tractable model. The error inherent in any model is a result of model truncation, errors in model parameters, and errors in initial and boundary conditions. Together these factors cause overall prediction model accuracy to degrade as the simulation evolves. Hence, it is important to forecast the evolution of a physical state variable with its attendant uncertainty given the uncertainties in the inputs to the numerical model. Based on the forecast of physical state and associated uncertainty, decisions can be made on deploying emergency responders, evacuating cities, sheltering or medical gear caching.

The optimal decision under uncertainty corresponds to maximizing the expected value of a utility function or minimize the expected value of a loss function [40]. The utility or its complement, the loss function, are defined to measure the consequences of the decision making process. The accurate computation of the expected loss requires the knowledge of the probability distribution of the physical state variable due to model and input uncertainties. The exact time evolution of state probability density function (pdf) is given by the Fokker-Planck-Kolmogorov Equation (FPKE) [27].

If FPKE could be solved for the state pdf, it would be possible to calculate statistical moments like the mean state and the error covariance at different times as well as different expectations such as the expected loss. Analytical solutions for the FPKE exist only for a stationary pdf and are restricted to a limited class of dynamical systems [9, 27]. Thus researchers are actively looking at numerical approximations to solve the FPKE [15–17, 20, 23], generally using the variational formulation of the problem. However, these methods are severely handicapped for even low dimensions because the discretization of the space over which the pdf lives is, computationally impractical.

To emulate the exact methods, many approximate techniques exist in the literature to approximate the uncertainty evolution problem, the most popular being Monte Carlo (MC) methods [8], Gaussian closure [12], Equivalent Linearization [28], and Stochastic Averaging [18, 19]. 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. The Markov Chain Monte Carlo (MCMC) or sequential Monte Carlo methods [31] are other attractive alternatives in the case of low-order nonlinear systems to solve the FPKE. Their applicability to higher-order systems, particularly in "plain-vanilla" forms, is limited by their high computational complexity and sensitivity to properties such as the rate of decay of the conditional pdf. As noted by Daum [7], sequential Monte Carlo methods are not immune to the "curse of dimensionality," and their effective use should take into account the smoothness constraint implied by the FPKE.

Recently Terejanu et al. [36] have proposed the Gaussian mixture model for accurately solving the FPKE in a computationally effective manner. The key idea is to approximate the state pdf by a finite sum of Gaussian density functions whose mean and covariance are propagated using linear theory. The weights corresponding to different Gaussian kernels are updated by requiring the mixture to satisfy the FPKE [36]. With this formulation, the mixture problem can be solved efficiently and accurately using convex optimization solvers, even if the mixture model includes many terms. Another advantage of the proposed method is that it decouples a large uncertainty characterization problem into many small scale problems. As a consequence, the algorithm can be parallelized on today's high performance computing systems. Although Gaussian mixture idea has been successfully applied to low and moderate dimension systems ($n = O(10)$), including the uncertainty propagation through two-body system and toxic cloud transported by wind [10, 36, 37], like any other method to solve the FPKE it only provides an approximate description of the uncertainty propagation problem by restricting the FPKE solution space to a small number of parameters.

In general, the uncertainty evolution process does not take into account the knowledge about the decision making process. Evaluation of the approximate state pdf provided by different methods is based on statistical measures, such as minimization of FPKE error, integral square error between true pdf and its approximation [36], mean square error [14] or expected exponential of estimation error [4, 29]. This process is independent of the user's decision needs and is referred here as the producer's perspective where the accuracy of the forecast is the main driver in the algorithm evaluation [14]. These assumptions make the problem tractable and computationally efficient, which satisfies the requirement of minimizing decision latency, but the approximations may be of little use when computing the expected loss, since they are not sensitive to the decision maker's loss function [37]. For example, an approximation which underestimates a tail of the forecast pdf where the main support of the loss function resides.

Ideally the uncertainty evolution should be performed from the user's perspective [25], i.e., it should take into account the structure of the utility or loss function. While in a static environment, one may be able to select an appropriate method for uncertainty propagation, in a dynamic environment with an ever-changing utility function this becomes a challenging task. The main objective of this work is to reconcile the two views into a decision-centric framework which provides both a more accurate approximation to the relevant state probability density function and a more precise expected utility value for the decision maker. This is achieved by incorporating contextual loss information held by the decision maker into the density forecasting process.

We use a Gaussian mixture approximation to the state pdf and propose a "non-intrusive" way of computing an approximate pdf that addresses the region of interest and is closer to the true pdf in the sense of minimizing FPKE error. Non-intrusive refers here to the fact the we do not require a new uncertainty propagation method when incorporating the loss function into the derivation. The interaction level between the Decision Maker (DM) and Density Forecasting (DF) is acting at the process refinement level which manages the resources of the density forecasting method, in this case the location of the Gaussian components.

A progressive selection method is designed to add new Gaussian components to the initial Gaussian mixture, such that probabilistic support is reaching the region of interest at the decision time. The initial weights of the added Gaussian components are set to zero and they are modified when propagated throughout the nonlinear dynamic system to minimize the error in the FPKE [36]. Therefore, if there is any probability density mass in the region of interest it will be represented by the non-zero weight of the new Gaussian components at the decision time.

We mention that the similar ideas have been explored in risk sensitive particle filters [39], which are not to be confused with risk sensitive filters [4, 29]. The risk sensitive particle filter modifies the sampling density of the standard particle filter so that more samples are generated in high risk regions of the state space. This is achieved with a risk function obtained using a Markov decision process to approximate the future risk of decisions from a particular state.

The structure of the paper is as follows: first the decision making problem is stated in Section 2 and the Gaussian Sum approximation to the forecast pdf is presented in Section 3. The progressive selection of Gaussian components is derived in Section 4 followed by two numerical examples in Section 5 to motivate and to illustrate the performance of the method. The conclusions and future work are discussed in Section 6.

2. PROBLEM STATEMENT

Consider a general n-dimensional continuous-time noise driven nonlinear dynamic system with uncertain

initial conditions given by the following equations:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{f}(t,\mathbf{x}(t)) + \mathbf{\Gamma}(t)\\ \mathbf{x}(t_0) &\sim p(t_0,\mathbf{x}_0)\end{aligned} \tag{1}$$

where $\mathbf{\Gamma}(t)$ represents a Gaussian white noise process with the correlation function $\mathbf{Q}\delta(t-\tau)$, and uncorrelated with the initial condition.

We are interested in finding the forecast probability density function $p(t,\mathbf{x}(t))$ whose time-evolution is given by the following partial differential equation known as the Fokker-Planck-Kolmogorov Equation (FPKE) [27]:

$$\begin{aligned}\frac{\partial}{\partial t}p(t,\mathbf{x}) = &-\frac{\partial p(t,\mathbf{x})^T}{\partial \mathbf{x}}\mathbf{f}(t,\mathbf{x}) - p(t,\mathbf{x})\mathrm{Tr}\left[\frac{\partial \mathbf{f}(t,\mathbf{x})}{\partial \mathbf{x}}\right]\\ &+\frac{1}{2}\mathrm{Tr}\left[\mathbf{Q}\frac{\partial^2 p(t,\mathbf{x})}{\partial \mathbf{x}\partial \mathbf{x}^T}\right].\end{aligned} \tag{2}$$

Given a state space region of interest at a particular decision time, t_d, which may be represented as a *loss function* by the decision maker, $L(\mathbf{x}_d,a_d)$, the expected loss of an action a_d is calculated as follows:

$$L(a_d) = \int L(\mathbf{x}_d,a_d)p(t_d,\mathbf{x}_d)d\mathbf{x}_d \tag{3}$$

where $\mathbf{x}_d$ is the state of the system at decision time, $t = t_d$.

If the FPKE in (2) can be solved exactly for the forecast pdf, $p(t_d,\mathbf{x}_d)$, it would be possible to obtain the expected loss and also find the optimal Bayesian decision [24], if a set of decisions exists. Although analytical steady state solutions for the FPKE exist for a limited class of dynamical systems, finding the solution for the generic nonlinear system in (1) is not a trivial task. In practice, we only know a numerical approximation to the state pdf $p(t_d,\mathbf{x}_d)$, denoted by $\hat{p}(t_d,\mathbf{x}_d)$. As a consequence of this, we can only compute an approximated value of the expected loss and hence optimal decision:

$$\hat{L}(a_d) = \int L(\mathbf{x}_d,a_d)\hat{p}(t_d,\mathbf{x}_d)d\mathbf{x}_d \tag{4}$$

$$\hat{a}_d = \arg\min_{a_d}\int L(\mathbf{x}_d,a_d)\hat{p}(t_d,\mathbf{x}_d)d\mathbf{x}_d. \tag{5}$$

The decision making process in the density forecasting context is presented in Fig. 1 (left). Obviously if we have a good approximation for the forecast pdf in the region of interest the same can be said for the expected loss. This situation becomes more dramatic when a large deviation exists between the actual and the estimated forecast pdf in the region of interest. In the case of evaluation of a single decision, the algorithm may underestimate the actual expected loss, $\hat{L}(a_d) \ll L(a_d)$, or overestimate it, misguiding the decision maker with respect to the magnitude of the situation. In the case when a optimal decision has to be chosen, the large

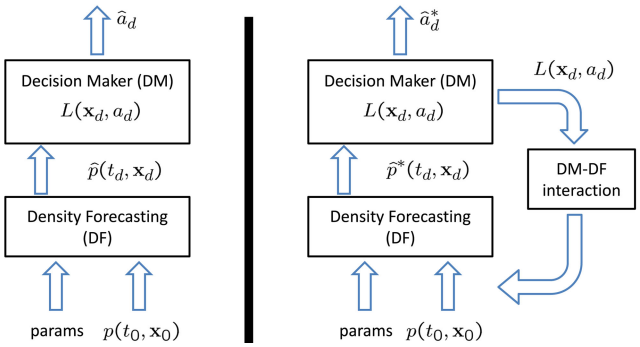

Fig. 1. Left figure represents the classic approach to decision making in the density forecasting context. The right figure shows the proposed model.

difference between forecast pdfs may result in picking not only a suboptimal decision but a very consequential one.

While one can derive a new method to approximate the forecast pdf by including the loss function in the derivation and overweighting errors in the region of interest to better approximate the expected loss, it will accomplish this at the expense of worsening the global approximation of the pdf. This will coarsen other estimates based on the forecast pdf, independent of the utility function, such as the mean of the pdf, the modes of the pdf, etc. The loss in global accuracy in estimating these statistics may end up misleading the decision maker with respect to the dominant behavior of the system.

