1.1 Temporal and Spatial Reconstruction of Atmospheric Puff Releases using Bayesian Inference

Derek Wade and Inanc Senocak
High Performance Simulation Laboratory for Thermo-fluids
Department of Mechanical & Biomedical Engineering
Boise State University
Boise, Idaho 83725
Email: derekwade@u.boisestate.edu

1 Introduction

An atmospheric dispersion release event involving the transport of airborne contaminants may originate from a continuous or instantaneous source. An instantaneous, or puff, release of hazardous contaminants can be accidentally or intentionally released into the atmosphere where its transport by wind is subject to various atmospheric conditions. Using a network of sensors, one can estimate the parameters of the release source using an appropriate forward model and the measurements obtained from the sensors. In this study, the Stochastic Event Reconstruction Tool (SERT) [1], which was developed to work for continuous releases where time of release was not a factor, is extended to reconstruct instantaneous releases. This extension will be referred to as the Puff Event Reconstruction Tool (PERT) and will be the focus of this paper[1]. In previous work, SERT was also developed into a Multi-source Event Reconstruction Tool (MERT) with a composite model ranking formulation [2].

The goal of this research is to develop a quick and efficient tool for characterizing a contaminant dispersion event once it has been detected by a sensor network. For use in threat reduction and mitigation, it is important for the software tool to run quickly and provide a probabilistic search region. A probabilistic search region may aid decision makers in their response strategies better than a deterministic single point estimate. Previous research in the source term estimation (STE) field has yielded methods that take deterministic and probabilistic approaches. Some researchers, such as [3–7], have used cost function minimization, adjoint models, and other optimization techniques to solve STE problems. Our STE method in [1,2], and the STE methods used by [8–10], use Bayesian inference as the core component of the STE algorithm. A solution adopting this probabilistic approach yields a probabilistic solution, which may be helpful for decision makers.

Using a data-driven Gaussian puff model, discussed in the following section, we estimate the source location, quantity of release, and time of release, along with other model input parameters. We use a Bayesian inference method with Markov chain Monte Carlo (MCMC) sampling to obtain parameter estimates probabilistically.

With the contaminant release time now being a factor in the instantaneous release case, the inverse problem introduces new challenges to the event reconstruction problem. Rather than a single, time-averaged set of concentration data in the continuous case, we must now take multiple time steps of data as input to the event reconstruction tool. To address this new complication faithfully, we make modifications to the original Bayesian inference method in SERT to ensure the convergence of MCMC chains for puff releases.

To ensure a quick and accurate source term reconstruction, we pursue a parallel implementation in

[1]Paper 1.1, 12th Conference on Artificial and Computational Intelligence and its Applications to the Environmental Sciences, 94th AMS annual meetings, Atlanta, GA, February 3rd, 2014

our method for MCMC chains, and determine the convergence of the simulation using statistical post-processing. Thanks to our parallel implementation, the entire simulation takes approximately 5 to 7 minutes using Intel E8400 3.0 GHz processors in a conventional workstation. We validate our method using a real trial scenario from the Fusion Field Trial 2007 (FFT07) study [11], and demonstrate successful temporal and spatial reconstruction with puff releases of passive contaminants.

2 Forward Model and Inference Engine Description

Since this research focuses on short range releases, a Gaussian puff model is an applicable forward model when flat terrain and steady wind conditions are present. In certain cases, where the puff splits or there are varying wind conditions, a different forward model may be selected to account for these additional dispersion factors. The FFT-07 trials were conducted over flat terrain, and certain trials were under approximately steady wind conditions. The primary reasoning behind the use of the Gaussian puff model is its low computational expense. Since our Bayesian inference method relies on many millions of calls to the forward model, the program may be executed on a conventional workstation in only a couple of minutes due to the low computational cost of our forward model. The formulation of the Gaussian puff model [12] used in PERT is shown in Eq. 1.

$$C(x,y,z,t) = \frac{Q}{2\pi^{3/2}\sigma_x\sigma_y\sigma_z\bar{u}}\exp\left(-\frac{(x-x_c)^2}{2\sigma_x^2}-\frac{(y-y_c)^2}{2\sigma_y^2}\right) \times\left[\exp\left(-\frac{(z-H)^2}{2\sigma_z^2}\right)+\exp\left(-\frac{(z+H)^2}{2\sigma_z^2}\right)\right] \quad (1)$$

It was shown in our previous work [1, 2] that formulating the puff diffusion parameters, σ_x, σ_y, and σ_z, in a data-driven manner improved the agreement of the model with the observed concentration field. These diffusion parameters were calculated according to the formulas recommended in [13] for type-D neutral conditions. We also assumed the equation for diffusion in the downwind direction was the same as the crosswind direction, while the coefficient differed. The stochastic parameters ζ_1, ζ_2, and ζ_3 have taken the place of the empirical coefficients in order to make the following diffusion equations data-driven. σ_x, σ_y, and σ_z correspond to the diffusion in the x, y, and z directions, respectively.

