LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-PROC-217190

Sequential Monte-Carlo Based Framework for Dynamic Data-Driven Event Reconstruction for Atmospheric Release

G. Johannesson, F.K. Chow, L. Glascoe, R.E. Glaser, W.G. Hanley, B. Kosovic, M. Krnjajic, S.C. Larsen, J.K. Lundquist, A.A. Mirin, J.J. Nitao, G.A. Sugiyama

November 18, 2005

2005 Joint Statistical Meetings
Minneapolis, MN, United States
August 7, 2005 through August 11, 2005

Sequential Monte-Carlo Based Framework for Dynamic Data-Driven Event Reconstruction for Atmospheric Release

Gardar Johannesson, Fotini K. Chow, Lee Glascoe, Ronald E. Glaser,
William G. Hanley, Branko Kosovic, Milovan Krnjajić, Shawn C. Larsen,
Julie K. Lundquist, Arthur A. Mirin, John J. Nitao, Gayle A. Sugiyama
Lawrence Livermore National Laboratory

Abstract

Atmospheric releases of hazardous materials are highly effective means to impact large populations. We propose an atmospheric event reconstruction framework that couples observed data and predictive computer-intensive dispersion models via Bayesian methodology. Due to the complexity of the model framework, a sampling-based approach is taken for posterior inference that combines Markov chain Monte Carlo (MCMC) and sequential Monte Carlo (SMC) strategies.

1. Introduction

Atmospheric event reconstruction infers characteristics of an unknown release of a chemical or biological agent into the atmosphere (i.e., how much? when? where?) and predicts its dispersion; Figure 1. This is accomplished by coupling together a forward-predictive atmospheric dispersion model and relatively sparse sensor data to "estimate" the unknown release input parameters of the dispersion model. The complexity of atmospheric dispersion and the wide range of possible release scenarios (stationary, moving, multiple, etc.) present difficult challenges. The complexity of the atmospheric dispersion process manifests itself in ill-quantified and varying errors associated with using a computer-based dispersion prediction model. This uncertainty is due to approximations made to the underlying physics of atmospheric dispersion, numeric errors associated with the code itself, and, not least, to uncertainty associated with the assumed-known input parameters related to topology, meteorology, and turbulence. In addition, accurate dispersion models are in general computationally demanding, while at the same time one seeks to conduct inference on the unknown release and its impact in a timely fashion and provide valuable feedback for release consequence management. Hence, we seek an inference method that is able to efficiently use a given dispersion model and propagate the uncertainty associated with the model and the sensor data to uncertainty about the characteristics of the release and its impact. To further complicate matters, the problem itself is dynamic: as new data is gathered concerning the release, we need to efficiently update our state-of-knowledge.

A team of scientists at Lawrence Livermore National Laboratory is implementing a dynamic Bayesian approach to conduct inference on the unknown release parameters. This approach allows us to couple together, in a probabilistic way, observed data and the dispersion model, along with prior knowledge about the characteristics of the source and the model-error. Initial posterior inference is carried out using Markov chain Monte Carlo (MCMC) via parallel chains, and therefore via parallel dispersion model runs, to yield posterior realizations. Posterior updates are conducted using a sequential Monte Carlo (SMC) based method, which utilizes past posterior realizations and is well suited for parallel runs of the dispersion model.

The next section follows with more details on the problem setup and notation. In Section 3., we give some details on the range of available dispersion models and conditions that can lead to computational improvements. The general outline of the dynamic Bayesian model is given in Section 4., with the following section describing the approach taken to generate posterior realizations via MCMC and SMC.

2. Problem Setup and Notation

Our main goal is to conduct dynamic (posterior) inference on an unknown atmospheric release and its impact as more information (data) accumulates. Let $t = 1, 2, \ldots$ index the time-points where we wish to update our state-of-knowledge and let:

$\tau_t \equiv$ the time-point of the t-th update and

$\mathcal{T}_t \equiv (\tau_{t-1}, \tau_t]$, the t-th time interval.

The time-points $\{\tau_t\}$ can either be fixed in advance or dynamically chosen based on the availability and coverage of the incoming data.

The unknown atmospheric release can be due to one or more simultaneous sources. Let:

$\boldsymbol{\theta}_{t,i} \equiv$ collection of parameters characterizing the i-th source in $\mathcal{T}_t$.

$\boldsymbol{\theta}_t \equiv \{\boldsymbol{\theta}_{t,i}\}$.

In particular, we focus on point sources, in which case, $\boldsymbol{\theta}_{t,i} = \{\mathbf{x}_{t,i}, r_{t,i}\}$, where:

$\mathbf{x}_{t,i} \equiv$ the location of the i-th source in $\mathcal{T}_t$.

$r_{t,i} \equiv$ the release rate of the i-th source in $\mathcal{T}_t$.

Hence, in the t-th time interval, the i-th point source is at location $\mathbf{x}_{t,i}$ with an average release rate of $r_{t,i}$. We let

$$\boldsymbol{\theta}_{1:t} \equiv \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_t\} \ ,$$

and similarly for other parameters indexed by t.

The impact of the release can be observed in various ways, for example by a network of sensor instruments. Given the data from a network of sensors, let:

$\mathbf{s}_j \equiv$ the location of the j-th sensor.

$c_{j,k} \equiv$ the k-th average concentration reported from the j-th sensor.

$\mathcal{T}_{j,k} \equiv (\tau^s_{j,k}, \tau^e_{j,k}) \equiv$ the time interval in which $c_{j,k}$ was measured over.

We note that the sensors do not need to be stationary nor does the data need to be associated with a point (i.e., it can be associated with spatial blocks, voxels). We define

$\mathbf{d}_t \equiv \{c_{j,k} : (j,k) \in \mathcal{I}_t\}$, where,

$\mathcal{I}_t \equiv \{(j,k) : \tau_{t-1} < \tau^e_{j,k} \leq \tau_t\}$, indexes the sensor data that is available for use at the t-th time point, but was not available at $t-1$.