In other words, if we name the computation of the expected loss of a given action as *impact assessment* and the computation of the moments and other quantities based on the pdf as *situation assessment*, one will require that both to be as accurate as possible. At the limit, if we can compute exactly the forecast pdf we accurately obtain both *impact assessment* and *situation assessment* since we can quantify exactly the probability of all the outcomes. The proposed decision-centric framework for density forecasting is in agreement with the information flow across the fusion levels of the JDL model proposed in [32]. Both an upward flow and a downward flow is necessary to obtain relevant inferences.

Since the decision maker holds important information regarding the use of the pdf obtained from the density forecasting method, we can incorporate this information in the uncertainty propagation process in a non-intrusive manner (do not have to derive a new method), by supplementing the inputs into the density forecasting module. The proposed method is shown in Fig. 1 (right), where a new interaction level is introduced between the decision maker and the uncertainty propagation, that uses the contextual information provided by the decision maker to supplement the inputs of the density forecasting process. In other words the proposed method changes the environment in which the density forecasting method is running.

Therefore, we want to find an approximation to the forecast pdf, $\hat{p}^*(t_d, \mathbf{x}_d)$, that addresses the interest held by the decision maker and provides both a better impact and situation assessment than $\hat{p}(t_d, \mathbf{x}_d)$. These objectives can be captured by the following two relations:

$$\int |p(t_d, \mathbf{x}_d) - \hat{p}^*(t_d, \mathbf{x}_d)|^2 d\mathbf{x}_d \leq \int |p(t_d, \mathbf{x}_d) - \hat{p}(t_d, \mathbf{x}_d)|^2 d\mathbf{x}_d \tag{6}$$

$$|\hat{L}^*(a_d) - L(a_d)| \leq |\hat{L}(a_d) - L(a_d)|. \tag{7}$$

In the present paper, we will design an interaction level between the decision maker and the uncertainty propagation module that approximates the pdf using a Gaussian mixture. The interaction level is adding new Gaussian components to the initial uncertainty, such that they will be positioned near the region of interest at the decision time. Their initial weights will be set to zero, thus the initial uncertainty is not changed, but the evolution of the weights is dictated by the error in the FPKE as in the Adaptive Gaussian Sum algorithm used to propagate the uncertainty in [36]. Thus if any probability density mass is moving naturally towards the region of interest, the weights of the new Gaussian components will become greater than zero. Therefore the method will find if there is any probability density mass in the region of interest.

In this paper we will consider only the forecast of the pdf when no measurements are available between the current time and the decision time. A suggestion, on how this can be used in the case when we have observations to assimilate between the current time and the decision time, is given in Section 4. In the following section we present the uncertainty propagation method for DF and in Section 4, the algorithm in the DM-DF interaction level is derived.

3. APPROXIMATION OF THE FORECAST PROBABILITY DENSITY FUNCTION

In this section, we briefly summarize the Gaussian mixture model approach to solve the FPKE; more details can be found in our prior work [30, 36]. The main idea of this approach is to approximate the state pdf by a finite sum of Gaussian density functions whose mean and covariance are propagated using linear theory. The weights corresponding to different Gaussian kernels are updated by requiring the mixture to satisfy the FPKE [36].

Let us consider the following equation depicting the Gaussian mixture model approximation for the forecast density function, $p(t, \mathbf{x})$:

$$\begin{aligned} \hat{p}(t, \mathbf{x}) &= \sum_{i=1}^{N} w_t^i \underbrace{\mathcal{N}(\mathbf{x}(t); \boldsymbol{\mu}_t^i, \mathbf{P}_t^i)}_{p_{g_i}} \\ \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_t^i, \mathbf{P}_t^i) &= |2\pi \mathbf{P}_t^i|^{-1/2} \\ &\quad \times \exp[-\tfrac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_t^i)^T (\mathbf{P}_t^i)^{-1} (\mathbf{x} - \boldsymbol{\mu}_t^i)] \end{aligned} \tag{8}$$

where $\boldsymbol{\mu}_t^i$ and $\mathbf{P}_t^i$ represent the mean and covariance of the ith component of the Gaussian pdf, and w_t^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:

$$\sum_{i=1}^{N} w_t^i = 1, \qquad w_t^i \geq 0, \qquad \forall t \tag{9}$$

In [2], it is shown that since all the components of the mixture pdf of (8) are Gaussian and thus, only estimates of their mean and covariance need to be maintained, they can be propagated between t and $t' = t + \Delta t$ using the linear system propagation methods such as the Extended Kalman Filter (EKF):

$$\dot{\boldsymbol{\mu}}_t^i = \mathbf{f}(t, \boldsymbol{\mu}_t^i) \tag{10}$$

$$\dot{\mathbf{P}}_t^i = \mathbf{A}_t^i \mathbf{P}_t^i + \mathbf{P}_t^i (\mathbf{A}_t^i)^T + \mathbf{Q} \tag{11}$$

$$\mathbf{A}_t^i = \left. \frac{\partial \mathbf{f}(t, \mathbf{x}(t))}{\partial \mathbf{x}(t)} \right|_{\mathbf{x}(t) = \boldsymbol{\mu}_t^i}. \tag{12}$$

Although, in this paper we present only the EKF model to propagate the mean and covariance of each of the Gaussian component, one can easily use some advanced linear propagation methods like unscented Kalman filter [13] or quasi-Gaussian Kalman filter [5] to propagate the mean and covariance more accurately.

The weights of the Gaussian components are not known and must be computed as part of the solution process. Using the following approximation for the total derivative of the weights, $\dot{w}_t^i = (1/\Delta t)(w_{t'}^i - w_t^i)$, the unknown weights $w_{t'}^i$ are found by minimizing the integral square FPKE error as discussed in [35, 36]. Substituting (8) in (2) leads to,

$$\begin{aligned} e(t, \mathbf{x}) &= \frac{\partial}{\partial t} \hat{p}(t, \mathbf{x}) + \frac{\partial \hat{p}(t, \mathbf{x})^T}{\partial \mathbf{x}} \mathbf{f}(t, \mathbf{x}) + \hat{p}(t, \mathbf{x}) \mathrm{Tr}\left[\frac{\partial \mathbf{f}(t, \mathbf{x})}{\partial \mathbf{x}} \right] \\ &\quad - \frac{1}{2} \mathrm{Tr}\left[\mathbf{Q} \frac{\partial^2 \hat{p}(t, \mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} \right] \\ &= \frac{1}{\Delta t} \sum_{i=1}^{N} p_{g_i} w_{t'}^i \\ &\quad + \sum_{i=1}^{N} \left(\frac{\partial p_{g_i}^T}{\partial \boldsymbol{\mu}_t^i} \dot{\boldsymbol{\mu}}_t^i + \mathrm{Tr}\left[\frac{\partial p_{g_i}}{\partial \mathbf{P}_t^i} \dot{\mathbf{P}}_t^i \right] - \frac{1}{\Delta t} p_{g_i} \right. \\ &\qquad + \frac{\partial p_{g_i}^T}{\partial \mathbf{x}} \mathbf{f}(t, \mathbf{x}) + p_{g_i} \mathrm{Tr}\left[\frac{\partial \mathbf{f}(t, \mathbf{x})}{\partial \mathbf{x}} \right] \\ &\qquad \left. - \frac{1}{2} \mathrm{Tr}\left[\mathbf{Q} \frac{\partial^2 p_{g_i}}{\partial \mathbf{x} \partial \mathbf{x}^T} \right] \right) w_t^i. \end{aligned} \tag{13}$$

Since the FPKE error of (13) is linear in Gaussian weights, the integral square FPKE error minimization problem can be written as the following quadratic pro-

gramming problem:

$$\begin{aligned} \min_{w^i_{t'}} \quad & \tfrac{1}{2}\mathbf{w}_{t'}^T\mathbf{M}_\mathbf{c}\mathbf{w}_{t'} + \mathbf{w}_{t'}^T\mathbf{N}_\mathbf{c}\mathbf{w}_t \\ \text{s.t} \quad & \mathbf{1}_{N\times 1}^T\mathbf{w}_{t'} = 1 \\ & \mathbf{w}_{t'} \geq \mathbf{0}_{N\times 1} \end{aligned} \tag{14}$$

where $\mathbf{w}_t \in \mathbb{R}^{N\times 1}$ is the vector of weights at time t, $\mathbf{w}_{t'} \in \mathbb{R}^{N\times 1}$ is the vector of unknown weights at time t', $\mathbf{1}_{N\times 1} \in \mathbb{R}^{N\times 1}$ is a vector of ones, $\mathbf{0}_{N\times 1} \in \mathbb{R}^{N\times 1}$ is a vector of zeros and the components of the two matrices $\mathbf{M}_\mathbf{c} \in \mathbb{R}^{N\times N}$ and $\mathbf{N}_\mathbf{c} \in \mathbb{R}^{N\times N}$ are given by

$$\begin{aligned} m_{c_{ij}} = & \frac{1}{\Delta t^2}|2\pi(\mathbf{P}_t^i + \mathbf{P}_t^j)|^{-1/2} \\ & \times \exp\left[-\frac{1}{2}(\boldsymbol{\mu}_t^i - \boldsymbol{\mu}_t^j)^T(\mathbf{P}_t^i + \mathbf{P}_t^j)^{-1}(\boldsymbol{\mu}_t^i - \boldsymbol{\mu}_t^j)\right] \\ & \qquad\qquad \text{for} \quad i \neq j \end{aligned} \tag{15}$$

$$m_{c_{ii}} = \frac{1}{\Delta t^2}|4\pi\mathbf{P}_t^i|^{-1/2} \qquad \text{for} \quad i = j \tag{16}$$

and,

$$\begin{aligned} n_{c_{ij}} = \frac{1}{\Delta t}p_{g_i}\int_V \Bigg(& \frac{\partial p_{g_j}^T}{\partial \boldsymbol{\mu}_t^j}\dot{\boldsymbol{\mu}}_t^j + \mathrm{Tr}\left[\frac{\partial p_{g_j}}{\partial \mathbf{P}_t^j}\dot{\mathbf{P}}_t^j\right] - \frac{1}{\Delta t}p_{g_j} \\ & + \frac{\partial p_{g_j}^T}{\partial \mathbf{x}}\mathbf{f}(t,\mathbf{x}) + p_{g_j}\mathrm{Tr}\left[\frac{\partial \mathbf{f}(t,\mathbf{x})}{\partial \mathbf{x}}\right] \\ & - \frac{1}{2}\mathrm{Tr}\left[\mathbf{Q}\frac{\partial^2 p_{g_j}}{\partial \mathbf{x}\partial \mathbf{x}^T}\right]\Bigg) d\mathbf{x}. \end{aligned} \tag{17}$$

Details on the derivation of the above relations can be found in [36, 38]. 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. By updating the forecast weights, not only can we obtain a more accurate estimate but also a better approximation to the forecast probability density function [35].