$$\sigma_x = \zeta_1 x(1+0.0001x)^{-1/2} \quad (2)$$
$$\sigma_y = \zeta_2 x(1+0.0001x)^{-1/2} \quad (3)$$
$$\sigma_z = \zeta_3 x(1+0.0015x)^{-1/2} \quad (4)$$

To perform our event reconstruction, we first formulate our problem as an inverse problem. An inverse problem may be stated as Eq. 5, where $\mathbf{m}$ is a set of forward model parameters, $\mathbf{d}$ is a set of observed concentration data, and F is our data-driven Gaussian puff forward model.

$$\mathbf{m} \approx F^{-1}(\mathbf{d}) \quad (5)$$

The complete list of the k number of forward model parameters for the case of a single instantaneous release is shown in Eq. 6. Model parameters x and y are the coordinates of the source location. θ is the wind direction calculated as an angle from the positive x-direction of the coordinate system. Q is the total amount of material that was instantaneously released into the atmosphere. As previously stated, ζ_1, ζ_2, and ζ_3, are the stochastic coefficients used in the calculation of downwind puff diffusion in the x, y, and z directions, respectively. Finally, Δt is the time since the contaminant release.

$$\mathbf{m} = [x, y, \theta, Q, \zeta_1, \zeta_2, \zeta_3, \Delta t] \quad (6)$$

The basis of the Bayesian formulation incorporated into our inference engine is Markov chain Monte Carlo (MCMC) via the Metropolis-Hastings algorithm [14]. The Bayesian inference engine used in SERT and MERT is explained in sufficient detail in [1, 2] and will not be in this paper. A sample puff traveling through the sensor domain is shown in Fig. 1.

3 Workflow and Data Stream

For a high-level view of the arrangement of processes involved from start to finish, a workflow diagram has been constructed and is shown in Fig. 2. Each component of the workflow will be discussed in detail in the following sections. First, concentrations are observed by the sensors in the sensor network. This data is then fed to the PERT pre-processor where it is processed and

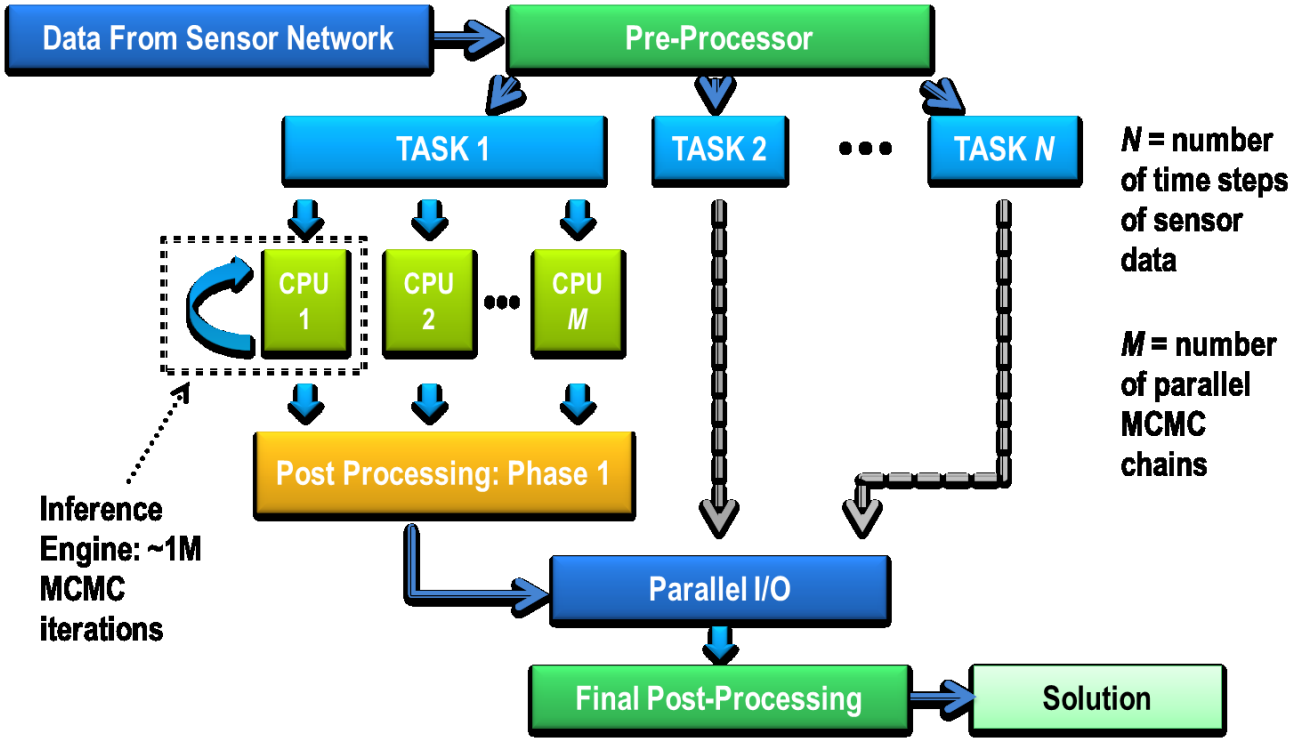

Fig. 2. Overall workflow diagram showing progression of processes and parallel data streams from input at sensor network to output of final solution.