Hence, $\mathbf{d}_{1:t} \equiv \{\mathbf{d}_1, \ldots, \mathbf{d}_t\}$ is all the data available from time periods 1 to t.

An atmospheric dispersion model (a computer simulation code) is used to relate the source parameters to the sensor data. The dispersion model yields concentration predictions given various input parameters, including the source parameters. In addition to source parameters, other major model input parameters relate to the underlying meteorology, topography, and turbulence. We will fix some of the needed input parameters, but treat others (in addition to the source parameters) as unknown and include them in our model. Let

$\boldsymbol{\phi} \equiv$ vector of additional dispersion model input parameters treated as unknown.

Given all the needed input parameters, let:

$\hat{C}(\mathbf{s}, \tau) \equiv \hat{C}(\mathbf{s}, \tau; \boldsymbol{\theta}_{1:t}, \boldsymbol{\phi}) \equiv$ the predicted concentration at location $\mathbf{s}$ at time $\tau \in (\tau_0, \tau_t]$, given a particular source history, $\boldsymbol{\theta}_{1:t}$, and other input parameters, $\boldsymbol{\phi}$.

$\hat{C}(B, \mathcal{T}) \equiv \hat{C}(B, \mathcal{T}; \boldsymbol{\theta}_{1:t}, \boldsymbol{\phi}) \equiv$ the predicted average concentration in the spatial block (voxel) B over the time period $\mathcal{T}$, $\mathcal{T} \subset (\tau_0, \tau_t]$.

To be in line with the observed sensor data, let:

$\hat{C}_{j,k} \equiv \hat{C}_{j,k}(\boldsymbol{\theta}_{1:t}, \boldsymbol{\phi}) \equiv \hat{C}(\mathbf{s}_j, \mathcal{T}_{j,k}; \boldsymbol{\theta}_{1:t}, \boldsymbol{\phi})$, for $\mathcal{T}_{j,k} \subset (\tau_0, \tau_t]$.

$\hat{\mathbf{C}}_t \equiv \hat{\mathbf{C}}_t(\boldsymbol{\theta}_{1:t^*}, \boldsymbol{\phi}) \equiv \{\hat{C}_{j,k} : (j,k) \in \mathcal{I}_t\}$, $t \leq t^*$, the model-predicted concentrations corresponding to $\mathbf{d}_t$.

The chosen dispersion model is not perfect, but an approximation to the underlying physical dispersion processes. Therefore, we further define:

$C(\mathbf{s}, \tau) \equiv$ the true (unknown) concentration at location $\mathbf{s}$ at time τ.

$C(B, \mathcal{T}) \equiv$ the true (unknown) average concentration over spatial block B in the time-interval $\mathcal{T}$.

In particular, let:

$C_{j,k} \equiv C(\mathbf{s}_j, \mathcal{T}_{j,k})$ the true (unknown) concentration corresponding to the k-th observation from the j-th sensor.

$\mathbf{C}_t \equiv \{C_{j,k} : (j,k) \in \mathcal{I}_t\}$, the true concentrations corresponding to $\mathbf{d}_t$.

The observed average concentrations, $\{c_{j,k}\}$, are thought to be related to the unknown true average concentrations, $\{C_{j,k}\}$, through

$$c_{j,k} = C_{j,k} + (\text{measurement error}).$$

The $C_{j,k}$'s are then thought to be related to the model predicted concentrations via,

$$C_{j,k} = \hat{C}_{j,k} + (\text{model error}).$$

3. On Atmospheric Dispersion Models

A core component of an atmospheric event reconstruction is the efficient use of atmospheric dispersion models. The complexity of these models ranges from the relatively simple and fast 2D Gaussian puff model INPUFF (Petersen & Lavdas, 1986) to the more complicated coupling of the National Atmospheric Release Advisory Center's (NARAC) dispersion code with assimilated observation and forecast meteorological data. For more computationally demanding dispersion models such as NARAC's LODI code (Nasstrom et al., 2000) and finite-element dispersion models such as NARAC's FEM3MP (Chan & Stevens, 2000), it is desirable to limit the number

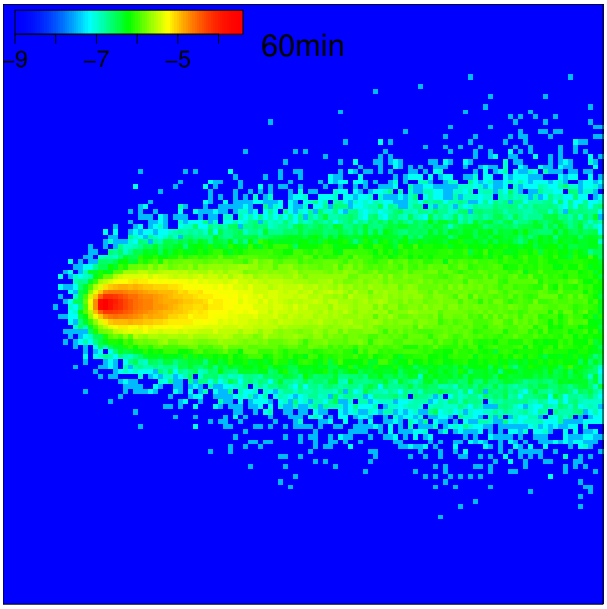

Figure 1: An example of LODI-predicted, near-surface, log-concentration plume due to a single point source emitting at a constant rate.

of model runs while exploring possible input parameters. We consider two conditions which, if satisfied for a given model, can dramatically accelerate parameter exploration.

Let

$\hat{G}^{\boldsymbol{\phi}}_{\mathbf{x},\mathcal{T}}(\mathbf{x}',\mathcal{T}') \equiv$ the average model-predicted concentration at location $\mathbf{x}'$ over the time interval $\mathcal{T}'$ due to a source at location $\mathbf{x}$ with a release rate of 1 in the time interval $\mathcal{T}$, zero release rate outside of $\mathcal{T}$.