The estimated pdf is used to compute the expected loss. We require that the loss function provided is positive, finite everywhere and it is able to distinguish the important states from the unimportant ones. For simplicity the loss function used in this work has the following form:

$$L(\mathbf{x}_d, a_d) = \mathcal{N}(\mathbf{x}_d; \boldsymbol{\mu}_L, \boldsymbol{\Sigma}_L). \tag{18}$$

Due to the approximations used in propagating the pdf it may happen that no or very little probability density mass exists in the region of interest at the decision time, depicted here by the loss function. In the following section we present an algorithm which adds new Gaussian kernels to the initial mixture such that they will be positioned in the region of interest defined by the loss function at the decision time, increasing the accuracy of the expected loss.

4. DECISION MAKER–DENSITY FORECASTING INTERACTION LEVEL

The iterative method proposed here, is adding a set of Gaussian components to the initial pdf that are sensitive to the loss function at the decision time. After propagation, these Gaussian components will be located near the center of support of the loss function at the decision time. Initially the weights of these components are set to zero, and they will be updated in the propagation step, using the method in Section 3, if any probability density mass is moving in their direction. The weights at the decision time will give their relative contributions in computing the expected loss with respect to the entire pdf.

An algorithm, called the Progressive Selection of Gaussian Components (PSGC), that bears similarity to the simulated annealing and the progressive correction used in particle filters [21], is proposed in selecting the initial Gaussian components sensitive to the loss function. The means of the new Gaussian components will be sampled from a proposed distribution, $p_{\mathrm{Smp}}(t_0, \mathbf{x}_0)$, which is recursively constructed to be sensitive to the contextual loss function. The support of the proposal distribution or sampling pdf is gradually mapped into a region that covers the support of the loss function at decision time.

The main idea in constructing the sampling pdf is as follows: initially set the sampling pdf equal to the uncertain initial condition in (1), select the means and covariances of a set of Gaussian components based on this distribution, propagate each one of them using the time update equations in the Extended Kalman Filter, Eqs. (10)–(11), until the decision time is reached, and based on the contributions to the expected loss find their corresponding weights. The new sampling pdf is just the weighted sum of the of the initially selected Gaussian components. The sampling process is repeated until all the Gaussian components are located in the support region of the loss function at the decision time. The remainder of the section details the derivation of this procedure.

Let initially the sample pdf, $p_{\mathrm{Smp}}(t_0, \mathbf{x}_0)$, to be equal to the initial uncertainty given by $p(t_0, \mathbf{x}_0)$, which is modeled using a Gaussian sum as in (8). Compute the mean and the variance of the sample pdf:

$$\boldsymbol{\mu}_0 = E[\mathbf{x}_0] = \int \mathbf{x}_0 p_{\mathrm{Smp}}(t_0, \mathbf{x}_0) d\mathbf{x}_0 \tag{19}$$

$$\begin{aligned} \mathbf{P}_0 &= E[(\mathbf{x}_0 - \boldsymbol{\mu}_0)(\mathbf{x}_0 - \boldsymbol{\mu}_0)^T] \\ &= \int (\mathbf{x}_0 - \boldsymbol{\mu}_0)(\mathbf{x}_0 - \boldsymbol{\mu}_0)^T p_{\mathrm{Smp}}(t_0, \mathbf{x}_0) d\mathbf{x}_0. \end{aligned} \tag{20}$$

For the first iteration the above two moments are computed as follows:

$$\boldsymbol{\mu}_0 = \sum_{i=1}^{N} w_0^i \boldsymbol{\mu}_0^i \tag{21}$$

$$\mathbf{P}_0 = \sum_{i=1}^{N} w_0^i [\mathbf{P}_0^i + (\boldsymbol{\mu}_0^i - \boldsymbol{\mu}_0)(\boldsymbol{\mu}_0^i - \boldsymbol{\mu}_0)^T]. \tag{22}$$

Assume that we want to add another M new Gaussian components to the initial pdf with zero weights and sensitive to the loss function. We sample the means of these Gaussian components from the proposal distribution such that their equally weighted sum gives the mean in (21).

$$\boldsymbol{\mu}^i \sim p_{\text{Smp}}(t_0, \mathbf{x}_0) \qquad \text{for} \quad i = 1 \ldots M-1 \tag{23}$$

$$\boldsymbol{\mu}^M = M\boldsymbol{\mu}_0 - \sum_{i=1}^{M-1} \boldsymbol{\mu}^i. \tag{24}$$

The default covariance of the Gaussian components is $\mathbf{D}$. We want to find the new covariance $\mathbf{D}^*$ such that the covariance of the new Gaussian components matches the covariance of the sample pdf, $\mathbf{P}_0$. Let $\mathbf{D}^* = \gamma\mathbf{D}$. Thus we want to find γ such that we minimize the following expression:

$$J_\gamma = \text{Tr}\left[\mathbf{P}_0 - \frac{1}{M}\sum_{i=1}^{M}(\gamma\mathbf{D} + (\boldsymbol{\mu}^i - \boldsymbol{\mu}_0)(\boldsymbol{\mu}^i - \boldsymbol{\mu}_0)^T)\right] \tag{25}$$

$$\gamma = \frac{1}{\text{Tr}[\mathbf{D}]}\text{Tr}\left[\mathbf{P}_0 - \frac{1}{M}\sum_{i=1}^{M}(\boldsymbol{\mu}^i - \boldsymbol{\mu}_0)(\boldsymbol{\mu}^i - \boldsymbol{\mu}_0)^T\right]. \tag{26}$$

Only solutions $\gamma > 0$ are accepted. Otherwise we repeat the sampling of the means, starting with (23). Once we have the first two moments of the new Gaussian components, we propagate them using the time update equations in the Extended Kalman Filter, Eqs. (10)–(11), until we reach the decision time. Let $\boldsymbol{\mu}_d^i$ and $\mathbf{P}_d^i$ be their means and covariances at the decision time, $t = t_d$. The Gaussian components will then be weighted based on their contribution to the expected loss. A larger contribution means a more sensitive component to the loss function, thus a larger weight.

To be able to compute the weights of the Gaussian components, make sure that all of them are fairly weighted, we are not running into numerical problems and also create an indicator to mark the end of the algorithm, we compute an inflation coefficient for the loss function. Let $\boldsymbol{\Sigma}_L^* = \alpha\boldsymbol{\Sigma}_L$ be the inflated covariance of the loss function.

The inflation coefficient α is found such that the expected loss computed using the most distant Gaussian component from the loss function is maximized. Let the mean and the covariance of the most distant component be denoted by $\boldsymbol{\mu}_d^{\max}$ and $\mathbf{P}_d^{\max}$ respectively.

$$\begin{aligned} J_{\max} &= \int \mathcal{N}(\mathbf{x}_d; \boldsymbol{\mu}_L, \alpha\boldsymbol{\Sigma}_L)\mathcal{N}(\mathbf{x}_d; \boldsymbol{\mu}_d^{\max}, \mathbf{P}_d^{\max})d\mathbf{x}_d \\ &= \mathcal{N}(\boldsymbol{\mu}_L; \boldsymbol{\mu}_d^{\max}, \alpha\boldsymbol{\Sigma}_L + \mathbf{P}_d^{\max}). \end{aligned} \tag{27}$$

An equivalent way to seek α is by minimizing the negative logarithm of the above expectation.

$$\begin{aligned} J_{\min} &= \log[\det(\alpha\boldsymbol{\Sigma}_L + \mathbf{P}_d^{\max})] \\ &\quad + (\boldsymbol{\mu}_L - \boldsymbol{\mu}_d^{\max})^T(\alpha\boldsymbol{\Sigma}_L + \mathbf{P}_d^{\max})^{-1}(\boldsymbol{\mu}_L - \boldsymbol{\mu}_d^{\max}) \end{aligned} \tag{28}$$

Let us denote $\mathbf{K} = \alpha\boldsymbol{\Sigma}_L + \mathbf{P}_d^{\max}$ and $\mathbf{U} = (\boldsymbol{\mu}_L - \boldsymbol{\mu}_d^{\max}) \cdot (\boldsymbol{\mu}_L - \boldsymbol{\mu}_d^{\max})^T$. We seek $\alpha > 0$ such that

$$\frac{\partial J_{\min}}{\partial \alpha} = 0 \tag{29}$$

$$\text{Tr}[\mathbf{K}^{-1}\boldsymbol{\Sigma}_L - \mathbf{K}^{-1}\mathbf{U}\mathbf{K}^{-1}\boldsymbol{\Sigma}_L] = 0. \tag{30}$$

After a few mathematical manipulations, (30) can be written in the following format:

$$\text{Tr}[\mathbf{K}^{-1}\boldsymbol{\Sigma}_L(\alpha\mathbf{I} + \mathbf{P}_d^{\max}\boldsymbol{\Sigma}_L^{-1} - \mathbf{U}\boldsymbol{\Sigma}_L^{-1})\mathbf{K}^{-1}\boldsymbol{\Sigma}_L] = 0. \tag{31}$$

Using the following notation, $\mathbf{A} = \mathbf{K}^{-1}\boldsymbol{\Sigma}_L$ and $\mathbf{B} = \alpha\mathbf{I} + \mathbf{P}_d^{\max}\boldsymbol{\Sigma}_L^{-1} - \mathbf{U}\boldsymbol{\Sigma}_L^{-1}$, (31) can be written as Tr[**ABA**] = 0. Observe that for $\alpha > 0$ the matrix $\mathbf{A}$ is symmetric and positive definite. Hence, by applying Lemma 1 from Appendix A to (31) we get,

$$\text{Tr}[\alpha\mathbf{I} + \mathbf{P}_d^{\max}\boldsymbol{\Sigma}_L^{-1} - \mathbf{U}\boldsymbol{\Sigma}_L^{-1}] = 0. \tag{32}$$