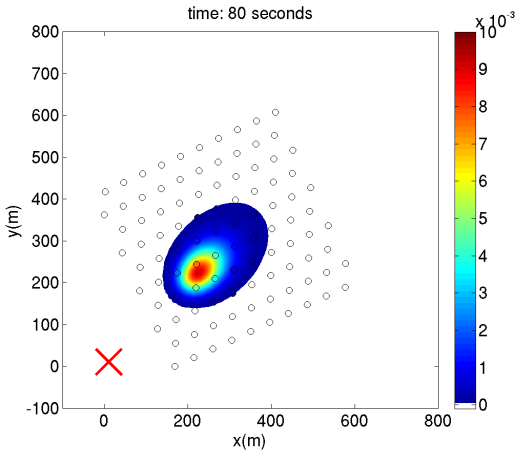

Fig. 1. Sample puff 80 s after release, traveling downwind across sensor domain.

formatted for use by the program. Each time step of data is then assigned to a different parallel task (i.e., N timesteps = N tasks). Multiple parallel MCMC chains are launched for each task, and the inference engine operates on each of these parallel chains. The task specific chains are then post processed and sent to a host process where all task results are combined and processed further. Finally, the solution is returned based on the specified number of time steps to be included in the result.

3.1 Pre-Processor

Once raw data has been collected by the sensor network, it must be processed for input to PERT. The first step is to scan all sensors in the sensor network and determine which sensors were not operating correctly during the period of interest. Using quality control flags in the raw data, sensors were discarded if they were flagged with errors for more than 50% of the run-time for a specific trial.

The raw data was recorded at a frequency of 50 Hz which provided a set of data every 20 ms. Due to the formulation of the forward model, noise signals from the 50 Hz data increased the difficulty of the reconstruction. The 50 Hz data is time averaged over 5 s intervals to produce smoother data and reduce noise in the concentration field. Figure 3 shows the raw data from a sensor over a 200 s span where the contaminant puff enters and then exits the sensor field. This can be seen with the rise and fall of concentration measurements at the sensor. Figure 4 shows the same sensor after time-averaging the data over 5 s intervals to smooth out the noise. The sensor data is then formatted into data files that will be used as input to PERT.

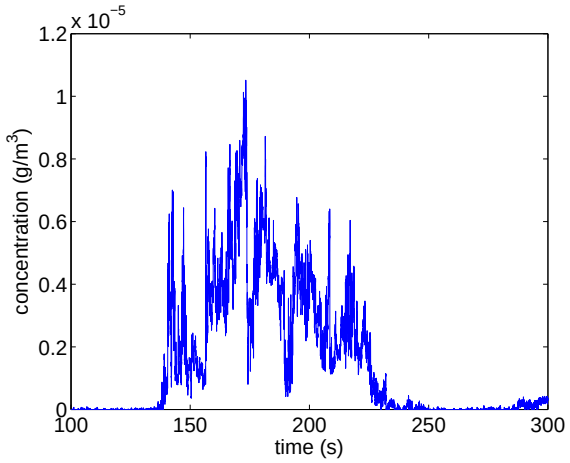

Fig. 3. Raw concentration data from a single sensor in the sensor network over a 200 s time span at 50 Hz.

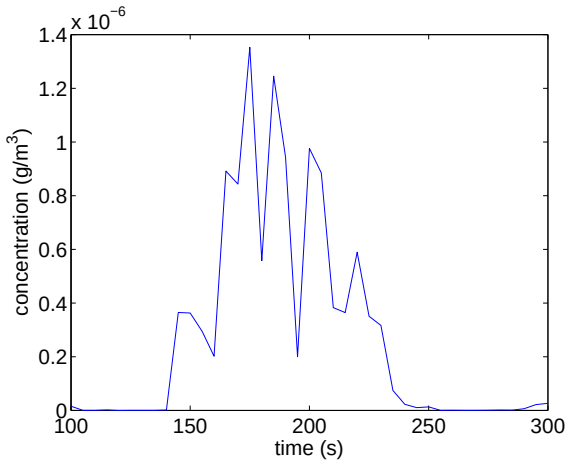

Fig. 4. Data from the same sensor in Fig. 3 time averaged over 5 s intervals.

3.2 Parallel Inference Engine

Each time step of concentration data represents a possible outcome from a forward model with the correct set of input parameters. Therefore, an event reconstruction can be performed for each time step, and, ideally, each reconstruction would produce the same solution. N parallel tasks are launched, representing the N time steps of data available. Multiple reconstructions can also be performed on each task. This includes launching subsequent parallel MCMC chains on each parallel task. An advantage to having multiple MCMC chains for each task is that we can estimate the convergence of our probabilistic solution, which is discussed in the next section.

