Efficient and Robust Localization of Multiple Radiation Sources in Complex Environments

Jren-Chit Chin* David K. Y. Yau*† Nageswara S. V. Rao‡
*Department of Computer Science, Purdue University, West Lafayette, IN 47907
†Advanced Digital Sciences Center, Illinois at Singapore
‡Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37830
Email: {jcchin, yau}@cs.purdue.edu, raons@ornl.gov

***Abstract*—We present a robust localization algorithm for multiple radiation sources using a network of sensors under random underlying physical processes and measurement errors. The proposed solution uses a hybrid formulation of particle filter and mean-shift techniques to achieve several important features that address major challenges faced by existing localization algorithms. First, our algorithm is able to maintain a constant number of estimation (source) parameters even as the number of radiation sources K increases. In existing algorithms, the number of estimation parameters is proportional to K and thus the algorithm complexity grows exponentially with K. Second, to decide the number of sources K, existing algorithms either require the information to be known in advance or rely on expensive statistical estimations that do not scale well with K. Instead, our algorithm efficiently learns the number of sources from the estimated source parameters. Third, when obstacles are present, our algorithm can exploit the obstacles to achieve better isolation between the source signatures, thereby increasing the localization accuracy in complex deployment environments. In contrast, incompletely specified obstacles will significantly degrade the accuracy of existing algorithms due to their unpredictable effects on the source signatures. We present extensive simulation results to demonstrate that our algorithm has robust performance in complex deployment environments, and its efficiency is scalable to many radiation sources in these environments.**

I. Introduction

There has been increasing interest in the detection and localization of radiation sources as part of the defense strategy against radiological dispersal devices (RDD) commonly known as dirty bombs. In particular, the higher concentration of people in urban areas increases the susceptibility to coordinated dirty bomb attacks. Harmful radioactive substances unleashed in such an attack could cause widespread health and environmental damage, since such substances would be hard to clean up and would cause adverse effects from long term exposure. The ability to quickly detect and localize illicit radiation sources is crucial. Such swift actions enable authorities to find and remove RDDs during transport or storage. Moreover, attackers may launch a coordinated attack involving multiple RDDs installed at different places. In such a scenario, we would need a system that can determine the number of the devices, as well as the strength and location of each device. The majority of current localization systems are designed for a single source, and are not effective for the multiple sources in a coordinated attack.

In dealing with multiple sources, existing localization approaches find the most probable location and strength, i.e. source parameters, of each source [1]–[3] by minimizing the error between modeled and actual measurements. The number of source parameters increases proportionally with the number of sources K, and thus the algorithm complexity grows exponentially with K. In addition, because the number of sources K is not known a priori, it needs to be estimated as well. Existing algorithms estimate the parameters of sources for a range of assumed values of K, i.e., $K = 1, 2, \ldots$ and use the selected K to compute the source parameters. This is inaccurate in large sensor networks due to: (i) a large range of possible K values but the lack of a priori knowledge about K; (ii) superposition of signal strengths from different sources; and (iii) large variance in sensor measurements caused by background radiation as well as interfering sources. These factors have been mostly addressed individually by the existing algorithms, but the models of these algorithms become significantly more complicated when the factors are jointly considered. For instance, a high measurement recorded by a sensor can be from a single strong source, or it may be induced by the combined strengths of multiple weak sources. Consequently, we may obtain equivocal results that appear equally valid for different values of K. Even if the estimate of K is accurate, multiple ambiguous solutions may exist due to the model dynamics, e.g., non-linear signal fading and a large number of interacting system parameters.

When obstacles may be present, building an accurate model can be infeasible as the shapes, sizes, materials, and locations of all the obstacles must be accurately known. These data are difficult to obtain, and incorporating them into a complete model is a non-trivial process. In this paper, we propose a hybrid formulation of particle filter and mean-shift techniques to address the above-mentioned challenges faced by existing algorithms. In our formulation, we first generate hypotheses about the source parameters (known as *particles*). When a measurement is received, the likelihoods of these particles being one of the sources are evaluated. Unlike a standard particle filter formulation, we *selectively* evaluate these particles based on the location of the sensor providing the reading.

Research supported in part by Mathematics of Complex, Distributed, Interconnected Systems Program, Office of Advanced Computing Research, U.S. Department of Energy under grant number DE-AC05-00OR22725; and in part by U.S. National Science Foundation under grant number CNS-0964086.

The selective evaluation is a major feature in our approach that enables localization of multiple sources efficiently in a complex environment. To compute the source parameters, the mean-shift technique is used on the particles to determine the most likely parameters of *all* the sources. Because the mean-shift technique determines the parameters of all the sources without knowledge of the number of sources K, we eliminate the expensive statistical test to estimate K.