Hence, $\hat{G}$ gives the concentration at $(\mathbf{x}',\mathcal{T}')$ due to a unit release-pulse at $(\mathbf{x},\mathcal{T})$. One can think of $\hat{G}$ as tracking the release forward in time, from the source to the sensor. Consider now a single point-source located at $\mathbf{x}_t$ in the t-th time interval and releasing at the rate of r_t; that is, $\boldsymbol{\theta}_t = \{\mathbf{x}_t, r_t\}$. We say that the dispersion model (code) is *linear* if

$$\text{(L):}\quad \hat{C}(\mathbf{x}',\mathcal{T}';\boldsymbol{\theta}_{1:T},\boldsymbol{\phi}) = \sum_{t=1}^{T} \hat{G}^{\boldsymbol{\phi}}_{\mathbf{x}_t,\mathcal{T}_t}(\mathbf{x}',\mathcal{T}')r_t. \quad (1)$$

That is, if the average predicted concentration at location $\mathbf{x}'$ over the time interval $\mathcal{T}'$, due to a point-source characterized by $\boldsymbol{\theta}_{1:T}$, can be written as the sum of scaled "unit-puffs" from each individual time period. Note that for a given series of source locations, $\mathbf{x}_{1:T}$, $\hat{C}$ can be evaluated for various release rate histories, $r_{1:T}$, without needing to recompute $\hat{G}$ (which requires running the dispersion model code).

How $\hat{G}$ is computed depends on the dispersion model in question. But typically, for a given source location $\mathbf{x}$ and a time period $\mathcal{T}$, a single dispersion model run yields

$$\mathcal{G}^{\boldsymbol{\phi}}_{\mathbf{x},\mathcal{T}} \equiv \{\hat{G}^{\boldsymbol{\phi}}_{\mathbf{x},\mathcal{T}}(\mathbf{x}'_j,\mathcal{T}'_j)\},$$

for a pre-specified set of receiving sensors, $\{\mathbf{x}'_j,\mathcal{T}'_j\}$, for example, a regular grid covering the spatial domain of interest. Hence, if the source location is changed in a given time period, a single dispersion model run is needed to recompute $\hat{G}$ for that period at all sensor sites, and $\hat{C}$ can be recomputed using (1). Clearly this is an improvement over evaluating $\hat{C}$ directly for the whole source period.

In atmospheric event reconstruction we are interested in trying out various source locations conditional on data from a given set of sensors. In some cases we can see the impact of various sources on a given sensor in a single computer run. We say that a dispersion model is *reversible* if

$$\text{(R):}\quad \hat{G}^{\boldsymbol{\phi}}_{\mathbf{x},\mathcal{T}}(\mathbf{x}',\mathcal{T}') = \hat{H}^{\boldsymbol{\phi}}_{\mathbf{x}',\mathcal{T}'}(\mathbf{x},\mathcal{T}), \quad (2)$$

where $\hat{H}$ is a "backward" dispersion model complement of $\hat{G}$. Typically, $\hat{H}$ can be computed by running the same dispersion code as is used to compute $\hat{G}$, except in a "reverse mode", where the source is placed at the sensor location and some of the input parameters are modified (e.g., wind-fields reversed, etc.). If a computer code is reversible, we can in a single run compute

$$\mathcal{H}^{\boldsymbol{\phi}}_{\mathbf{x}',\mathcal{T}'} \equiv \{\hat{H}^{\boldsymbol{\phi}}_{\mathbf{x}',\mathcal{T}'}(\mathbf{x}_i,\mathcal{T}_i)\} = \{\hat{G}^{\boldsymbol{\phi}}_{\mathbf{x}_i,\mathcal{T}_i}(\mathbf{x}',\mathcal{T}')\},$$

for a given (large) set of potential sources, $\{\mathbf{x}_i,\mathcal{T}_i\}$. In this case, the number of dispersion model runs needed is equal to the number of sensor observations; not equal to the number of potential source locations, which can be very large.

4. Model Development: A Bayesian Approach

Given data from t time periods, $\mathbf{d}_{1:t}$, our main goal is to conduct inference on the source parameters, $\boldsymbol{\theta}_{1:t}$ via the posterior distribution,

$$\pi_t(\boldsymbol{\theta}_{1:t}) \equiv p(\boldsymbol{\theta}_{1:t} \,|\, \mathbf{d}_{1:t}),$$

the probability distribution of the source parameters, $\boldsymbol{\theta}_{1:t}$, conditional on the observed data, $\mathbf{d}_{1:t}$. In addition, as our setup is dynamic and we seek a smooth transition from posterior inference at time $t-1$ to time t; from $\pi_{t-1}(\boldsymbol{\theta}_{1:t-1})$ to $\pi_t(\boldsymbol{\theta}_{1:t})$.

The source terms are not the only parameters of interest as we also seek posterior inference on the impact of the release, that is, on the resulting concentration levels,

$$\pi_t(C(\mathbf{u},\tau)) \equiv p(C(\mathbf{u},\tau)\,|\,\mathbf{d}_{1:t}),\ 0<\tau\le\tau_t,\ \mathbf{u}\in D,$$

where D is our spatial domain of interest. In general, we seek to have access to the joint posterior

$$\pi_t(\mathbf{C}_{1:t},\boldsymbol{\theta}_{1:t},\boldsymbol{\phi}) \equiv p(\mathbf{C}_{1:t},\boldsymbol{\theta}_{1:t},\boldsymbol{\phi}\,|\,\mathbf{d}_{1:t}).$$

The problem setup, as introduced in the previous section, has two natural hierarchical components, one due to the temporal dynamics and the second one due to the from-source-to-sensor flow. We therefore adopt a dynamic model with a natural conditional hierarchy within each time step. When proceeding from time period $t-1$ to t, the basic model components are:

Data Model: A conditional probability distribution describing the possible variation in the newly available sensor data, $\mathbf{d}_t$, given the (hypothesized) true underlying concentrations and past data,

$$p(\mathbf{d}_t\,|\,\mathbf{C}_t,\mathbf{d}_{1:t-1},\mathbf{C}_{1:t-1},\boldsymbol{\xi}), \tag{3}$$

where $\boldsymbol{\xi}$ is a vector of (potential) measurement-error parameters.