Therefore we accept solutions $\alpha > 1$ that satisfy the following relation

$$\alpha = \frac{1}{n}\text{Tr}[(\mathbf{U} - \mathbf{P}_d^{\max})\boldsymbol{\Sigma}_L^{-1}]. \tag{33}$$

For $\alpha \leq 1$ we stop the algorithm, because all the Gaussian components, including the most distant one, are located near the center of support of the loss function. Otherwise, α is used to compute the inflated covariance $\boldsymbol{\Sigma}_L^* = \alpha\boldsymbol{\Sigma}_L$, and the weights of the Gaussian components are obtained based on their approximation to the loss function by solving the following optimization problem:

$$\mathbf{w} = \arg\min_{\mathbf{w}} \frac{1}{2}\int \left(\mathcal{N}(\mathbf{x}_d; \boldsymbol{\mu}_L, \boldsymbol{\Sigma}_L^*) - \sum_{i=1}^{M} w^i \mathcal{N}(\mathbf{x}_d; \boldsymbol{\mu}_d^i, \mathbf{P}_d^i)\right)^2 d\mathbf{x}_d. \tag{34}$$

The optimization in (34) is equivalent to solving the following quadratic programming problem:

$$\begin{aligned} \mathbf{w} &= \arg\min_{\mathbf{w}} \tfrac{1}{2}\mathbf{w}^T\mathbf{M}\mathbf{w} - \mathbf{w}^T\mathbf{N} \\ &\text{subject to} \quad \mathbf{1}_{M\times 1}^T\mathbf{w} = 1 \\ &\mathbf{w} \geq \mathbf{0}_{M\times 1} \end{aligned} \tag{35}$$

where $\mathbf{w} \in \mathbb{R}^{M\times 1}$ is the vector of weights and the entries of $\mathbf{M} \in \mathbb{R}^{M\times M}$ and $\mathbf{N} \in \mathbb{R}^{M\times 1}$ are given by:

$$m_{ij} = \mathcal{N}\{\boldsymbol{\mu}_d^j; \boldsymbol{\mu}_d^i, \mathbf{P}_d^i + \mathbf{P}_d^j\} \tag{36}$$

$$n_i = \mathcal{N}\{\boldsymbol{\mu}_L; \boldsymbol{\mu}_d^i, \mathbf{P}_d^i + \boldsymbol{\Sigma}_L^*\}. \tag{37}$$

Given the weights found in (35), the new pdf used to sample the new means is given by,

$$p_{\text{Smp}}(t_0, \mathbf{x}_0) = \sum_{i=1}^{M} w^i \mathcal{N}(\mathbf{x}_0; \boldsymbol{\mu}^i, \beta\mathbf{D}^*) \tag{38}$$

where $\beta \leq 1$ is a coefficient that controls the decrease of the initial variance. If α has decreased from the previous iteration this means that the Gaussian components are getting closer to the loss function and therefore we can decrease the variance of the initial distribution to finely tune the position of the Gaussian components, otherwise $\beta = 1$. The process is repeated starting with (19), only this time the first two moments in Eqs. (21)–(22), are computed using the M components that construct the sample pdf in (38):

$$\boldsymbol{\mu}_0 = \sum_{i=1}^{M} w^i \boldsymbol{\mu}^i \tag{39}$$

$$\mathbf{P}_0 = \sum_{i=1}^{M} w^i [\beta\mathbf{D}^* + (\boldsymbol{\mu}^i - \boldsymbol{\mu}_0)(\boldsymbol{\mu}^i - \boldsymbol{\mu}_0)^T]. \tag{40}$$

If $\alpha < 1$ or the maximum number of time steps has been reached, then the algorithm is stopped and the new initial Gaussian mixture is obtained as follows:

$$\begin{aligned} p_{\text{NEW}}(t_0, \mathbf{x}_0) &= p(t_0, \mathbf{x}_0) + \sum_{j=1}^{M} 0 \times \mathcal{N}(\mathbf{x}_0; \boldsymbol{\mu}^j, \beta\mathbf{D}^*) \\ &= \sum_{i=1}^{N} w_0^i \mathcal{N}(\mathbf{x}_0; \boldsymbol{\mu}_0^j, \mathbf{P}_0^i) + \sum_{j=1}^{M} 0 \times \mathcal{N}(\mathbf{x}_0; \boldsymbol{\mu}^j, \beta\mathbf{D}^*). \end{aligned} \tag{41}$$

The entire algorithm to select the Gaussian components is presented in Table I and graphical illustrations are presented in Fig. 2. In the case of multiple loss functions, the algorithm is run once for each one of the loss functions, creating sets of initial Gaussian components sensitive to their loss function.

While not the scope of this paper, the above method can also be applied when measurements are available between the current time and the decision time. The PSGC algorithm will be applied every time a measurement has been assimilated and the a posteriori pdf has been found. The drawback of this procedure is that the number of Gaussian components will increase linearly with the number of measurements. Instead of adding new Gaussian components, a better way to deal with this situation is to allocate from the total of N Gaussian components, M which are designated to be sensitive to the loss function and the rest $M - N$ to capture the dominant evolution of the pdf.

The Decision-Centric Density Forecasting is obtained by running first the PSGC algorithm, derived in this section, to supplement the initial uncertainty with M new Gaussian components sensitive to the loss function. The new initial Gaussian mixture obtained, (41), is then propagated using the Adaptive Gaussian Sum algorithm presented in Section 3. The following section presents the application of the Decision-Centric Density Forecasting method to a toxic cloud transported by wind and a low dimensional numerical example where a number of performance measures are computed.

ALGORITHM 1 *Progressive Selection of Gaussian Components*

Require: t_d—decision time
$p(t_0, \mathbf{x}_0) = \sum_{i=1}^{N} w_0^i \mathcal{N}(\mathbf{x}_0; \boldsymbol{\mu}_0^j, \mathbf{P}_0^i)$—initial probability density function
M—number of extra Gaussian components
$\mathbf{D}$—default Gaussian component covariance
w_{tol}—add only Gaussian components with weights greater than this threshold
$L(\mathbf{x}_d) = \mathcal{N}\{\mathbf{x}_d; \boldsymbol{\mu}_L, \boldsymbol{\Sigma}_L\}$—loss function
maxiter—maximum number of iterations

$p_{\text{Smp}}(t_0, \mathbf{x}_0) = p(t_0, \mathbf{x}_0)$, $\alpha = \infty$, $\gamma = -1$
while ($\alpha > 1$) & maxiter **do**
  The mean and the covariance of the sample pdf, if first iteration (21)–(22), otherwise (39)–(40)
  $\boldsymbol{\mu}_0 = E[\mathbf{x}_0] = \int \mathbf{x}_0 p_{\text{Smp}}(t_0, \mathbf{x}_0) d\mathbf{x}_0$
  $\mathbf{P}_0 = E[(\mathbf{x}_0 - \boldsymbol{\mu}_0)(\mathbf{x}_0 - \boldsymbol{\mu}_0)^T]$
    $= \int (\mathbf{x}_0 - \boldsymbol{\mu}_0)(\mathbf{x}_0 - \boldsymbol{\mu}_0)^T p_{\text{Smp}}(t_0, \mathbf{x}_0) d\mathbf{x}_0$
  while ($\gamma < 0$) **do**
    Get the means of the Gaussian components
    Draw $\boldsymbol{\mu}^i \sim p_{\text{Smp}}(t_0, \mathbf{x}_0)$ for $i = 1 \ldots M-1$
    Set $\boldsymbol{\mu}^M = M\boldsymbol{\mu}_0 - \sum_{i=1}^{M-1} \boldsymbol{\mu}^i$
    $\gamma = \dfrac{1}{\text{Tr}[\mathbf{D}]} \text{Tr}\left[\mathbf{P}_0 - \dfrac{1}{M} \sum_{i=1}^{M} (\boldsymbol{\mu}^i - \boldsymbol{\mu}_0)(\boldsymbol{\mu}^i - \boldsymbol{\mu}_0)^T\right]$
  end while
  Get the covariance of the Gaussian components
  $\mathbf{P}_0^i = \mathbf{D}^* = \gamma\mathbf{D}$
  Propagate the moments from $t = t_0$ to $t = t_d$
  $\dot{\boldsymbol{\mu}}_t^i = \mathbf{f}(t, \boldsymbol{\mu}_t^i)$
  $\dot{\mathbf{P}}_t^i = \mathbf{A}_t^i \mathbf{P}_t^i + \mathbf{P}_t^i (\mathbf{A}_t^i)^T + \mathbf{Q}$
  Get the most distant component by computing the Mahanalobis distance
  $d_i = (\boldsymbol{\mu}_L - \boldsymbol{\mu}_d^i)^T (\mathbf{P}_d^i + \boldsymbol{\Sigma}_L)^{-1} (\boldsymbol{\mu}_L - \boldsymbol{\mu}_d^i)$
  $\boldsymbol{\mu}_d^{\max}, \mathbf{P}_d^{\max} = \arg\max(d_i)$
  Compute optimal α and the inflated matrix Σ_L^*
  $\alpha = \dfrac{1}{n} \text{Tr}[((\boldsymbol{\mu}_d^{\max} - \boldsymbol{\mu}_L)(\boldsymbol{\mu}_d^{\max} - \boldsymbol{\mu}_L)^T - \mathbf{P}_d^{\max}) \Sigma_L^{-1}]$
  if $\alpha < 1$ **then** $\alpha = 1$ **end if**
  $\Sigma_L^* = \alpha \Sigma_L$
  Elements of $\mathbf{M} \in \mathbb{R}^{M\times M}$ and $\mathbf{N} \in \mathbb{R}^{M\times 1}$
  $m_{ij} = \mathcal{N}\{\boldsymbol{\mu}_d^j; \boldsymbol{\mu}_d^i, \mathbf{P}_d^i + \mathbf{P}_d^j\}$
  $n_i = \mathcal{N}\{\boldsymbol{\mu}_L; \boldsymbol{\mu}_d^i, \mathbf{P}_d^i + \boldsymbol{\Sigma}_L^*\}$
  Compute the weights
  $\mathbf{w} = \arg\min_{\mathbf{w}} \frac{1}{2} \mathbf{w}^T \mathbf{M} \mathbf{w} - \mathbf{w}^T \mathbf{N}$
    subject to $\mathbf{1}_{M\times 1}^T \mathbf{w} = 1$ and $\mathbf{w} \geq \mathbf{0}_{M\times 1}$