Our main contributions are as follows:

- We propose a novel localization approach for multiple radiation sources in complex environments that (i) is efficient in handling multiple sources; (ii) does not require the number of sources to be known in advance, or require its expensive statistical estimation; and (iii) does not require detailed specifications of obstacles in these environments.
- We introduce the notion of *fusion range* for use in the particle filter. This allows our algorithm to handle multiple sources without explicitly modeling all the sources. Subsequently, our algorithm has efficiency that scales with a large number of sources.
- We provide quantitative results to show that the proposed algorithm can accurately localize radiation sources and achieve low false positive and false negative rates.

The balance of the paper is organized as follows. We discuss related work on source localization in Section II. In Section III, we formulate the problem under a realistic environment. We provide necessary background on Bayesian estimation and particle filter in Section IV. Then, we present the proposed localization algorithm in Section V. Performance results are provided in Section VI, Section VII concludes.

II. Related Work

The detection, localization, and tracking of signal sources (or *targets*) of various kinds, such as radiation sources, chemical plumes, wideband radio signals, and acoustic signals, have been well studied [1], [3]–[10].

For radiation sources, the localization of both single and multiple sources has been considered. Typically, the single source localization problem is solved using least square methods or maximum likelihood estimation (MLE) methods, which search for the most probable source parameters fitting the sensor measurements [11], [12]. Rao *et al.* [4] adapt the TDOA algorithm [13] to log-space and exploit the log-differences in source strength measurements from three sensors to infer the source location. To tolerate noise, Rao *et al.* [14] propose the mean-of-estimator (MoE) method which first uses TDOA to localize the source by all subsets of three sensors and then linearly combines the localization results to give the final answer. A similar fusion approach is presented in [5], where sensor data are fused using the iterative pruning (ITP) algorithm. These single source localization algorithms are not applicable when there are multiple sources.

In localizing multiple sources, existing algorithms need to estimate the number of sources K in addition to the source parameters. For instance, Ding *et al.* [15] localize multiple signal sources by modeling the targets with a Gaussian mixture model and then using Akaike's Information Criterion (AIC) or Bayesian Information Criterion (BIC) to estimate K. Then, expectation maximization (EM) and mean-shift algorithms are used to localize the K sources. Their source model is generic, and application to real-world radiation source models is not discussed. In [1], [2], the authors first estimate the number of sources using a similar model selection algorithm, and then compute the source parameters using the MLE method. As reported in [2], the accuracy of such model selection degrades when the number of sources increases. In addition, using EM or MLE to estimate source parameters is computationally expensive when the model has many parameters, where each additional source increases the number of parameters in the model by three. As reported in [2], the algorithms do not scale beyond four sources. In [16], the authors propose to solve the multiple-source localization problem using convex optimization assuming that the sources are located in a grid over the region of interest. The proposed method discretizes the search space and attempts to localize a source in each discretized location in the space. Depending on the granularity of the discretization, the algorithm can take up to 209 s to run for a scenario with 196 sensors on a dual-core Intel Pentium CPU at 2.40 GHz with 4 GB RAM, as reported in the paper. This prohibits the algorithm from scaling to a large network of sensors.

Localizing targets in environments with obstacles has also been considered. In [17], the algorithm solves the localization problem assuming that the placement of obstacles is known. The algorithm discretizes the search space and finds the probability p of a target being located inside each cell. Because the location is known, p can be easily calculated with considerations of the obstacle. Nonetheless, the algorithm has complexity exponential in the number of sensors. In [18], the proposed method uses mobile robots to search for a radiation point source. The authors focus on planning the robot movements such that each move maximizes the information gain in the search. The detection and localization of the source during the search is performed using a particle filter. Other efforts have mainly studied how obstacles may affect sensor communication and motion planning rather than how they may affect the source localization algorithms. For instance, the problem of tracking a target using mobile robots, while detecting and avoiding obstacles that may block the target signals, has been studied in [19], [20].

III. Problem Formulation

We consider the localization of K radiation point sources of unknown strengths using a network of radiation sensors in a two-dimensional surveillance area with obstacles. Let $\mathcal{A} = \{\mathbf{A}_1, \ldots, \mathbf{A}_K\}$ denote the set of radiation sources. Each radiation source is parameterized by a three-value vector $\mathbf{A}_j = \langle A_j^{\mathrm{x}}, A_j^{\mathrm{y}}, A_j^{\mathrm{str}} \rangle$, for $1 \leq j \leq K$. The position of the source is given by its xy- coordinates (in cm) and the strength of the source is a positive real number in micro-Curies (μCi). For convenience, we define $A_j^{\mathrm{pos}} = \left(A_j^{\mathrm{x}}, A_j^{\mathrm{y}}\right)$.