In the current inference engine, five parallel MCMC chains are launched for each task. Each of the chains is started at a different random point within the state-space and is allowed to run for a set number of iterations. The parallel operations are carried out by Message Passing Interface (MPI) processes. Each of the chains completes 1 million iterations for each of the k model parameters of interest. An in-depth discussion of the Bayesian inference processes can be found in [1] and [2].

3.3 Convergence of Parallel MCMC Chains

It is important to verify the convergence of our simulation so that we may infer the most probable forward model parameters in our solution. Upon completion of the MCMC runs, the first half of each chain is discarded as a conservative burn-in time, and the variance for each individual chain is determined. Those variances are then averaged, and this quantity is referred to as the *average within-chain variance*, or $\sigma^2_{w.c.ave}$. The chains are then mixed together to create one larger chain, and its variance is then determined. The variance of this mixed chain is referred to as the mixture variance, or σ^2_{mix}. As stated in [15], the chains will have mixed upon convergence, so the mixture variance should be the same as the average within chain variance.

A value called the "potential scale reduction factor", or $\hat{R}$, was proposed in [16] and can be used to describe the level of convergence. It is computed according to Eq. 7. The value of $\hat{R}$ should be 1 for a perfectly mixed solution, however [15] suggests that a value less than 1.1 is acceptable in practice. It is important to note that each stochastic parameter must converge, and that checking the convergence of a single variable does not describe the convergence of the complete solution. We consider the solution as converged once all variables achieve $\hat{R} \leq 1.1$.

$$\hat{R} = \sqrt{\frac{\sigma^2_{mix}}{\sigma^2_{w.c.ave}}} \tag{7}$$

The statistical software R [17] is used with the `coda` package [18] to calculate the potential scale reduction factor during our task-level post processing step. Using this package, we are also able to determine a sufficient number of iterations to reach convergence. As described in [15], the `gelman.plot` function in the `coda` package produces a plot of $\hat{R}$ vs. iteration every 50 iterations. This is a visual way to check for false convergence and is quite helpful in determining the correct number of iterations required for a converged result. Figure 5 shows an example plot of the potential scale reduction factor for the y variable. The `gelman.diag` function calculates $\hat{R}$ with a

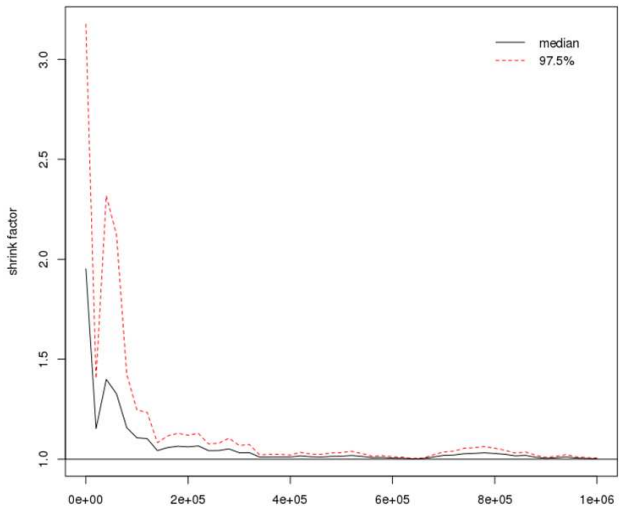

Fig. 5. Potential scale reduction factor for y model parameter showing estimated convergence over 1 million MCMC iterations.

97.5% quantile since the convergence is estimated due to a finite chain length.

3.4 Final Post-Processing

The final step in the workflow is to combine the task-level solutions into a composite solution. The posterior probability densities from each task solution are combined to create a single posterior probability density for each variable. To arrive at the most probable solution, a histogram is created for each variable. The histogram bin with the highest count is determined and the midpoint of that bin is selected as the most probable value for that variable. Since we are using bin midpoints as the solutions, it is important to determine the correct bin width for the histogram of each variable. The optimal bin width, W, for an unbiased estimation of the probability density has been formulated by [19] and the calculation is shown in Eq. 8.

$$W = 3.49\sigma N^{-1/3}, \tag{8}$$

where σ is the sample standard deviation and N is the number of samples.