Process Model: A probability model describing the possible variation in $\mathbf{C}_t$ given past values and current and past model predictions,

$$\begin{aligned} &p(\mathbf{C}_t\,|\,\hat{\mathbf{C}}_t,\mathbf{C}_{1:t-1},\hat{\mathbf{C}}_{1:t-1},\boldsymbol{\eta}) \\ &\quad = p(\mathbf{C}_t\,|\,\mathbf{C}_{1:t-1},\boldsymbol{\theta}_{1:t},\boldsymbol{\phi},\boldsymbol{\eta}), \end{aligned} \tag{4}$$

where $\boldsymbol{\eta}$ is a vector of model-error parameters and the second expression follows since $\hat{\mathbf{C}}_{t^*} = \hat{\mathbf{C}}_{t^*}(\boldsymbol{\theta}_{1:t},\boldsymbol{\phi})$, $t^*=1,\ldots,t$.

Parameter Model: A prior parameter model:

$$p(\boldsymbol{\theta}_t\,|\,\boldsymbol{\theta}_{1:t-1})p(\boldsymbol{\phi})p(\boldsymbol{\xi})p(\boldsymbol{\eta}). \tag{5}$$

Note that, jointly,

$$\begin{aligned} &p(\mathbf{C}_t,\boldsymbol{\theta}_t\,|\,\mathbf{C}_{1:t-1},\boldsymbol{\theta}_{1:t-1},\boldsymbol{\phi},\boldsymbol{\eta}) \\ &\quad = p(\mathbf{C}_t\,|\,\mathbf{C}_{1:t-1},\boldsymbol{\theta}_{1:t},\boldsymbol{\phi},\boldsymbol{\eta})p(\boldsymbol{\theta}_t\,|\,\boldsymbol{\theta}_{1:t-1}). \end{aligned}$$

The joint distribution of the model parameters can therefore be written as

$$\begin{aligned} &p(\mathbf{C}_{1:t},\boldsymbol{\theta}_{1:t},\boldsymbol{\phi},\boldsymbol{\eta},\boldsymbol{\xi}) = p(\boldsymbol{\xi})p(\boldsymbol{\eta})p(\boldsymbol{\phi}) \\ &\quad\times\prod_{t^*=1}^{t} p(\mathbf{C}_{t^*},\boldsymbol{\theta}_{t^*}\,|\,\mathbf{C}_{1:t^*-1},\boldsymbol{\theta}_{1:t^*-1},\boldsymbol{\phi},\boldsymbol{\eta}), \end{aligned}$$

where we define $\mathbf{C}_{1:0} = \boldsymbol{\theta}_{1:0} = \emptyset$ (an empty set of parameters), and similarly for other parameters.

Due to the dynamic setup and the hierarchical breakdown within each time step, inference flows naturally from time period $t-1$ to t. Assume at time $t-1$ we have access to the joint posterior distribution of all parameters of interest,

$$\begin{aligned} &\pi_{t-1}(\mathbf{C}_{1:t-1},\boldsymbol{\theta}_{1:t-1},\boldsymbol{\phi},\boldsymbol{\eta},\boldsymbol{\xi}) \\ &\quad = p(\mathbf{C}_{1:t-1},\boldsymbol{\theta}_{1:t-1},\boldsymbol{\phi},\boldsymbol{\eta},\boldsymbol{\xi}\,|\,\mathbf{d}_{1:t-1}). \end{aligned}$$

The dynamic parameter model (5) for the source terms yields the one-step-ahead predictive distribution as

$$\begin{aligned} \pi_{t-1}(\boldsymbol{\theta}_{1:t}) &= p(\boldsymbol{\theta}_{1:t}\,|\,\mathbf{d}_{1:t-1}) \\ &= p(\boldsymbol{\theta}_t\,|\,\boldsymbol{\theta}_{1:t-1})p(\boldsymbol{\theta}_{1:t-1}\,|\,\mathbf{d}_{1:t-1}) \\ &= p(\boldsymbol{\theta}_t\,|\,\boldsymbol{\theta}_{1:t-1})\pi_{t-1}(\boldsymbol{\theta}_{1:t-1}). \end{aligned}$$

More generally, jointly we have that,

$$\begin{aligned} &\pi_{t-1}(\mathbf{C}_{1:t},\boldsymbol{\theta}_{1:t},\boldsymbol{\phi},\boldsymbol{\eta},\boldsymbol{\xi}) \\ &\quad = p(\mathbf{C}_t,\boldsymbol{\theta}_t\,|\,\mathbf{C}_{1:t-1},\boldsymbol{\theta}_{1:t-1},\boldsymbol{\phi},\boldsymbol{\eta}) \\ &\qquad\times\pi_{t-1}(\mathbf{C}_{1:t-1},\boldsymbol{\theta}_{1:t-1},\boldsymbol{\phi},\boldsymbol{\eta},\boldsymbol{\xi}). \end{aligned}$$

Then, given (potential) new data at time step t, the joint posterior at t is given by

$$\begin{aligned} &\pi_t(\mathbf{C}_{1:t},\boldsymbol{\theta}_{1:t},\boldsymbol{\phi},\boldsymbol{\eta},\boldsymbol{\xi}) \propto p(\mathbf{d}_t\,|\,\mathbf{C}_t,\mathbf{d}_{1:t-1},\mathbf{C}_{1:t-1},\boldsymbol{\xi}) \\ &\qquad\times\pi_{t-1}(\mathbf{C}_{1:t},\boldsymbol{\theta}_{1:t},\boldsymbol{\phi},\boldsymbol{\eta},\boldsymbol{\xi}). \end{aligned} \tag{6}$$

We shall now discuss the individual model-steps in more details, while in the next section we shall outline how one can conduct inference using the above joint posterior distribution.