In a surveillance area without obstacles, the strength of $\mathbf{A}$ measured at location $\mathbf{x}$ is

$$\mathcal{I}_{FS}(\mathbf{x}, \mathbf{A}) = A^{\text{str}} \left(1 + |\mathbf{x} - A^{\text{pos}}|^2\right)^{-1}. \quad (1)$$

The above model has been widely used in existing work [2], [4], [12], [18] and has been verified experimentally in [5].

The surveillance area may contain a number of obstacles that shield the radiation rays. The obstacles can be of any types of materials, of any shape, and located anywhere in the surveillance area unknown to the system. The material of the obstacle determines its effectiveness in absorbing the radiation, i.e., materials of higher atomic numbers and higher densities are more effective in absorbing gamma radiation. In addition, the amount of gamma rays absorbed in the shielding material is proportional to the thickness of the material. For instance, 1 cm thick of lead can absorb as much radiation[1] as 6 cm thick of concrete [21]. For simplicity and without loss of generality, we assume that each obstacle is made of a homogeneous material. Assuming that the radiation source is on one side of the obstacle, we can model the intensity of the source on the other side of the obstacle by

$$\mathcal{I}_S(l, \mathbf{A}) = A^{\text{str}} e^{-\mu l}, \quad (2)$$

where l denotes the thickness of the obstacle and μ denotes the attenuation coefficient for the material of the obstacle. The effects of obstacles on gamma radiation have been extensively studied and the values of μ are published in [21].

In a realistic setup, gamma rays from a radiation source may traverse through both free space and obstacles. The intensity at some location $\mathbf{x}$, considering both free space and obstacles, can be obtained by combining Eq. (1) and (2),

$$\mathcal{I}(\mathbf{x}, \mathbf{A}) = \frac{A^{\text{str}}}{1+|\mathbf{x}-A^{\text{pos}}|^2} \exp\left(\sum_{b \in (\mathcal{B} \cap \overline{\mathbf{xA}})} -\mu_b l_b\right), \quad (3)$$

where $\mathcal{B}$ denotes a set of obstacles, $\mathcal{B} \cap \overline{\mathbf{xA}}$ denotes the set of obstacles that intersect with the straight line between $\mathbf{x}$ and A^{pos}, l_i denotes the total thickness of the obstacle b along $\overline{\mathbf{xA}}$, and μ_b denotes the attenuation coefficient of obstacle b. This model has been used in existing work as well [22].

Radiation sensors are placed at known locations in the surveillance area. Radiation sensors measure the radiation intensity by counting the number of ionizations occurring due to gamma radiation in a fixed interval. The intensity is usually reported in counts per minute (CPM). Let S_i, for $1 \leq i \leq N$, denote the location of sensor i. The sensor i will record the total intensity from all the radiation sources. In addition, each sensor i will record a background radiation B_i (in CPM) that is universally present due to cosmic rays and the decay of naturally occurring radio-isotopes such as Carbon-14. and Potassium-40. Each sensor has a different efficiency in counting the number of ionizations due to different manufacturing technologies and sizes of the sensors, which causes bias in the measurements. The efficiency constant E_i in our model allows to correct this bias. The value of E_i can be obtained through a calibration procedure described in [5]. Given all of the above, the expected intensity (in CPM) at S_i is modeled by

$$I_i = 2.22 \times 10^6 \times E_i \sum_{j=1}^{K} \mathcal{I}(S_i, \mathbf{A}_j) + B_i. \quad (4)$$

The constant 2.22×10^6 is the conversion factor from micro-Curie to CPM. Given the expected intensity I_i, the measurements by sensor i, $m(S_i)$, are modeled as a Poisson process with average rate $\lambda = I_i$.

[1]Gamma ray with energy 1 MeV.

IV. Recursive Bayesian Estimation and Particle Filter

Recursive Bayesian estimation is widely used to estimate a set of variables (called *state variables*), given a set of observations. The objective is to find the best estimate of the state variables considering all the observations available, and refine the estimates as more observations become available.

In the formulation, the state x_t at iteration t is predicted using all information available prior to t, D_t, by computing the prior PDF

$$\mathrm{P}(x_t | D_{t-1}) = \int \mathrm{P}(x_t | x_{t-1}) \mathrm{P}(x_{t-1} | D_{t-1}) \, dx_{t-1},$$

where $D_t = \{y_i \mid i = 1 \ldots t\}$, and y_i denotes the observation at iteration i. Then, as new information is acquired at t, the posterior PDF is computed via Bayes rule as

$$\mathrm{P}(x_t | D_t) = \frac{\mathrm{P}(y_t | x_t) \mathrm{P}(x_t | D_{t-1})}{\int \mathrm{P}(y_t | x) \mathrm{P}(x | D_{t-1}) \, dx}.$$

The estimated state $\hat{x}_t$ is then computed by finding the expectation of $\mathrm{P}(x_t | D_t)$.