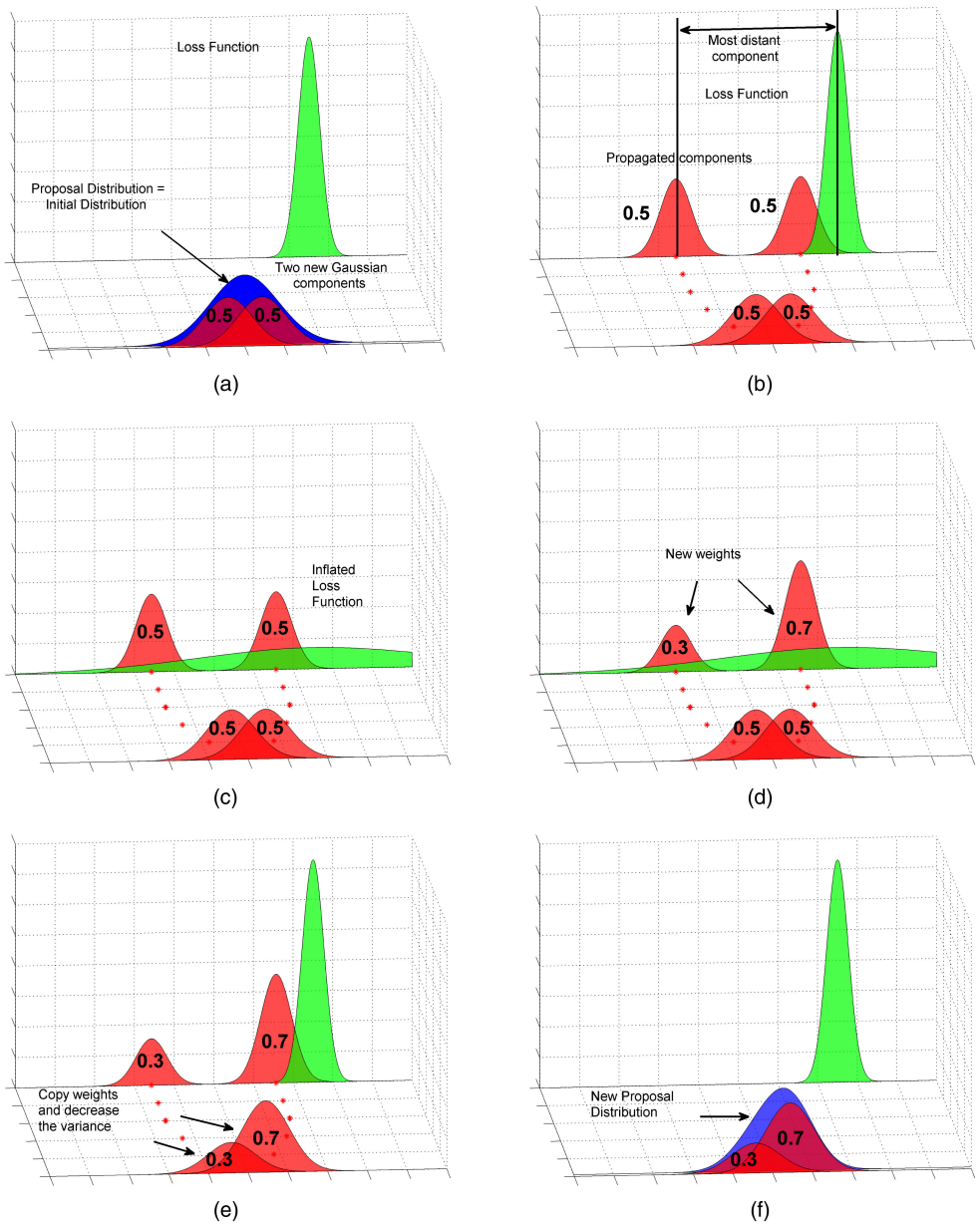

Fig. 2. Illustration for the PSGC algorithm. (a) Steps 1–8. (b) Steps 9–19. (c) Steps 11–12. (d) Steps 13–14. (e) Steps 15–16. (f) Goto Step 3 if $\sigma > 1$ and maximum number of iterations not reached.

15: **if** α is getting smaller **then** choose $\beta < 1$
else $\beta = 1$ **end if**
16: Set $p_{\text{Smp}}(t_0, \mathbf{x}_0) = \sum_{j=1}^{M} w^j \mathcal{N}\{\mathbf{x}_0; \boldsymbol{\mu}^j, \beta \mathbf{D}^*\}$
17: **end while**
18: Set $p_{\text{NEW}}(t_0, \mathbf{x}_0) = p(t_0, \mathbf{x}_0) + \sum_{j=1}^{M, w^j \geq w_{\text{tol}}} 0$
$\times \mathcal{N}\{\mathbf{x}_0; \boldsymbol{\mu}^j, \beta \mathbf{D}^*\}$
19: **return** $p_{\text{NEW}}(t_0, \mathbf{x}_0)$

5. NUMERICAL RESULTS

Chemical, Biological, Radiological, and Nuclear (CBRN) incidents are rare events but very consequential, which mandates extensive research and operational efforts in mitigating their outcomes. Many puff dispersion models, such as SCIPUFF [33] and RIMPUFF [22], try to model the atmospheric transport and diffusion of toxic plumes. Similarly, BIGFLOW [1] can be used to analyze the contaminant transport problem in the nonlinear porous media. While inherently stochastic and highly nonlinear, these mathematical models are able to capture just a part of the dynamics of the real phenomenon and the forward integration yields an uncertain prediction. The decision maker takes actions based on the expected loss computed using both the predicted uncertainty and the loss function, which here maps a region of interest in the state space into a threat level, such as the population density in a town. Thus the ability to propagate the uncertainty and errors throughout the dynamic system is of great importance. As mentioned previously, the present method can be also ap-

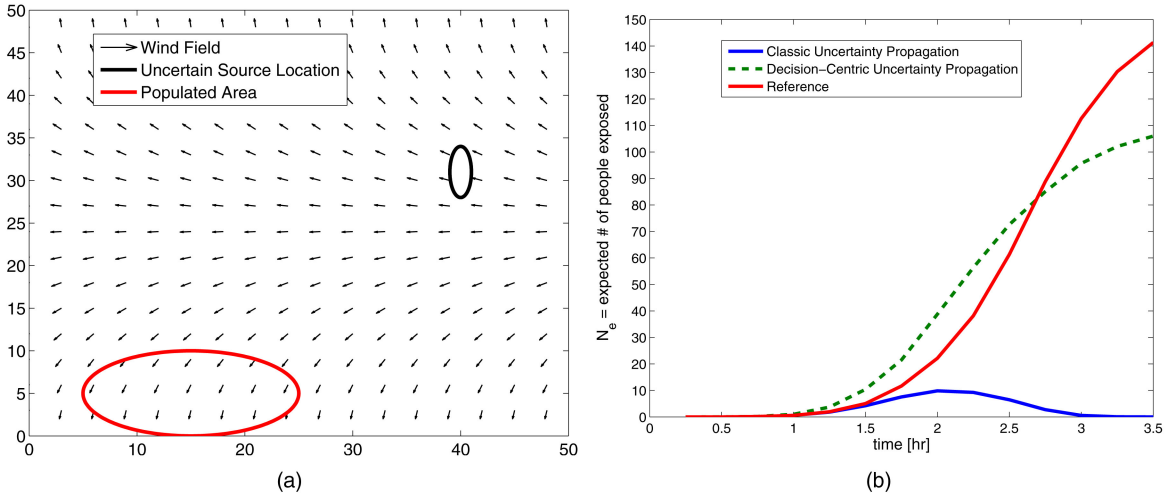

Fig. 3. (a) Chemical release scenario. (b) Evolution of number of people exposed.

plied for nonlinear filtering and smoothing problem which are relevant for source determination (localization and characterization) in the case of covert releases [3, 6, 34].

5.1. Example 1

To motivate the concept of incorporating contextual information into the uncertainty propagation algorithm, we consider the following noise driven nonlinear dynamic system that simulates the advection and the dispersion of a chemical material released from an uncertain location. A similar system has been previously used in [26, 34].

The instantaneous amount of material released is represented using a Gaussian-shaped puff, states of which evolve using the following equations, which describe a wind pattern as shown in Fig. 3(a).

$$\begin{aligned} \dot{x}(t) &= -a\sin(by(t)) + w_1(t) \\ \dot{y}(t) &= -a\cos(by(t)) + w_2(t) \\ \dot{s}(t) &= a \end{aligned} \tag{42}$$

where (x,y) is the position of the center of the puff, and the downwind distance from the source $s(t_k) = s_k$ is used to compute the puff radius at time $t = t_k$,

$$\sigma_k = p_y s_k^{q_y}. \tag{43}$$

Due to the simplicity of the model and the lack of knowledge about the initial conditions, release location in (44), the model forecast tends to become less accurate for longer simulations.

$$\begin{aligned} &p(t_0, x(t_0), y(t_0)) \\ &\quad = \mathcal{N}([x,y]^T; [40,31]^T, \mathrm{diag}\{[1,9]^2\}) \\ &s(t_0) = 0. \end{aligned} \tag{44}$$

The puff radius depends on meteorological conditions specified by the Karlsruhe-Jülich diffusion coefficients [22] which are set to $p_y = 0.466$, $q_y = 0.866$. The wind speed is considered to be $a = 10$ mph and the variable b depends on the boundaries of the domain and is set here to $\pi/50$. The process noise, $[w^1(t), w^2(t)]^T$, is a vector whose components are independent Gaussian white noise processes induced in the process model due to the uncertainty in the wind field. The autocorrelation function of the process noise is given by $\mathbf{Q} = 2\mathbf{I}_{2\times 2}\delta(t-\tau)$.

The concentration at each grid point, at time $t = t_k$, is computed using the following relation,

$$C_k(x_g, y_g) = \frac{M}{2\pi\sigma_k^2} \exp\left(-\frac{(x_k - x_g)^2 + (y_k - y_g)^2}{2\sigma_k^2} \right) \tag{45}$$

where $x_k = x(t_k)$, $y_k = y(t_k)$, and the instantaneous mass released is $M = 10$ kg.