4.1 The Measurement Data Model

The measurement data model (3) relates the observed data to the unobserved concentration plume due to the release. If the data is recorded by a network of well established sensors, one can often elicited the needed distributions from the sensors' specifications.

A sensor's accuracy is often specified in terms of its detection and saturation limits, along with its within-range accuracy (Gibbons & Coleman, 2001). The measurement errors can often be assumed to be independent between different sensors and also roughly temporally uncorrelated. As such, we can write the data model (3) as,

$$\begin{aligned} &p(\mathbf{d}_t\,|\,\mathbf{C}_t,\mathbf{d}_{1:t-1},\mathbf{C}_{1:t-1},\boldsymbol{\xi}) = p(\mathbf{d}_t\,|\,\mathbf{C}_t) \\ &\quad = \prod_{(j,k)\in\mathcal{I}_t} p(c_{j,k}\,|\,C_{j,k}), \end{aligned} \tag{7}$$

where $p(c_{j,k} \mid C_{j,k})$ is the distribution of the k-th observation from the j-th sensor; $(j,k) \in \mathcal{I}_t$. Given the specifications of the instrument, we assume

$$p(c_{j,k} \mid C_{j,k}) = \begin{cases} f(c_{j,k}; C_{j,k}), & \text{if } l < c_{j,k} < u, \\ \int_0^l f(dc; C_{j,t}), & \text{if } c_{j,k} = l, \\ \int_u^\infty f(dc; C_{j,k}), & \text{if } c_{j,k} = u, \end{cases}$$

where f is a given probability density, l is the sensor's detection limit, and u is the sensor's saturation limit. We take the density f to be given by a truncated Gaussian distribution (truncated below zero) with untruncated mean equal to $C_{j,k}$ and variance $a^2 + b^2 * C_{j,k}^2$, with a and b elicited from the sensor's specifications. Note that the untruncated Gaussian distribution has standard deviation a when $C_{j,k} = 0$, which can be related to the detection limit of the instrument (i.e., its sensitivity at low concentrations). For large concentrations, the relative standard deviation is approximately b, which again can be related to the sensor's spec.

4.2 The Concentration Process Model

The concentration model (4) captures the model-error associated with using the dispersion model to predict the impact of the release. This error process is meant to capture both the physical (and numerical) errors associated with the model and the impact of feeding it with inaccurate input parameters (e.g., meteorology). We expect to see both spatial and temporal correlated model-errors. Additionally, we expect the (marginal) variation of the errors to increase with the predicted concentration levels. However, the exact form of both the correlation structure and the amount of variation is tightly linked to the dispersion model in question; a less accurate and capable dispersion model needs a more flexible error-model than a better established dispersion model. In addition, we prefer to specify the model in the dynamic fashion as hinted at in (4). The conditional specification in (4) allows for an easier transition from time period $t-1$ to t than a joint specification.

At time t, with new data $\mathbf{d}_t = \{c_{j,k} : (i,j) \in \mathcal{I}_t\}$, we assume that the needed concentrations for (7) are given by,

$$\begin{aligned} \log C_{j,k} &= \log \hat{C}_{j,k}(\boldsymbol{\theta}_{1:t}, \boldsymbol{\phi}) + \delta_{j,k} \\ &= \log \hat{C}_{j,k} + \delta_{j,k}, \quad (j,k) \in \mathcal{I}_t, \end{aligned} \tag{8}$$

where $\delta_{j,k} = \delta(\mathbf{s}_j, \mathcal{T}_{j,k})$ is the model-error term (on log-scale); note that we are conditioning on the model input parameters $\boldsymbol{\theta}_{1:t}$ and $\boldsymbol{\phi}$. This approach mirrors that suggested by Bayarri et al. (2005). Since the model is specified on log-scale, $C_{j,k}$ is never exactly equal to zero. However, this is not of practical concern. In the case when $\hat{C}_{j,k}$ is predicted to be exactly equal to zero, we replace it by a small value which has no practical difference from zero concentration (in terms of sensors' detection limits, etc.).

The key component to (8) is how the "bias" term, $\delta_{j,k}$, is treated. Let

$$\boldsymbol{\delta}_t \equiv \mathbf{vec}\{\delta_{j,k} : (j,k) \in \mathcal{I}_t\}.$$

We are currently investigating the following autoregressive model for the $\boldsymbol{\delta}_t$'s;

$$\boldsymbol{\delta}_t = \mathbf{F}_{1:t-1}\boldsymbol{\delta}_{1:t-1} + \boldsymbol{\nu}_t,$$

where $\mathbf{F}_{1:t-1}$ is a known matrix and $\boldsymbol{\nu}_t = \mathbf{vec}\{\nu_{j,k} : (j,k) \in \mathcal{I}_t\}$ is a stochastic error term. We take the elements of $\mathbf{F}_{1:t-1}$ to be such that,

$$\delta_{j,k} = \delta_{j,k-1} + \nu_{j,k}, \quad (j,k) \in \mathcal{I}_t,$$

where $\boldsymbol{\delta}_0 \equiv \mathbf{vec}\{delta_{j,0}\}$ is given an initial distribution. The error term $\boldsymbol{\nu}_t$ is assumed multivariate Gaussian with zero mean and variance-covariance matrix that exhibits spatial and temporal correlation, and variation that depends on the predicted concentration levels.

If the dispersion model is linear (1), an alternative approach might be to model the errors associated with $\hat{G}_{\boldsymbol{\phi}}$ in (1); that is, write

$$C(\mathbf{s}, \mathcal{T}) = \sum_{t=1}^{T} G_t(\mathbf{s}, \mathcal{T}),$$

where $G_t(\mathbf{s}, \mathcal{T}) = \hat{G}^{\boldsymbol{\phi}}_{\mathbf{x}_t, \mathcal{T}_t}(\mathbf{s}, \mathcal{T}) r_t +$ error, in a the case of a single source, with a natural extension to multiple sources.