The above formulation is general and could be applied to any state estimation problem. With linear models and Gaussian noise of known variance, an analytical solution of the above can be found. In many cases, however, analytical solutions cannot be found and thus the procedure has practical limitations.

Particle filter addresses the Bayesian estimation by using random samples (called *particles*) to approximate the PDF in the estimation. When the number of samples becomes very large, the samples can be proved to provide an exact representation of the required PDF. A main feature of particle filter is that it can be applied to non-linear, non-Gaussian problems even if there is no general analytic expression for the required (PDF.

Particle filter works in two phases: *predict* and *update*. The *predict* phase approximates the prior PDF by evolving particles from $t-1$. Then, the *update* phase approximates the posterior PDF by computing the probability that a particle represents the current state, for each particle. The state estimate is then computed as a weighted sum of all the particles. The process is repeated when the next measurement arrives.

In a straightforward application of particle filter in radiation source localization, one would consider the xy-coordinates and the source strength to be the state variables. The intensity measurements then drive the predict and update phases of the

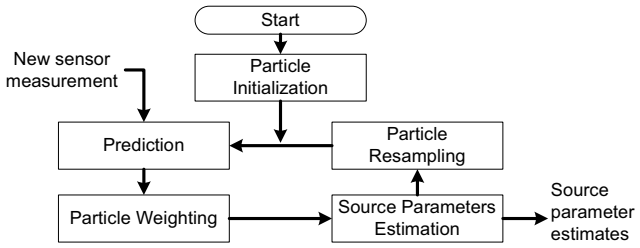


Fig. 1. Overview of the proposed algorithm.

particle filter. Specifically, for each particle, the expected radiation intensity at each sensor is computed according to Eq. (4). Then, the probability of obtaining the particular measurement is assigned as the weight of the particle. Over time, the expectation of all the particles will give an accurate estimate of the source parameters. This simple approach performs well in scenarios with only one source. With multiple sources, however, the number of state variables increases proportionally to the number of sources. This causes the parameter space to increase exponentially. Consequently, the number of particles needs to be increased exponentially so as to maintain a representative PDF for the estimation. As a result, this approach does not scale to a large number of sources. Furthermore, this approach requires the number of sources to be known ahead of time, which limits its applicability in many realistic situations.

Another problem with the above approach, or with any existing radiation localization algorithm, is that the detailed parameters of obstacles—shapes, locations, and attenuation coefficients—are assumed to be known, so that the model in Section III can be applied. Incorporating such data into the localization system is a burden to the user. Moreover, the attenuation coefficients of the obstacles need to be measured on-site unless accurate blueprints of the deployment buildings are readily available. Without the complete data, the expected intensities from the sources cannot be computed because radiation sources can induce a wide range of readings on radiation sensors depending on the quantity and nature of obstacles located between the sources and the sensors.

V. Algorithm Design

In this section, we describe our algorithm for localizing multiple sources in a complex environment (i.e., with the presence of unknown or incompletely known obstacles) using source strength measurements by a network of sensors. The proposed algorithm recursively refines the location and strength estimates of all the sources based on newly acquired measurements and all prior information. Fig. 1 outlines the flow of the algorithm. The algorithm starts by initializing a collection of particles, each of which hypothesizes the location and strength of a source. In each iteration, as a new sensor measurement becomes available, the algorithm predicts the locations and strengths of the sources based on the previous estimates. Then, the weights are updated according to the newly received sensor measurement and the prior weights. After weighting the particles, the source parameter estimates are computed. Finally, a resampling procedure normalizes the weights of the particles. The procedure then repeats as a new sensor measurement arrives.

The highlights of our formulation are as follows:

- We only consider one measurement in each iteration instead of all the measurements as in a typical formulation. Furthermore, there is no ordering on the measurements. This allows the algorithm to proceed as soon as possible, without waiting for all the measurements. This improves the robustness of the algorithm, and is particularly relevant in wireless sensor networks because wireless transmissions are unreliable due to environmental factors, low transmission power, and malfunctioning of unreliable sensors. Moreover, the network latency is usually high due to multi-hop wireless forwarding and signal interference among a large number of communicating sensors.
- Instead of explicitly modeling each radiation source, which is typically done [2], we simply compute the parameter estimates without attributing them to specific sources. When multiple estimates resolve to the same location parameters, we classify them as belonging to the same source, and multiple sources are distinguished when their estimates resolve to different locations.
- We introduce the notion of *fusion range* for use in the particle filter, in which particles are selectively updated based on the locations of the sensors. The fusion range mitigates interference between particles when there are multiple sources.
- We apply a mean-shift technique to compute the source parameters. Using mean-shift is crucial in order to recover the parameters of multiple sources because our formulation does not explicitly model the individual sources. This is an important departure from traditional particle filter. Details of integrating mean-shift into our particle filter formulation are described in Section V-D.