These equations capture the main characteristics of puff-based dispersion models for a particular wind field. The weights of different Gaussian components have been updated every $\Delta t = 0.25$ hr using the error in the FPKE. The impulse chemical release is done in a region of 50×50 sqmi and the total simulation time is 3.5 hrs. The source location and its uncertainty as well as the decision region of interest, here the populated area with 10,000 residents, represented with a Gaussian function, $D(x,y)$ in (46), are shown in Fig. 3(a).

$$D(x,y) = 10{,}000 \times \mathcal{N}([x,y]^T; [15,5]^T, \mathrm{diag}\{[10,5]^2\}). \tag{46}$$

First, we propagate the uncertainty using the first-order Taylor expansion. We will call this method Classic Uncertainty Propagation. To evaluate the effect of the uncertainty propagation in the process of decision making, we compute the probability of the chemical concentration exceeding a critical value, $c_t = 0.0001$, which is assumed to be harmful. The decision maker may decide to evacuate or not the populated area, based on the evolution of this probability or hazard map and the number of people placed at risk due to exposure above

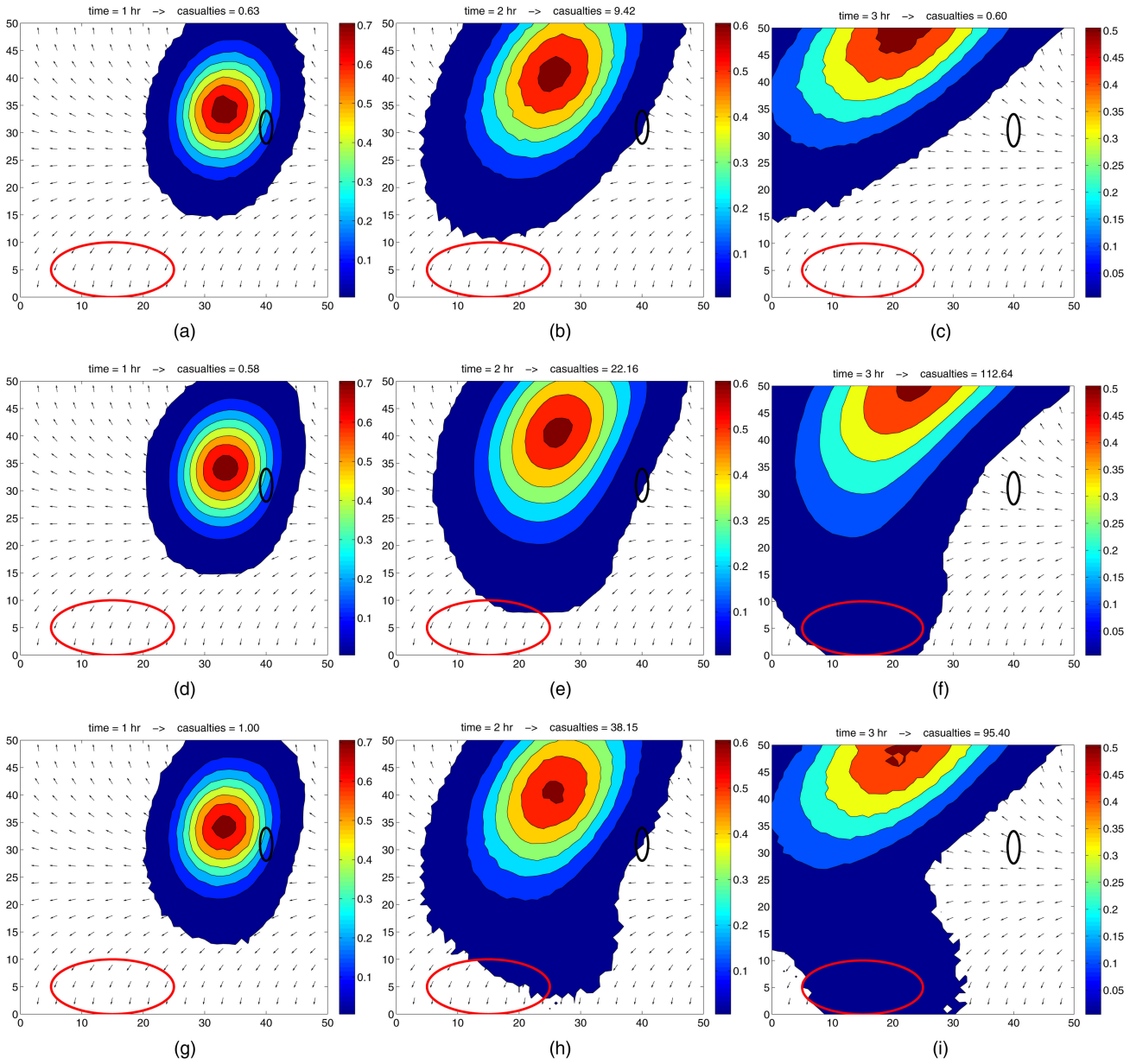

Fig. 4. Hazard maps: Probability evolution of chemical concentrations exceeding the critical value. (a) Classic uncertainty propagation after 1 hr. (b) Classic uncertainty propagation after 2 hrs. (c) Classic uncertainty propagation after 3 hrs. (d) Reference after 1 hr (Monte Carlo simulation). (e) Reference after 2 hrs (Monte Carlo simulation). (f) Reference after 3 hrs (Monte Carlo simulation). (g) Decision-centric uncertainty propagation after 1 hr. (h) Decision-centric uncertainty propagation after 2 hrs. (i) Decision-centric uncertainty propagation after 3 hrs.

safe concentration thresholds. Fig. 4(a), (b), (c) shows the probability evolution of the chemical concentration exceeding the consequential value for the Classic Uncertainty Propagation.

As reference, we use a Monte Carlo simulation, using 5,000 samples to evaluate as close as possible the probability of consequential concentrations. The evolution of the probability is presented in Fig. 4(d), (e), (f). We observe that in reality, consequential chemical concentrations are well into the populated area after 2 hrs. We refer to this method as Reference.

We apply the method presented in this paper to generate at most 5 new Gaussian components to be added to the initial condition. Their means and variances are returned by the PSGC algorithm, Algorithm 1. The initial weights of the new Gaussian components have been set to zero. The default value for the β coefficient is 0.9 and Gaussian components are included only if their weights are greater than $w_{\text{tol}} = 10^{-3}$. The label used for this method is Decision-Centric Uncertainty Propagation and its corresponding hazard map is presented in Fig. 4(g), (h), (i).

By accounting for the populated region, we are able to track the probability that consequential chemical concentrations are reaching that region. The expected number of people exposed to critical concentrations at dif-

ferent times are presented in Fig. 3(b) and is computed using (47). The classic uncertainty propagation is underestimating the magnitude of the situation, misguiding the decision maker to make a consequential decision such as not evacuating the region.

$$N_e = \int D(x,y)P(C_k(x,y) \geq c_t)dx\,dy. \tag{47}$$

Our method, while still using the same principles of first-order Taylor expansion, is able to capture the probability density mass in the region of interest by adding Gaussian components that are sensitive to this area and to estimate the expected number of people exposed in the same order of magnitude as the Reference.

While this example presents a particular wind pattern, the method can also be applied in more realistic scenarios using the Lagrangian puff atmospheric dispersion model RIMPUFF [22] in connection with any wind forecasting module such as WRF or MM5 [11].

The next example evaluates the performance of the decision centric forecasting method against a number of performance measures.

5.2. Example 2

To better illustrate the steps of the proposed method, as well as to evaluate its performance against a number of performance measures, we also consider the following low dimensional continuous-time dynamic system with uncertain initial condition given by:

$$\begin{aligned} &\dot{x} = \sin(x) + \Gamma(t) \qquad \text{where} \quad Q = 1 \\ &p(t_0,x_0) = \mathcal{N}(x_0; -0.3, 0.3^2). \end{aligned} \tag{48}$$

The noise free dynamic system in (48) is of particular interest since it exhibits chaotic behavior caused by multiple equilibrium states. The state space region of interest is depicted by the following loss function, and the time of decision is at $t_d = 8$ sec.

$$L(x_d) = \mathcal{N}\left(x_d; \frac{\pi}{2}, 0.1^2\right). \tag{49}$$

First we compute an accurate numerical solution based on the discretization of the FPKE, and this will stand as the reference probability density function. The performance measures for this method will be labeled as REF. The evolution of the pdf using this method can be seen in Fig. 5(a).

Three other approximations for the pdf are provided including the method presented in this paper. The first approximation propagates the initial uncertainty using the Extended Kalman Filter time update equations, Eqs. (10)–(11), labeled later as EKF. The evolution of the pdf for this method is presented in Fig. 5(b).

For the next approximation method, we add 5 Gaussian components to the initial uncertainty, creating a Gaussian mixture with 6 components. The means of the new components are just the result of back propagation (from $t_d = 8$ sec to $t_0 = 0$ sec) of 5 equidistant samples taken in the 3 sigma bound of the loss function support. The variance of the new components is set to 10^{-10} and their initial weights are set to zero. The label used for this method is GS_BCK and the evolution of the pdf is shown in Fig. 5(c). While all the means of the new Gaussian components are positioned in the loss function support region, their variances get large and the probability density mass in that region is difficult to be visualized.

TABLE 1
Performance Measures—500 Monte Carlo Runs

	$\hat{L}_d$	$\hat{R}_{\text{err}}$	ISD	WISD
REF	0.0332	N/A	N/A	N/A
EKF	4.93E-09	1.0000	0.1840	0.0015
GS_BCK	0.0001	0.9968	0.0536	0.0015
GS_DEC (mean)	0.0256	0.2300	0.0470	0.0004
GS_DEC: Percentile Table—500 Observations				
Percent	$\hat{L}_d$	$\hat{R}_{\text{err}}$	ISD	WISD
0.0%	0.0010	0.0151	0.0368	0.0002
5.0%	0.0142	0.0230	0.0378	0.0003
10.0%	0.0177	0.0271	0.0380	0.0003
25.0%	0.0229	0.0566	0.0387	0.0003
50.0%	0.0257	0.2270	0.0491	0.0003
75.0%	0.0313	0.3090	0.0514	0.0004
90.0%	0.0323	0.4670	0.0574	0.0006
95.0%	0.0324	0.5710	0.0601	0.0007
100.0%	0.0327	0.9700	0.0705	0.0014

We apply the method presented in this paper to generate at most 5 new Gaussian components to be added to the initial condition. Their means and variances are returned by the progressive selection algorithm, Algorithm 1. The initial weights of the new Gaussian components have been set to zero. The default value for the β coefficient is 0.9 and Gaussian components are included only if their weights are greater than $w_{\text{tol}} = 10^{-3}$. The label used for this method is GS_DEC and its corresponding pdf is presented in Fig. 5(d).