4 Application to FFT-07 Data Set

In 2007, the United States Army Test and Evaluation Command conducted the FUSION Field Trials of 2007 (FFT-07) [11]. Continuous and instantaneous releases of a propylene gas tracer were measured by a sensor network comprised of 100 sensors spaced 50 m

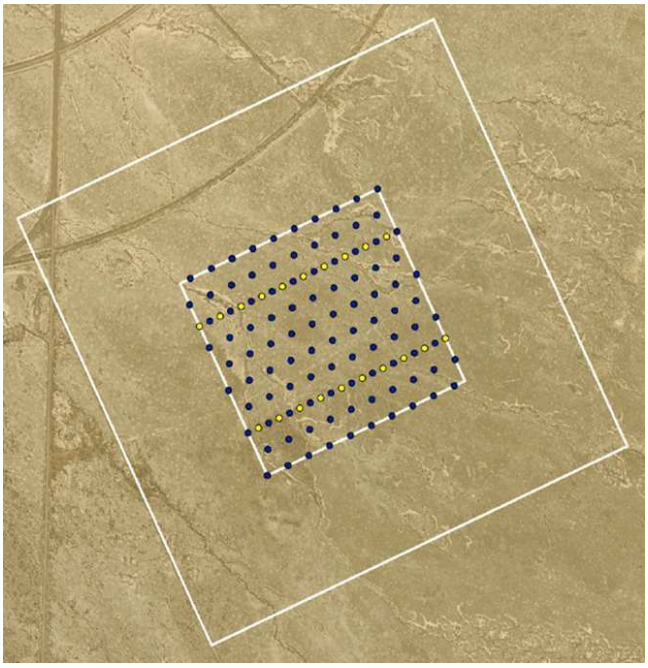

Fig. 6. Sensor layout for the FFT-07 data set [11] with the outer white box representing the 1 km $\times$ 1 km test space.

apart in a 1 km $\times$ 1 km test space. The sensor arrangement and test space was described in the FFT-07 report [11], and is shown in Fig.6.

In section 3.4, we referred to the composite solution created by combining the posterior probablity densities from the reconstruction at each time step. Figures 7, 8, 9, and 10 show these composite results in a "windowed" fashion, where each progression includes an additional 25% (30 s) of the available time step data. We view the results in this manner so we can see how the solution is affected as more data is included and how the solution changes the longer the contaminant puff is traveling in the sensor domain.

The darkest regions can be interpreted as the areas of highest probability. We note that throughout the solution, the true source location, indicated by a Red marker $(\times)$, is encompassed by a region of high probability. Using only the first 30 s of data, we see that the region of highest probability is within 50 m of the true source location. We also note in Fig. 7 that a bi-modal posterior distribution has developed. Upon further exploration this was determined to be caused by the puff splitting into two smaller puffs, each with its own central point of high concentration. The puff later merges back into one larger puff as it diffuses downwind and the contribution of this merger is shown by the larger single area of high probability produced in Fig. 10. Due to the steady wind assumptions of the Gaussian puff model, the longer the puff travels in the domain,

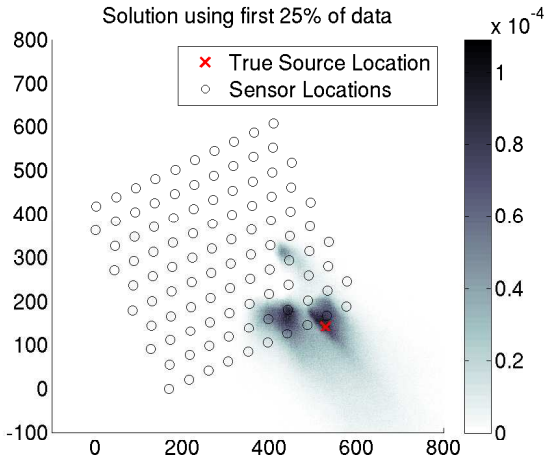

Fig. 7. Composite posterior probability density for source location using time 0 to 30 s.

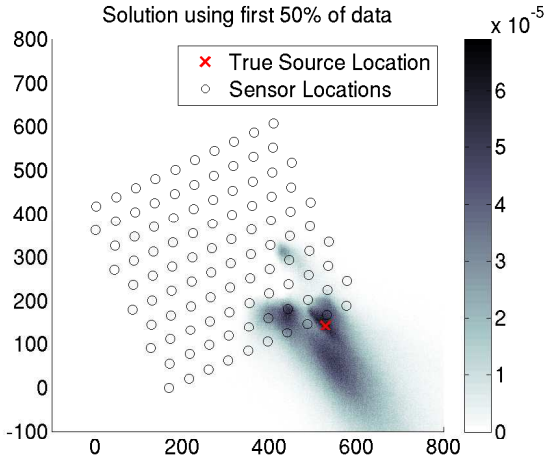

Fig. 8. Composite posterior probability density for source location using time 0 to 60 s.

unsteady conditions may cause the puff to deviate from a constant line of travel. We attribute some of the spread of the posterior distribution to the deviation of the wind from steady conditions.

Figures 11, 12, 13, and 14 show the windowed results for the estimate of time since release, or Δt. A value of zero corresponds to a correct estimate of the time since release. We can see that within the first minute of the plume entering the sensor domain, the agreement of release time is favorable. Beyond the first minute, however, we see that the results shift strongly away from the correct result. We believe that this is due to the same factor affecting the source location results, the deviation of the actual puff from the forward model formulation. From these results, it is clear that earlier data is much more beneficial when using a model that does not account for variations in wind conditions.