Combining the above, our algorithm solves the multiple radiation source localization problem without (i) knowing the number of sources, (ii) increasing the complexity of the parameter space as the number of sources increases, and (iii) knowing all the obstacles in details. The detailed algorithm steps are described in the following sub-sections.

A. Particle initialization

We define the initial iteration to be $t = 0$. At this time, the particle filter is initialized as follows. Let $\mathcal{P} = \{\mathbf{p}_i^{(t_i)} \mid i = 1, ...N_{\mathcal{P}}\}$ be a set of particles in a surveillance area. Each particle $\mathbf{p}_i^{(t_i)}$ is a three-value vector in the parameter space $\mathbb{A}$ denoting the hypothesized source position and strength. The superscript t_i is an integer that denotes the iteration at which the particle is updated. We will drop the superscript where there is no ambiguity in the iteration number. We initialize $\mathcal{P}$ with uniformly random particles. The uniformly random distribution is used because we do not assume any a priori knowledge about the location or strength of the source. If prior knowledge is available, the particles can be initialized according to the pre-existing distribution.

Doing so will reduce the number of iterations required to obtain accurate estimates of the sources.

The number of particles in $\mathcal{P}$, given by $|\mathcal{P}| = N_{\mathcal{P}}$, affects the coverage of $\mathcal{P}$ in $\mathbb{A}$. A larger coverage will result in a more accurate estimate in a shorter time. We associate each particle $\mathbf{p} \in \mathcal{P}$ with a weight $w(\mathbf{p})$ such that $w(\mathbf{p}) \approx \sum_j \mathrm{P}(\mathbf{A}_j = \mathbf{p})$ and $\sum_{\mathcal{P}} \mathbf{p}_i = 1$. As such, this weight measures the probability that an actual radiation source is located at $\mathbf{p}$. It is updated as a new sensor reading becomes available. However, as the actual source position is unknown, $\sum_j \mathrm{P}(\mathbf{A}_j = \mathbf{p})$ cannot be computed. In Section V-C, we will discuss in details how we can approximate this probability using sensor measurements.

B. Prediction

After initializing the particles, the localization process can begin. Our algorithm refines the source estimates whenever there is a new sensor reading. At iteration t, a sensor located at S_i delivers a measurement $m(S_i)$. Using this measurement, we construct a set $\mathcal{P}'$ such that all the particles in $\mathcal{P}'$ are less than d_i distance away from S_i:

$$\mathcal{P}' = \left\{ \mathbf{p}_i \;\middle|\; \|S_i - p_i^{\mathrm{pos}}\|^2 \leq d_i^2 \right\}. \tag{5}$$

This distance d_i is defined as the *fusion range* and it is specific to S_i. The value of d_i is selected such that a particle located at p is within the fusion range of a handful of sensors. The fusion range controls the affected particles in the current iteration. It prevents the current update from affecting particles that are far from the sensor providing this reading. Even in open space, sources that are far away will not make significant contributions to the sensor's readings. By limiting the affected particles, we would not use the particles to distinguish between different sources explicitly. Thus, the number of estimation parameters, and thus the complexity of our algorithm, remains constant regardless of the number of sources.

Let us illustrate what would happen without applying the fusion range. In this case, all the particles will be affected in each iteration as with a typical particle filter formulation. Fig. 2 illustrates the behavior of the algorithm without the fusion range. When multiple sources exist, the particles will gravitate towards a source when the sensors near the source send updates. In Fig. 2(a), the particles were previously concentrated near source B. At this iteration, the sensors send measurement updates one after another. Specifically, after the iteration $t = 108$, the sensors at (0,0), ..., (0, 100), (20, 0), ... send updates at $t = 109, \ldots, 114, 115, \ldots$ respectively. In Fig. 2(b), the particles start to move away from source B. After the update from the sensor at (20, 100), the particles are concentrated near source A as shown in Fig. 2(c). When the sensors near source B update, the particles move away from source A as shown in Fig. 2(d).

Given $\mathcal{P}'$ which was updated prior to t, we could in general predict the location of the source at t using some movement model $\mathcal{F}_{\mathrm{movement}} : \mathbb{A} \to \mathbb{A}$, and produce the updated set $\mathcal{P}''$. In this paper, we assume the sources to be static; therefore $\mathcal{P}'' = \mathcal{P}'$.