4.3 Source and Input Parameters Model

The release can be due to a single or multiple sources, it can be stationary or moving, and have a very complex release-rate profile. Hence, the source parameter model (5), $p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_{1:t-1})$, needs to be very flexible.

We approximate the impact of the i-th source in the t-th time period using a point source at location $\mathbf{x}_{t,i}$ emitting at a constant release rate of $r_{t,i}$; $\boldsymbol{\theta}_{t,i} = \{\mathbf{x}_{t,i}, r_{t,i}\}$. For stationary sources with slowly varying release rates (compared to the length of the time periods $\{\mathcal{T}_t\}$), one expects this representation to be sufficient. However, for continuously moving sources, the time periods $\{\mathcal{T}_t\}$ need to be sufficiently

small to reduce the impact that results from the point-source representation (i.e., the source "jumps" from one location to another as one moves from one time period to the next).

We assume that the dynamics of the location of the i-th source to be given by,

$$\mathbf{x}_{t,i} = \mathbf{x}_{t-1,i} + \boldsymbol{\nu}^x_{t,i}, \quad t = 1, 2, \ldots,$$

where $\mathbf{x}_{0,i}$ is given an initial distribution $f^x_i(\cdot)$ and $\boldsymbol{\nu}^x_{t,i}$ are independently, zero-mean Gaussian distributed. Note that if the $\{\boldsymbol{\nu}_{t,i}\}$ are given a zero variance the source is stationary. The model above can easily be extended to include more sophisticated target-tracking dynamics (see, e.g., Ramachandra, 1999)

For the release rate, we assume that,

$$\log r_{t,i} = \log r_{t-1,i} + \nu^r_{t,i}, \quad \text{if } r_{t-1,i} > 0,$$

where $\nu^r_{t,i}$ are independently, zero-mean Gaussian distributed. If $r_{t-1,i} = 0$, then we assume that $r_{t,i}$ is either zero or follows an initial non-zero release-rate distribution;

$$p(r_{t,i} \,|\, r_{t-1,}) = \alpha_i I(r_{t,i} = 0) + (1 - \alpha_i) f^r_i(r_{t,i}),$$

where α_i is a given mixture probability and $f^r_i(\cdot)$ a given initial release-rate distribution.

We take the prior for the additional model input parameters $\boldsymbol{\phi}$ to be given by an ensemble of *a priori* likely values;

$$p(\boldsymbol{\phi}) = \sum_{i=1}^{M} \omega_i I(\boldsymbol{\phi} = \boldsymbol{\phi}_i),$$

where the $\{\omega_i\}$ are weights that sum to 1 and $\{\boldsymbol{\phi}_i\}$ a pre-specified ensemble.

5. Posterior Inference

We resort to sampling-based methods to conduct inference based on the joint posterior distribution (6). In designing posterior sampler for (6) there are two criteria we keep in mind: (1) efficient generation of an initial posterior sample at any given time period and (2) smooth transition from a posterior sample from one time period to the next. We shall now outline the basic steps taken; further details are in Johannesson et al. (2004).

The event reconstruction is initiated when concentrations above limit of detection are observed for the first time. Hence, at a given time period t_0, we seek to draw realizations from

$$\pi_{t_0}(\mathbf{C}_{1:t_0}, \boldsymbol{\theta}_{1:t_0}, \boldsymbol{\phi}, \boldsymbol{\eta}, \boldsymbol{\xi}),$$

where one or more of the sensor concentrations of $\mathbf{d}_{1:t_0}$ are above the limit of detection. The second inference (and the first of many to come) is when a new data $\mathbf{d}_{t_0+1}$ becomes available and we want to draw realizations from

$$\pi_{t_0+1}(\mathbf{C}_{1:t_0+1}, \boldsymbol{\theta}_{1:t_0+1}, \boldsymbol{\phi}, \boldsymbol{\eta}, \boldsymbol{\xi}),$$

without completely discarding the effort that went into generating realizations from time periods 1 to t_0. The approach we take is to first carry out a classical Markov chain Monte Carlo (MCMC) sampling from the posterior distribution π_{t_0} and then update those realizations using sequential Monte Carlo (SMC) techniques when proceeding to sample from π_{t_0+1}. The basic ingredients needed are Markov chain transition kernels for MCMC and important sampling proposal distributions for SMC.

In what follows, let

$$\mathbf{X}_t \equiv \{\mathbf{C}_t, \boldsymbol{\theta}_t\}$$

and $\mathbf{X}_{1:t} = \{\mathbf{C}_{1:t}, \boldsymbol{\theta}_{1:t}\}$.

5.1 Markov Chain Transition Kernels

A joint Markov chain transition kernel is constructed using a sequence of proposal distributions with the resulting joint proposal being either accepted or rejected according to the Metropolis-Hasting acceptance ratio (e.g., Gelman et al., 2004, p. 289).