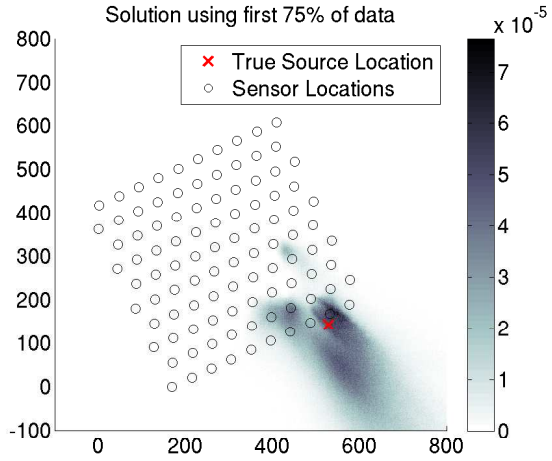

Fig. 9. Composite posterior probability density for source location using time 0 to 90 s.

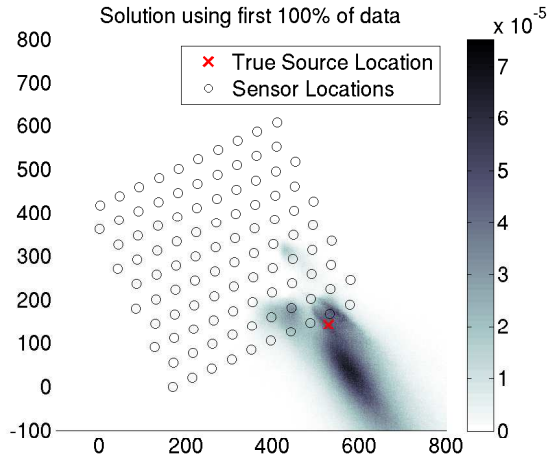

Fig. 10. Composite posterior probability density for source location using time 0 to 120 s.

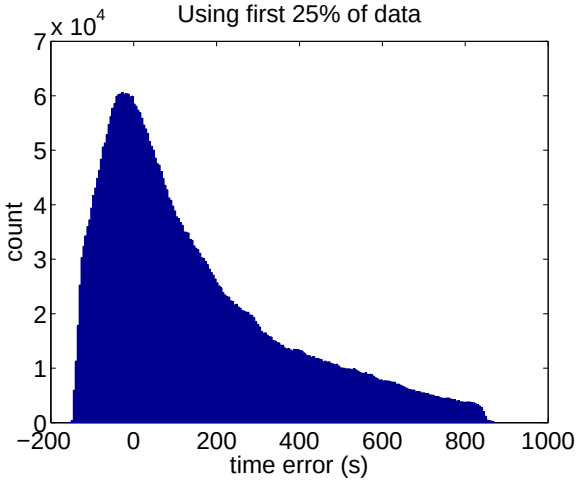

Fig. 11. Composite posterior probability density for time since release, Δt, using time 0 to 30 s.