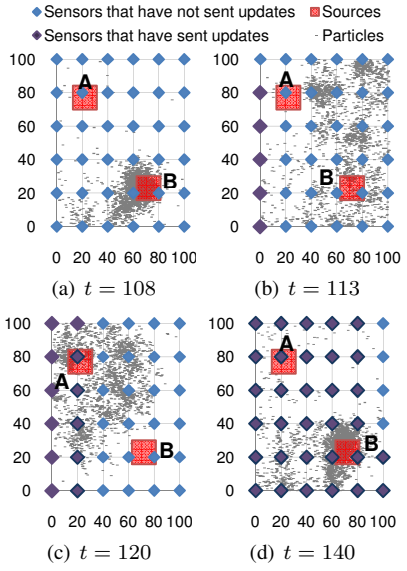


Fig. 2. Progression of particles filter without fusion range.

C. Particle weighting

After the previous step, we compute the weight of each particle, which is the likelihood that an actual radiation source has the same parameters as the particle. Although we do not know the actual source parameters, we can estimate this probability using the sensor measurements. Assuming that the radiation source is at $\mathbf{p}$, we can compute the expected sensor reading at S_i using Eq. (4). The probability of obtaining $m(S_i)$ given $\mathbf{p}$ follows the Poisson distribution. Subsequently, we update the weight of each particle in $\mathcal{P}''$ such that

$$w\left(\mathbf{p}_i^{(t)}\right) = \mathrm{P}\left(m(S_i) \;\middle|\; \mathbf{p}_i^{(t)}\right) w\left(\mathbf{p}_i^{(t')}\right).$$

After updating the weights, we merge $\mathcal{P}''$ with $\mathcal{P}$ to form a global view of all the particles. Let $\mathcal{P}^{\mathrm{new}}$ denote the merged set. $\mathcal{P}^{\mathrm{new}}$ is defined as $\mathcal{P}^{\mathrm{new}} = (\mathcal{P} - \mathcal{P}') \bigcup \mathcal{P}''$. The weights of all the particles are normalized again after the merging process such that $\sum_{\mathcal{P}^{\mathrm{new}}} w(\mathbf{p}_i) = 1$.

The merging process plays an important role in enabling the particle filter to localize multiple sources without increasing the number of estimation variables. A typical particle filter formulation updates all the particles in every iteration. Each sensor measurement affects all the particles including particles that are far away from the sensor which are not relevant to the current measurement, or particles estimating mainly other radiation sources. Furthermore, the sensor measurements close to those particles (which will be received in the future) will provide a better view at those locations. By updating particles that are close to the sensor providing the measurement only, we allow multiple sources to co-exist in the area, and yet do not require separate variables for different sources. Equally importantly, this allows our algorithm to localize multiple sources without knowing the number of sources.

D. Source parameter estimation

Given the updated $\mathcal{P}^{\mathrm{new}}$ and associated weights, we now compute the position and strength estimates of the radiation

sources. Our step here is very different from the traditional particle-filter approach where the estimate is computed as a weighted sum of all the particles. In our case, if the weighted sum of all the particles were computed, we would not be able to differentiate the multiple sources. Instead, we will obtain the centroid of the sources.

From this point onwards, the old $\mathcal{P}$ is no longer used. We will refer to $\mathcal{P}^{\text{new}}$ as $\mathcal{P}$. To obtain the estimates for individual sources, we construct the probability distribution function

$$L_{\mathcal{P}}(\mathbf{x}) = \mathrm{P}(\mathbf{x}|\mathcal{P}) = \left(\sum_{\mathcal{P}} w(\mathbf{p}_i)\right)^{-1} \sum_{\mathcal{P}} w(\mathbf{p}_i)\, \phi_{\mathbf{H}}(\mathbf{x} - \mathbf{p}_i).$$

In the above, $\phi_{\mathbf{H}}$ is the Gaussian kernel function

$$\phi_{\mathbf{H}}(\mathbf{x}) = (2\pi)^{-3/2} |\mathbf{H}|^{-1/2} \exp\left(-\tfrac{1}{2}\mathbf{x}^{\top}\mathbf{H}^{-1}\mathbf{x}\right) \quad (6)$$

and $\mathbf{H}$ is a symmetric positive definite matrix denoting the kernel bandwidth [23]. In theory, the function $L_{\mathcal{P}}(\mathbf{x})$ is a mixture distribution, where the number of mixture components represents the number of sources. Estimating the source parameters, therefore, is equivalent to finding the distribution parameters of each mixture component. This is accomplished by finding the values of $\mathbf{x}^*$ that maximize $L_{\mathcal{P}}$, i.e. $\nabla L_{\mathcal{P}}(\mathbf{x}^*) = \mathbf{0}$.