The evolution the Gaussian components for the last two methods is also achieved using the Extended Kalman Filter time update equations, but it is interrupted every $\Delta t = 0.5$ sec to adjust the weights of different Gaussian components using the optimization in (14).

The following performance measures have been computed for the methods used in the experiment:

$$\hat{L}_d = \int L(x)\hat{p}(t_d,x_d)dx_d \tag{50}$$

$$\hat{R}_{\text{err}} = \frac{1}{L_d}|L_d - \hat{L}_d| \tag{51}$$

$$\text{ISD} = \int |p(t_d,x_d) - \hat{p}(t_d,x_d)|^2 dx_d \tag{52}$$

$$\text{WISD} = \int L(x)|p(t_d,x_d) - \hat{p}(t_d,x_d)|^2 dx_d. \tag{53}$$

In Fig. 5(e) the forecast pdf is plotted at time t_d for all the methods, for a particular Monte Carlo run. Our method, GS_DEC, is able to better estimate the probability density mass in the region of interest.

In Table I, we present the performance measures after 500 Monte Carlo runs. The expected loss given

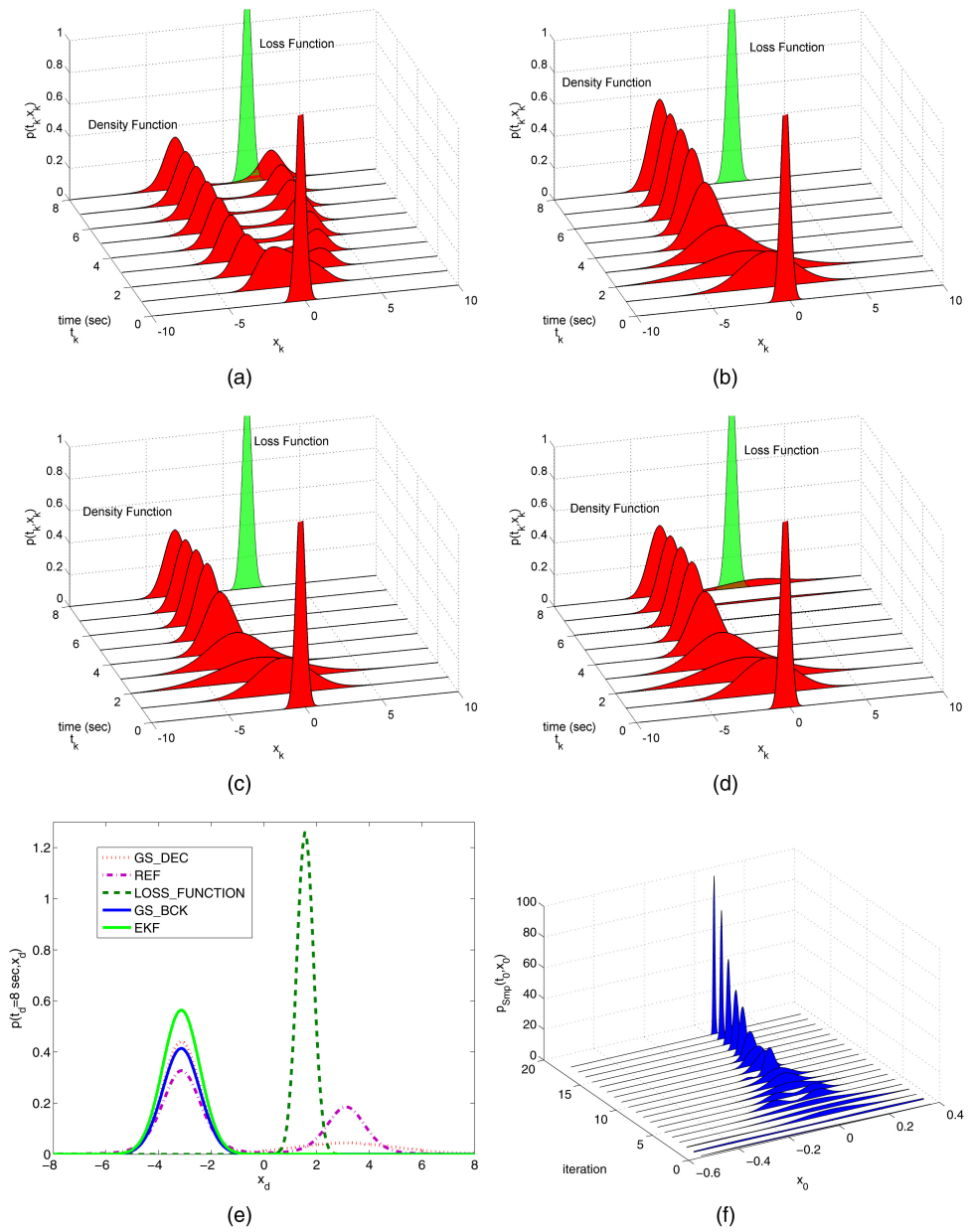

Fig. 5. The evolution of the forecast pdf and the sampling pdf. (a) REF: Numerical approximation FPKE. (b) EKF: first order Taylor expansion approximation. (c) GS_BCK: back propagated means. (d) GS_DEC: progressive selection of Gaussian components. (e) Probability density function at f_d = 8 sec. (f) The evolution of the pdf, $ps_{mp}(t_0, \mathbf{x}_0)$, used to sample the means of the Gaussian components.

by the GS_DEC method is consistently better over all the Monte Carlo runs than the EKF and the GS_BCK method. We are also able to consistently give an overall better approximation to the pdf and in the region of interest than the EKF method, which justifies the use of this method. Compared with the GS_BCK we do a better job on average in approximating the pdf which suggests that there is a trade off in selecting the Gaussian components regarding their means and variances.

In Fig. 5(f) it is plotted the evolution of the pdf, $p_{\text{Smp}}(t_0, \mathbf{x}_0)$, used to sample the means of the new Gaussian components for a particular Monte Carlo run. The pdf used in the first iteration is our initial uncertainty and we see how it converges, as the number of iterations increases, to a particular region in the state space that is sensitive to the loss function at the decision time.

6. CONCLUSIONS AND FUTURE WORK

A decision-centric view to create an interaction level between the decision maker and the density forecasting module has been designed, such that we can incorporate contextual information held by the decision maker into the uncertainty propagation process to better approximate the probability density function and the expected loss value.

The Progressive Selection of Gaussian Components algorithm is run once at the beginning of the simulation

to supplement the initial uncertainty with new Gaussian components that are sensitive to the loss function at the decision time. The weights of all the Gaussian components are then updated during the propagation based on the error in the Fokker-Planck-Kolmogorov Equation. This way we obtain not only a better approximation of the probability density function in the region of interest but also a better approximation overall. The cost of this overall improvement is an increase in the number of Gaussian components. The principal benefit is not the modest increase in accuracy overall, but the significantly enhanced accuracy within the decision maker's region of interest.

Although the novel method in this paper is presented only in the pure forecast context, it is equally relevant in solving nonlinear filtering problems when measurements are available. The implementation of the method in stochastic filtering context is briefly discussed in the paper, and its performance evaluation on numerical examples is set as future work.

APPENDIX

LEMMA 1 *If* $\mathrm{Tr}[\mathbf{ABA}] = 0$ *and* $\mathbf{A}$ *is symmetric and positive definite then* $\mathrm{Tr}[\mathbf{B}] = 0$.

PROOF Let $\mathbf{A} = \mathbf{VSV}^T$ be a singular value decomposition of matrix $\mathbf{A}$, where $\mathbf{V}$ is a unitary matrix and $\mathbf{S}$ is a diagonal matrix. Our trace can now be written as $\mathrm{Tr}[\mathbf{ABA}] = \mathrm{Tr}[\mathbf{VSV}^T\mathbf{BVSV}^T] = \mathrm{Tr}[\mathbf{S}^2\mathbf{B}]$.

If $\mathrm{Tr}[\mathbf{S}^2\mathbf{B}] = 0$ then $\mathbf{S}^2\mathbf{B}$ is a commutator. Thus there is $\mathbf{X}$ and $\mathbf{Y}$ such that $\mathbf{S}^2\mathbf{B} = \mathbf{XY} - \mathbf{YX}$. But $\mathbf{B} = \mathbf{S}^{-2}\mathbf{XY} - \mathbf{S}^{-2}\mathbf{YX} = \mathbf{X}^*\mathbf{Y} - \mathbf{YX}^*$, where $\mathbf{X}^* = \mathbf{S}^{-2}\mathbf{X}$. Therefore $\mathbf{B}$ is also a commutator, hence $\mathrm{Tr}[\mathbf{B}] = 0$.

ACKNOWLEDGMENT

This work was supported under Contract No. HM1582-08-1-0012 from ONR.

REFERENCES

[1] R. Ababou and A. C. Bagtzoglou
Bigflow: A numerical code for simulating flow in variably saturated, heterogeneous geologic media. Theory and user's manual, version 1.1.
Technical report, Nuclear Regulatory Commission, Washington, D.C. (United States). Div. of Regulatory Applications, NUREG/CR-6028, 1993.

[2] D. Alspach and H. Sorenson
Nonlinear Bayesian estimation using Gaussian sum approximations.
IEEE Transactions on Automatic Control, **17**, 4 (1972), 439–448.

[3] A. C. Bagtzoglou and S. A. Baun
Near real-time atmospheric contamination source identification by an optimization-based inverse method.
Inverse Problems in Science and Engineering, **13** (2005), 241–259.

[4] R. N. Banavar and J. L. Speyer
Properties of risk-sensitive filters/estimators.
IEE Proceedings of Control Theory and Applications, **145** (1998), 106–112.