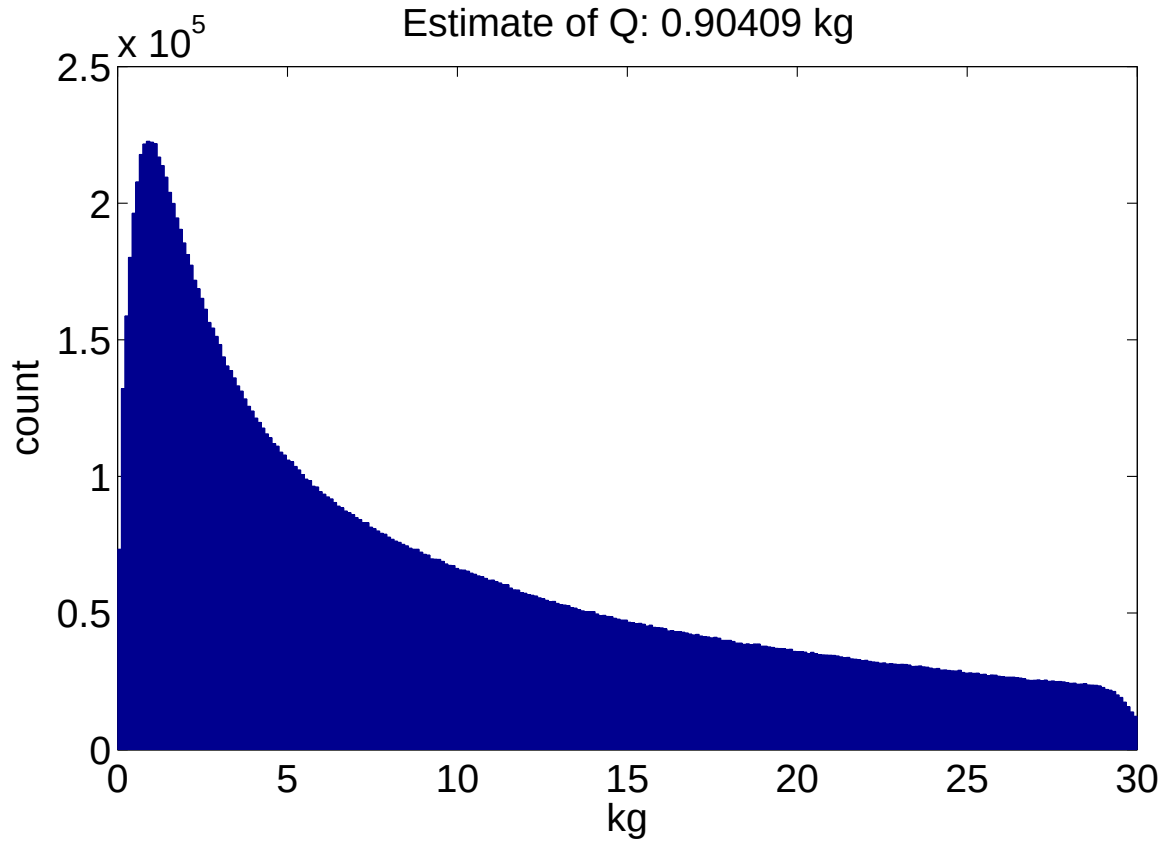

Fig. 15. Composite posterior probability density for contaminant release quantity, Q, in kilograms, using time 0 to 120 s.

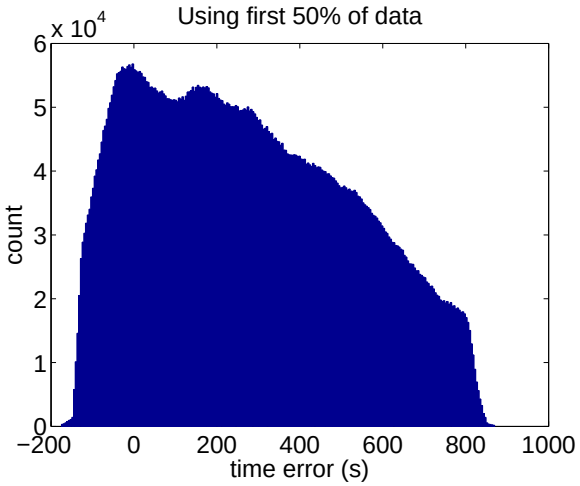

Fig. 12. Composite posterior probability density for time since release, Δt, using time 0 to 60 s.

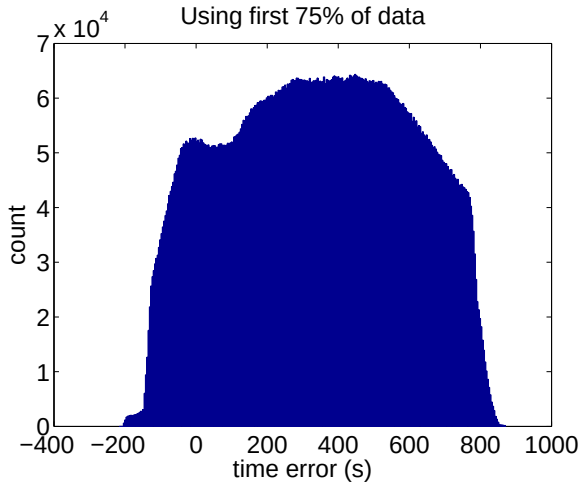

Fig. 13. Composite posterior probability density for time since release, Δt, using time 0 to 90 s.

One of the final goals of the event reconstruction was to determine the quantity of contaminant, Q, released into the atmosphere. Using the full two minutes of time during which the puff was present in the sensor domain, we estimated a release quantity of 0.9 kg. The true release quantity for Trial 5 was 1.21 kg. The posterior probability density for Q is shown in Fig. 15. The slight underestimation, approximately 25% error, is possibly due to the smoothing of the concentration measurements, which reduced the magnitudes of the peak concentrations.

5 Conclusion

The Puff Event Reconstruction Tool (PERT) was developed to reconstruct instantaneous contaminant re-

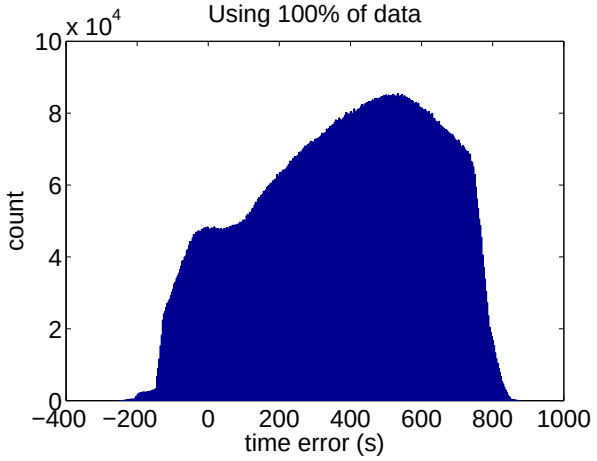

Fig. 14. Composite posterior probability density for time since release, Δt, using time 0 to 120 s.

leases. A trial from the FFT-07 data set has been reconstructed to test the performance of PERT. Using a workflow with multiple levels of parallelism, the tool quickly estimates characteristics of the source term. While there was some variation in the results due to the puff splitting and unsteady winds, the tool provided a probable estimate of source location approximately 100 m from the true source.