Note that there will be multiple solutions (i.e., local optima) when there are multiple sources. We are interested in finding all the solutions each of which corresponds to the estimate of one true source. This can be efficiently accomplished by using the mean-shift algorithm [23]. Using mean-shift, we find $\mathbf{x}^*$ by repeatedly applying $\mathbf{x}_{i+1} = M(\mathbf{x}_i)$ for $i = 1, 2, \ldots$ until convergence, where $M(\mathbf{x})$ is the mean-shift term derived from $\nabla L_{\mathcal{P}}(\mathbf{x})$ and given by

$$M(\mathbf{x}) = \frac{\sum_{\mathcal{P}} \mathbf{p}_i w(\mathbf{p}_i)\, \phi_{\mathbf{H}}(\mathbf{x} - \mathbf{p}_i)}{\sum_{\mathcal{P}} w(\mathbf{p}_i)\, \phi_{\mathbf{H}}(\mathbf{x} - \mathbf{p}_i)}. \quad (7)$$

Given an arbitrarily selected initial value of $\mathbf{x}_1$, the mean-shift algorithm will converge to the closest $\mathbf{x}^*$. To find all the radiation sources, we repeat the algorithm with different values of $\mathbf{x}_1$ and merge all the results that converge to the same $\mathbf{x}^*$.

The advantages of using the mean-shift algorithm in estimating the source parameters are: (i) the algorithm does not require knowledge of the number of sources when combined with our particle filter formulation, and (ii) the algorithm is robust under noisy sensor data.

E. Particle resampling

The last step of our algorithm, known as *resampling* in the literature, replaces particles of low weights with particles of higher weights. Resampling is important because it prevents the particle filter from degenerating. In a particle filter without resampling, all the particles will have decreasing weights over time, except for the one closest to the source. Eventually, all but one particles will have zero weight. The estimates will not be refined further due to the lack of diversity in the particles.

As an example, consider a scenario with a single static radiation source. We initialize $N_{\mathcal{P}}$ particles randomly in the space $\mathbb{A}$. Without resampling, in each iteration, the particles close to the radiation source will have increasingly larger weights whereas the particles far away from the radiation source will have increasingly lower weights. Over time, all the particles except the particle closest to the radiation source will have zero weight. In this case, the particle filter degenerates to a single particle. The parameter estimation will not be refined any further due to the lack of diversity in the particles. The resampling step essentially solves this problem by removing particles of low weights and multiplying particles of high weights. In our algorithm, resampling is only performed on the particles that are affected in the current iteration, which are in $\mathcal{P}''$. Particles that are not affected in the current iteration are not resampled because they were previously resampled and are unmodified since.

Resampling is accomplished by sampling with replacement from $\mathcal{P}''$ with probabilities $\frac{w(\mathbf{p}_i)}{\sum_{\mathcal{P}} w(\mathbf{p}_j)}$ for all $\mathbf{p}_i \in \mathcal{P}''$. The resampled particles are assigned uniform weights. During resampling, when the same particle is duplicated, we introduce zero-mean Gaussian noise into the duplicated particles [24]. The standard deviation of the noise is a tunable parameter given by σ_N. This prevents multiple particles from having the same parameters, which will eventually collapse to a single point. In our algorithm, resampling is only performed on the particles that are affected in the current iteration, which are in $\mathcal{P}''$. Particles that are not affected in the current iteration are not resampled because they were previously resampled and are unmodified since.

The above resampling procedure eliminates particles that do not correspond to any actual sources. As time proceeds, areas with no radiation sources initially will have few or no particles. If a new radiation source moves to these areas, they will be undetected because Eq. (5) produces a null set. As a provision for new radiation sources entering the area, we randomly replace a small percentage of particles, e.g., 5%, with random particles. This ensures that the new radiation sources will be detected and localized quickly.

VI. Evaluation

We conducted experiments to evaluate the performance of our source localization algorithm. We measured the performance of the algorithm by (i) localization error, (ii) number of false positives, and (iii) number of false negatives. In measuring the localization accuracy, the Euclidean distance between the actual source position and the closest estimate is used. However, each estimate must estimate a single source only. If no estimate is within 40 units from an actual source, the source is considered a false negative. The estimates that cannot be traced to any actual source are considered false positives.