Let $\mathbf{X}_{1:t}$ be the current state of the Markov chain; for the moment we shall ignore other (potential) non-dynamic parameters ($\boldsymbol{\phi}$, $\boldsymbol{\eta}$, and $\boldsymbol{\xi}$). A new state, $\mathbf{X}^*_{1:t}$, is proposed by modifying a subset of the parameters making up $\mathbf{X}_{1:t}$ through a sequence of proposal distributions;

$$\begin{aligned} q_t(\mathbf{X}^*_{1:t} \,|\, \mathbf{X}_{1:t}) = \\ q_t(\boldsymbol{\theta}^*_{1:t} \,|\, \boldsymbol{\theta}_{1:t}, \mathbf{C}_{1:t}) q_t(\mathbf{C}^*_{1:t} \,|\, \boldsymbol{\theta}^*_{1:t}, \mathbf{C}_{1:t}). \end{aligned} \tag{9}$$

Hence, a source parameters proposal is carried out first, followed by model-error proposal (i.e., proposal for the true concentration levels). The joint proposal $\mathbf{X}_{1:t} = \{\mathbf{C}^*_{1:t}, \boldsymbol{\theta}^*_{1:t}\}$ is then either accepted or rejected according to the Metropolis-Hastings proposal ratio,

$$\begin{aligned} \rho_t(\mathbf{X}^*_{1:t}; \mathbf{X}_{1:t}) = \\ \frac{p(\mathbf{d}_{1:t} \,|\, \mathbf{X}^*_{1:t}) p(\mathbf{X}^*_{1:t}) q_t(\mathbf{X}_{1:t} \,|\, \mathbf{X}^*_{1:t})}{p(\mathbf{d}_{1:t} \,|\, \mathbf{X}_{1:t}) p(\mathbf{X}_{1:t}) q_t(\mathbf{X}^*_{1:t} \,|\, \mathbf{X}_{1:t})}, \end{aligned} \tag{10}$$

with $\mathbf{X}^*_{1:t}$ becoming the next state of the Markov chain if $\rho_t(\mathbf{X}^*_{1:t}; \mathbf{X}_{1:t}) > u$, where u is a random draw from a uniform distribution between 0 and 1.

For the source parameters $\boldsymbol{\theta}_{1:t}$ we adopt random-walk proposal distributions for both the source location vectors and the release-rate vectors. We have

experimented both with "blind" random-walk proposal and Langevin-type (gradient-based) proposal distributions (e.g., Liu, 2001, ch. 9) with good results. Both the source location and the release-rate proposal steps include a random components which determine how many parameters and which to propose changes to, including leaving the source parameters unchanged.

The concentration proposal is carried out through the model-errors of (8), conditioning on the predicted concentrations $\hat{C}_{1:t}(\boldsymbol{\theta}^*_{1:t}, \boldsymbol{\phi})$. We are currently exploring multi-variate random walk for the model-errors $\{\boldsymbol{\delta}_t\}$ and hyper-parameters associated with the model errors $(\boldsymbol{\eta})$.

Proposal distribution can be created for other model parameters (e.g., $\boldsymbol{\xi}$ and $\boldsymbol{\phi}$), if not fixed, and added to the joint proposal (9).

5.2 Sequential Proposal Distributions

For updating past posterior realizations from time periods 1 through $t-1$ to reflect newly available data $\mathbf{d}_t$, we use sequential Monte Carlo (SMC) methods; see, for example Doucet et al. (2001) and Liu (2001). For the time dependent parameters of the model, SMC needs a dynamic proposal distribution that generates realizations of $\mathbf{X}_{1:t}$, conditional on a realizations of $\mathbf{X}_{1:t-1}$ from π_{t-1}. These new realizations are then weighted to reflect the new data.

In most cases the new data $\mathbf{d}_t$ carries limited information about the most recent source parameters, $\boldsymbol{\theta}_t$, as it can take some time for the source to disperse to the sensors. That is, $\boldsymbol{\theta}_t$ has in general little impact on the predicted concentrations $\hat{\mathbf{C}}_t(\boldsymbol{\theta}_{1:t}, \boldsymbol{\phi})$, with earlier release parameters mostly being responsible for the current concentration levels at the sensor sites. This suggests we can be relaxed in how we exactly extend $\mathbf{X}_{1:t-1}$ to $\mathbf{X}_{1:t}$, but need to be more concerned about the impact of the new data on past model parameters $(\mathbf{X}_{1:t-1})$. We therefore adopt the auxiliary SMC method suggested by Pitt & Shephard (1999, 2001) to select which of the past realizations are carried forward, followed by MCMC rejuvenation, as suggested by MacEachern et al. (1999), Gilks & Berzuini (2001), and Godsill & Clapp (2001). The whole procedure is outlined in Table 1. The auxiliary SMC steps are 1 and 2. Step 1 assigns weights to past realizations $\{\mathbf{X}^{(i)}_{1:t-1}\}$ that partly reflect the new data. To do this, a deterministic extension of $\mathbf{X}_{1:t-1}$ to time period t is carried out in step 1.1, via the function $\hat{f}_t$, which can be taken to be very simple (e.g., assume zero release at t). The newly created weights $\{v^{(i)}_{1:t-1}\}$ are used to discriminate between past realizations when choosing which one to extend to time period t via classical SMC step; step 2. Step 2 yields a weighted importance sample $\{\tilde{w}^{(i)}_{1:t}, \tilde{\mathbf{X}}^{(i)}_{1:t}\}$. Note that so far, model parameters associated with the time period 1 to $t-1$, and parameters that do not vary with time, have not been altered from previous realizations. The MCMC step 3 carries out perturbation of past (and current) model parameters, using the MCMC transition kernels of previous section.

Table 1: SMC-MCMC Hybrid.

0. (Initial Sample) Start with the initial, equal-weighted sample $\{\mathbf{X}^{(i)}_{1:t_0} : i = 1, \ldots, N\}$, derived from the initial MCMC samples.

For $t = t_0 + 1, \ldots$:

1 (Proposal Weights) For $i = 1, \ldots, N$:

1.1 Put $\hat{\mathbf{X}}^{(i)}_t = \hat{f}_t(\mathbf{X}^{(i)}_{1:t-1})$.

1.2 Compute $v^{(i)}_{1:t-1} = p(\mathbf{d}_t \,|\, \hat{\mathbf{X}}^{(i)}_t)$.

2 (Extending to time t) For $i = 1, \ldots, N$:

2.1 Sample $\tilde{I}_i \in \{1, \ldots, N\}$ with $p(\tilde{I}_i = j) \propto v^{(j)}_{1:t-1}$; $j = 1, \ldots, N$.

2.2 Generate $\tilde{\mathbf{X}}^{(i)}_t \sim q_t(\mathbf{X}_t \,|\, \mathbf{X}^{(\tilde{I}_i)}_{1:t-1})$ and let $\tilde{\mathbf{X}}^{(i)}_{1:t} \equiv (\mathbf{X}^{(\tilde{I}_i)}_{1:t-1}, \tilde{\mathbf{X}}^{(i)}_t)$.