Using data collected soon after the release yielded better results, both spatially and temporally, than the results produced using data from the latter half of the trial. Overall, PERT was able to successfully characterize the source term, especially the release quantity parameter, which was very close to the true quantity. Windowed results were used to view the various stages of the composite solution. The windowed views showed how each period of data contributed to the overall solution.

To account for cases where the puff splits or encounters unsteady winds, a more complex puff model may be required to capture those elements of the contaminant dispersion. However, a more complex forward model may come at an increased computational cost, and therefore, an increased run time for PERT.

Acknowledgments

This work is supported by a grant from the U.S. National Science Foundation (Award #1043107).

References

[1] Senocak, I., Hengartner, N., Short, M., and Daniel, W., 2008. "Stochastic event reconstruction of atmospheric contaminant dispersion using bayesian inference". *Atmospheric Environment*, **42**, pp. 7718–7727.

[2] Wade, D., and Senocak, I., 2013. "Stochastic reconstruction of multiple source atmospheric contaminant dispersion events". *Atmospheric Environment*, **74**, pp. 45–51.

[3] Annunzio, A., Young, G., and Haupt, S., 2012. "A multi-entity field approximation to determine the source location of multiple atmospheric contaminant releases". *Atmospheric Environment*, **62**, pp. 593–604.

[4] Annunzio, A., Young, G., and Haupt, S., 2012. "Utilizing state estimation to determine the source location for a contaminant". *Atmospheric Environment*, **46**, pp. 580–589.

[5] Henze, D., Seinfeld, J., and Shindell, D., 2009. "Inverse modeling and mapping US air quality influences of inorganic $PM_{2.5}$ precursor emissions using the adjoint of GEOS-Chem". *Atmospheric Chemistry and Physics*, **9**, pp. 5877–5903.

[6] Allen, C. T., Young, G. S., and Haupt, S. E., 2007. "Improving pollutant source characterization by better estimating wind direction with a genetic algorithm". *Atmospheric Environment*, **41**, pp. 2283–2289.

[7] Akcelik, V., Biros, G., Draganescu, A., Hill, J., Ghattas, O., and Waanders, B., 2005. "Dynamic data-driven inversion for terascale simulations: Real-time identification of airborne contaminants". In SC '05 Proceedings of the 2005 ACM/IEEE conference on Supercomputing.

[8] Chow, F., Kosovic, B., and Chan, S., 2008. "Source inversion for contaminant plume dispersion in urban environments using building-resolving simulations". *Journal of Applied Meteorology and Climatology*, **47**, pp. 1553–1572.

[9] Keats, A., Yee, E., and Lien, F., 2007. "Bayesian inference for source determination with applications to a complex urban environment". *Atmospheric Environment*, **41**, pp. 465–479.

[10] Yee, E., 2008. "Theory for reconstruction of an unknown number of contaminant sources using probabilistic inference". *Boundary-Layer Meteorology*, **127**, pp. 359–394.

[11] Storwold Jr., D., 2007. Detailed test plan for the fusing sensor information from observing networks (FUSION) field trial 2007 (FFT-07). Tech. Rep. WDTC-TP-07-078, United States Army.

[12] Arya, S., 1999. *Air Pollution Meteorology and Dispersion*. Oxford University Press.

[13] Hanna, S., Briggs, G., and R.P. Hosker, J., 1982.

Handbook on Atmospheric Diffusion. Technical Information Center U.S. Department of Energy.

[14] Metropolis, N., Rosenbluth, A., Rosenbluth, M., and Teller, E., 1953. "Equations of state calculations by fast computing machines". *Journal of Chemical Physics,* **224**, pp. 560–586.

[15] Brooks, S., Gelman, A., Jones, G., and Meng, X., 2011. *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC.

[16] Gelman, A., and Rubin, D., 1992. "Inference from iterative simulation using multiple sequences". *Statistical Science,* **7**(4), pp. 457–511.

[17] R Core Team, 2012. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

[18] Plummer, M., Best, N., Cowles, K., and Vines, K., 2006. "Coda: Convergence diagnosis and output analysis for mcmc". *R News,* **6**(1), pp. 7–11.

[19] Scott, D., 1979. "On optimal and data-based histograms". *Biometrika,* **66**(3), pp. 605–610.