[5] S. Chakravorty, M. Kumar, and P. Singla
A quasi-Gaussian Kalman filter.
American Control Conference, 2006, June 2006, 6 pp.

[6] Y. Cheng and T. Singh
Source term estimation using convex optimization.
In *The 11th International Conference on Information Fusion*, Cologne, Germany, July 1–3, 2008.

[7] F. Daum and J. Huang
Curse of dimensionality and particle filters.
In *Proceedings of IEEE Aerospace Conference*, vol. 4, 2003, 8–15.

[8] A. Doucet, N. de Freitas, and N. Gordon
Sequential Monte-Carlo methods in practice.
Springer-Verlag, Apr. 2001.

[9] A. T. Fuller
Analysis of nonlinear stochastic systems by means of the Fokker-Planck equation.
International Journal of Control, **9**, 6 (1969).

[10] D. Giza and P. Singla
An approach for nonlinear uncertainty propagation: Application to orbital mechanics.
In *AIAA Guidance Navigation and Control Conference*, 2009.

[11] G. A. Grell, J. Dudhia, and D. R. Stauffer
A description of the fifth generation Penn State/NCAR Mesoscale Model (MM5).
Technical report, NCAR/TN-398+STR, NCAR Technical Note, Mesoscale and Microscale Meteorology Division, National Center for Atmospheric Research, Boulder, CO, 1995.

[12] R. N. Iyengar and P. K. Dash
Study of the random vibration of nonlinear systems by the Gaussian closure technique.
Journal of Applied Mechanics, **45** (1978), 393–399.

[13] S. J. Julier and J. K. Uhlmann
Unscented filtering and nonlinear estimation.
Proceedings of the IEEE, **92**, 3 (2004), 401–422.

[14] R. W. Katz and A. H. Murphy, (eds.)
Economic value of weather and climate forecasts.
Cambridge University Press, 1997.

[15] M. Kumar, P. Singla, S. Chakravorty, and J. L. Junkins
A multi-resolution approach for steady state uncertainty determination in nonlinear dynamical systems.
In *38th Southeastern Symposium on System Theory*, 2006.

[16] M. Kumar, P. Singla, S. Chakravorty, and J. L. Junkins
The partition of unity finite element approach to the stationary Fokker-Planck equation.
In *2006 AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Keystone, CO, Aug. 21–24, 2006.

[17] H. C. Lambert, F. E. Daum, and J. L. Weatherwax
A split-step solution of the Fokker-Planck equation for the conditional density.
Fortieth Asilomar Conference on Signals, Systems and Computers, 2006., Oct.–Nov. 2006, 2014–2018.

[18] T. Lefebvre, H. Bruyninckx, and J. De Schutter
Comment on a new method for the nonlinear transformations of means and covariances in filters and estimators.
IEEE Transactions on Automatic Control, **47**, 8 (2002).

[19] T. Lefebvre, H. Bruyninckx, and J. De Schutter
Kalman filters of non-linear systems: A comparison of performance.
International Journal of Control, **77**, 7 (2004), 639–653.

[20] G. Muscolino, G. Ricciardi, and M. Vasta
Stationary and non-stationary probability density function for non-linear oscillators.
International Journal of Non-Linear Mechanics, **32** (1997), 1051–1064.

[21] C. Musso, N. Oudjane, and F. Legland
Improving regularized particle filters.
In A. Doucet, N. de Freitas, and N. Gordon, (Eds.), *Sequential Monte Carlo Methods in Practice*, Statistics for Engineering and Information Science, New York, 12 (2001), 247–271.

[22] S. T. Nielsen, S. Deme, and T. Mikkelsen
Description of the atmospheric dispersion module RIMPUFF.
Technical Report RODOS(WG2)-TN(98)-02, Riso National Laboratory, P.O. Box 49, DK-4000 Roskilde, Denmark, 1999.

[23] M. D. Paola and A. Sofi
Approximate solution of the Fokker-Planck-Kolmogorov equation.
Probabilistic Engineering Mechanics, **17** (2002), 369–384.

[24] G. Parmigiani and L. Inoue
Decision Theory: Principles and Approaches.
John Wiley & Sons, 2009.

[25] M. H. Pesaran and S. Skouras
Decision-based methods for forecast evaluation.
In M. P. Clements and D. F. Hendry, (Eds.), *A Companion to Economic Forecasting*, Blackwell Publishers, 11 (2002), 241–267.

[26] K. V. U. Reddy, Y. Cheng, T. Singh, and P. D. Scott
Data assimilation in variable dimension dispersion models using particle filters.
In *The 10th International Conference on Information Fusion*, Quebec City, Canada, 2007.

[27] H. Risken
The Fokker-Planck Equation: Methods of Solution and Applications.
Springer, 1989.

[28] J. B. Roberts and P. D. Spanos
Random Vibration and Statistical Linearization.
Wiley, 1990.

[29] S. Sadhu, S. Bhaumik, A. Doucet, and T. K. Ghoshal
Particle-method-based formulation of risk-sensitive filter.
Signal Processing, **89**, 3 (2009), 314–319.

[30] P. Singla and T. Singh
A Gaussian function network for uncertainty propagation through nonlinear dynamical system.
18th AAS/AIAA Spaceflight Mechanics Meeting, Galveston, TX, Jan. 27–31, 2008.

[31] I. H. Sloan and H. Woniakowski
When are quasi-Monte Carlo algorithms efficient for high dimensional integrals?
Journal of Complexity, 1998, 1–33.

[32] A. N. Steinberg and C. L. Bowman
Rethinking the JDL data fusion levels.
In *Proceedings of the National Symposium on Sensor and Data Fusion*, John Hopkins Applied Physics Laboratory, 2004.

[33] I. R. Sykes, S. F. Parker, D. S. Henn, and B. Chowdhury
SCIPUFF Version 2.2.
Technical report, L-3 Titan Corporation, NJ, Jan. 2006.

[34] G. Terejanu, T. Singh, and P. D. Scott
Unscented Kalman filter/smoother for a CBRN puff-based dispersion model.
In *The 10th International Conference on Information Fusion*, Quebec City, Canada, 2007.

[35] G. Terejanu, P. Singla, T. Singh, and P. D. Scott
A novel Gaussian sum filter method for accurate solution to nonlinear filtering problem.
In *The 11th International Conference on Information Fusion*, Cologne, Germany, 2008.

[36] G. Terejanu, P. Singla, T. Singh, and P. D. Scott
Uncertainty propagation for nonlinear dynamical systems using Gaussian mixture models.
Journal of Guidance, Control, and Dynamics, **31** (2008), 1623–1633.

[37] G. Terejanu, P. Singla, T. Singh, and P. D. Scott
Decision based uncertainty propagation using adaptive Gaussian mixtures.
In *The 12th International Conference on Information Fusion*, Seattle, WA, July 2009.

[38] G. Terejanu, P. Singla, T. Singh, and P. D. Scott
Adaptive Gaussian sum filter for nonlinear Bayesian estimation.
Under review, submitted to *IEEE Transactions on Automatic Control*, Sept. 2009.

[39] S. Thrun, J. Langford, and V. Verma
Risk sensitive particle filters.
In *Advances in Neural Information Processing Systems 14*, MIT Press, 2002.

[40] J. von Neumann and O. Morgenstern
Theory of Games and Economic Behavior.
Princeton University Press, 1944.

Gabriel Terejanu received his B.E. degree in automation, specialization robots, from University of Craiova, Romania in 2004 and the M.S. and Ph.D. degrees in computer science and engineering from University at Buffalo, in 2007 and 2010, respectively.

He is currently a postdoctoral fellow in the Center of Predictive Engineering and Computational Sciences (PECOS) at the Institute for Computational Engineering and Sciences (ICES), University of Texas at Austin. His research interests are in information fusion, model validation, uncertainty quantification, and decision making under epistemic and aleatory uncertainty.

Puneet Singla received the B.S. degree from the Indian Institute of Technology, Kanpur, India, in 2000 and the M.S. and Ph.D. degrees from Texas A&M University, in 2002 and 2006, respectively, all in aerospace engineering.

He is currently an assistant professor of mechanical and aerospace engineering with the University at Buffalo. His work in attitude estimation included algorithms supporting a successful experiment StarNav that won the STS-107. He has authored more than 60 papers to date, including ten journal papers covering a wide array of problems, including attitude estimation, dynamics and control, approximation theory, and uncertainty propagation. He is the principal author of a new textbook *Multi-Resolution Methods for Modeling and Control of Dynamical Systems* (Boca Raton, FL: CRC Press, 2008).

Tarunraj Singh received his B.E., M.E., and Ph.D. degrees in mechanical engineering from Bangalore University, Indian Institute of Science, and the University of Waterloo, respectively.

He was a postdoctoral fellow in the Aerospace Engineering Department of Texas A&M University prior to starting his tenure at the University at Buffalo in 1993, where he is currently a professor in the Department of Mechanical and Aerospace Engineering. He was a von Humboldt Fellow and spent his sabbatical at the Technische Universitat Darmstadt in Germany and at the IBM Almaden Research center in 2000–2001. He was a NASA Summer Faculty Fellow at the Goddard Space Flight Center in 2003. His research is supported by the National Science Foundation, AFOSR, NSA, Office of Naval Research and various industries including MOOG Inc., Praxair and Delphi Thermal Systems. His research interests are in robust vibration control, optimal control, nonlinear estimation and intelligent transportation. Dr. Singh has published over 100 refereed journal and conference papers and has presented over 30 invited seminars at various universities and research laboratories.

Peter D. Scott received the B.S., M.S., and Ph.D. degrees from the Phillips School of Electrical Engineering, Cornell University.

He has served as assistant and associate professor in the University at Buffalo with primary appointment in the Electrical and Computer Engineering Department, subsequently in the Computer Science and Engineering Department. He has held adjunct faculty positions in the Department of Biophysical Sciences and currently with the School of Medicine and Department of Electrical Engineering in the University at Buffalo. He held a sabbatical faculty position with the Department of Electrical Engineering at the University of Connecticut and was Director of the Surgical Computing Laboratory, The Buffalo General Hospital. He has published over a hundred books, articles and peer reviewed conference publications. His research has been supported by NSF, NIH Heart and Lung Institute, New York Science and Technology Institute, the Wendt Foundation, and various agencies of the DOD. His research interests include optimal control and estimation, bioelectric phenomena, neural networks, data fusion and management of uncertainty.