2.3 Compute the importance-sample weight

$$\tilde{w}^{(i)}_{1:t} = \frac{p(\mathbf{d}_t \,|\, \tilde{\mathbf{X}}^{(i)}_t) p(\tilde{\mathbf{X}}^{(i)}_t \,|\, \tilde{\mathbf{X}}^{(i)}_{1:t-1})}{q_t(\tilde{\mathbf{X}}^{(i)}_t \,|\, \mathbf{X}^{(i)}_{1:t-1})} \frac{1}{v^{(i)}_{1:t-1}}.$$

3 (MCMC Perturbation) For $i = 1, \ldots, N$:

3.1 Select $\tilde{I}_i \in \{1, \ldots, N\}$ with $p(\tilde{I}_i = j) \propto \tilde{w}^{(j)}_{1:t}$; $j = 1, \ldots, N$, and put $\mathbf{X}^{(i,0)}_{1:t} = \tilde{\mathbf{X}}^{(\tilde{I}_i)}_{1:t}$.

3.2 For $j = 1, \ldots, B$: [B given]

3.2.1 Propose $\check{\mathbf{X}}_{1:t} \sim q_t(\check{\mathbf{X}}_{1:t} \,|\, \mathbf{X}^{(i,j-1)}_{1:t})$.

3.2.2 Compute the M-H ratio (10), and if accepted put $\mathbf{X}^{(i,j)}_{1:t} = \check{\mathbf{X}}_{1:t}$, otherwise put $\mathbf{X}^{(i,j)}_{1:t} = \mathbf{X}^{(i,j-1)}_{1:t}$.

3.3 Put $\mathbf{X}^{(i)}_{1:t} = \mathbf{X}^{(i,B)}_{1:t}$, then $\{\mathbf{X}^{(i)}_{1:t} : i = 1, \ldots, N\}$ is an equal-weighted sample from $\pi_t(\mathbf{X}_{1:t})$.

6. Discussion

We have carried out numerous studies testing different aspects of the proposed framework with great success; see, for example, Johannesson et al. (2004) and Chow et al. (2006). As outlined, the framework is able to efficiently take advantage of large high performance parallel computers, even if the dispersion model itself is not parallelized. Further work aims at investigating models that capture the error associated with the dispersion model and finalizing the implementation of the atmospheric event reconstruction computer framework.

7. Acknowledgment

This work was funded by LDRD project number 04-ERD-037 and was performed under the auspices of the U.S. Department of Energy by the University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

References

Bayarri, M. J., Berger, J. O., Paulo, R., Sacks, J., Cafeo, J. A., Cavendish, J., Lin, C. H., & Tu, J. (2005). A framework for validation of computer models. Technical report, National Institution of Statistical Sciences.

Chan, S. & Stevens, D. (2000). An evaluation of two advanced turbulence models for simulating the flow and dispersion around buildings. In *The Proceedings of the Millennium NATA/CCMS Int. Tech. Meeting on Air Pollution Modeling and its Application*, (pp. 355–362)., Boulder, CO. The American Meteorological Society.

Chow, F. K., Kosovic, B., & Chan, S. T. (2006). Source inversion for contaminant plume dispersion in urban environments using building-resolving simulations. In *The Proceedings of the Sixth Symposium on the Urban Environment*, Atlanda, GA. The American Meteorological Society.

Doucet, A., de Freitas, J. F. G., & Gordon, N. J. (2001). *Sequential Monte Carlo methods in practice*. New York: Springer.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2004). *Bayesian Data Analysis* (2nd ed.). Boca Raton, Florida: Hapman and Hall/CRC.

Gibbons, R. D. & Coleman, D. E. (2001). *Statistical Methods for Detection and Quantification of Environmental Contamination*. New York: John Wiley & Sons, Inc.

Gilks, W. R. & Berzuini, C. (2001). Following a moving target — Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society B*, *63*, 127–146.

Godsill, S. & Clapp, T. (2001). Improvement strategies for Monte Carlo particle filters. In A. Doucent, N. de Freitas, & N. Gordon (Eds.), *Sequential Monte Carlo Methods in Practice* (pp. 139–158). New York: Springer.

Johannesson, G., Hanley, W. G., & Nitao, J. (2004). Dynamic Bayesian models via Monte Carlo: an introduction with examples. Technical Report UCRL-TR-207173, Lawrence Livermore National Laboratory.

Liu, J. S. (2001). *Monte Carlo Strategies in Scientific Computing*. New York: Springer.

MacEachern, S. N., Clyde, M., & Liu, J. S. (1999). Sequential importance sampling for nonparametric Bayes models: The next generation. *Canadian Journal of Statistics*, *27*, 251–267.

Nasstrom, J. S., Sugiyama, G., Leone, J. M., & Ermak, D. L. (2000). A real-time atmospheric dispersion modeling system. In *The Proceedings of the 11th Joint Conference on the Applications of Air Pollution Meteorology*, (pp. 285–289)., Long Beach, CA. The American Meteorological Society.

Petersen, W. & Lavdas, L. (1986). Inpuff 2.0: A multiple source gaussian puff dispersion algorithm — user's guide. Technical Report EPA/600/8-86/024, EPA.

Pitt, M. K. & Shephard, N. (1999). Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, *23*, 356–359.

Pitt, M. K. & Shephard, N. (2001). *Sequential Monte Carlo methods in practice*, chapter Auxiliary variable based particle filters. New-York: Springer.

Ramachandra, K. V. (1999). *Kalman Filtering Techniques for Radar Tracking*. New York: Marcel Dekker.