We have simulated a 100×100 surveillance area with $N = 6 \times 6$ sensors placed in a uniform grid covering the entire surveillance area. All the sensors receive a background radiation of 0, 5, 10, or 50 CPM. These values are chosen to match typical environmental background radiation, which is about 5–20 CPM. Several radiation sources of strengths 4–1000 μCi are randomly placed in the surveillance area. These source strengths correspond to typical dirty bombs that could be used in an actual terrorist attack [5]. To put these

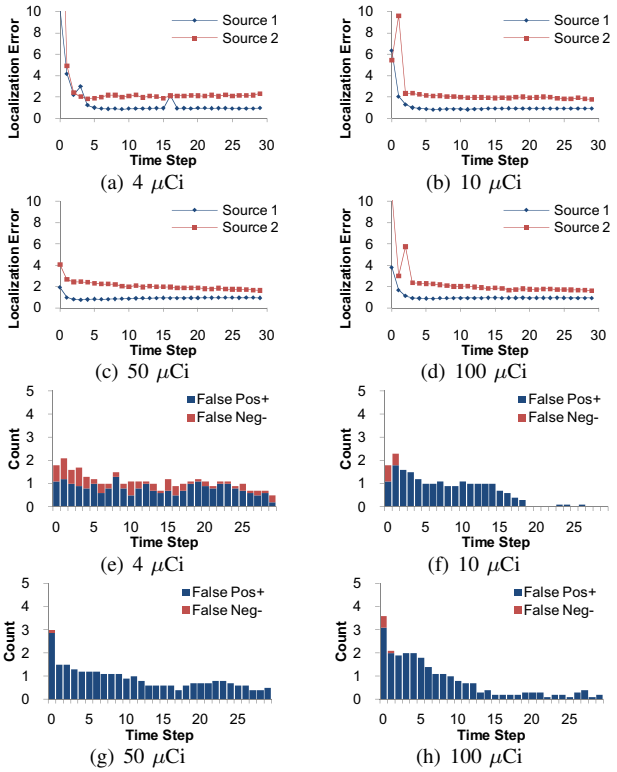


Fig. 3. Localization error (a–d) and false positives/negatives (e–h) with two sources of various strengths (without obstacles). Two sources are located at (47, 71) and (81, 42). Background radiation is 5 CPM.

source strengths further in perspective, the radiation source in a radiotherapy machine is more than 1000 Curies strong [25]. Each simulation is repeated 10 times and the average results are reported. In all the simulations, the standard deviation of resampling noise σ_N is set to 3.0, and the fusion range is $d_i = 28$ for $1 \leq i \leq N$. We use the same fusion range for all the sensors because they are arranged in a uniform grid.

In our simulations, sensor measurements arrive sequentially and in order. Here, we introduce the notion of time step T. In each time step, each sensor in the surveillance area submits one measurement update. Therefore, each time step is equivalent to N iterations. All the simulations reported have 30 time steps.

A. *Multiple source results*

We report an experiment with two sources of various strengths located at (47, 71) and (81, 42). The performance of the algorithm is shown in Fig. 3. In the figure, the large localization error in the first few time steps arises because we initialize the particles uniformly randomly without additional knowledge. As a result, the algorithm does not have enough information to accurately localize the sources. As shown in Fig. 4, particles start to cluster at the sources as early as $T = 1$. As more sensor measurements become available, the localization error quickly reduces to a small value.

Because the particles are randomly located at $T = 0$, the algorithm may produce many false positives because many places are thought likely to contain the source given the limited number of measurements. As time proceeds, false positives are reduced because the sensor measurements do not give high

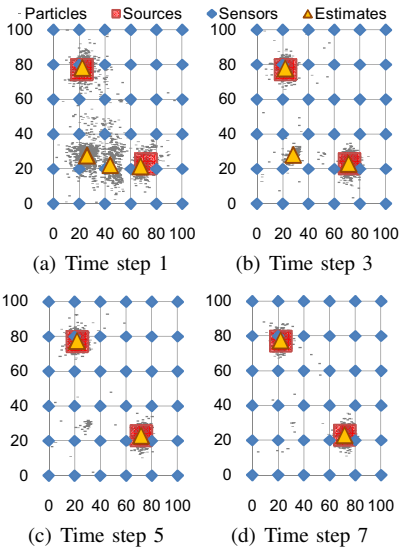


Fig. 4. Progression of particle filter over time.

weights to their particles. In addition, there is an increase in the number of false positives for stronger sources because the radiation from strong sources can reach a long distance. This increases the ambiguity in the sensor measurements because the reading could be due to a strong source located far away or a weaker source located closer to the sensor. Thus, the algorithm is unable to reduce the weights of particles far away from the actual source, which produces false positives. The problem can be mitigated by reducing the fusion range. However, without knowing the source strengths in advance, reducing the fusion range can increase the false negatives because weaker sources are missed. The false negatives, on the other hand, are close to zero except when the source strength is very weak (e.g., 4 μCi) as shown in Fig. 3(e). This is because the radiation from weak sources appears to be similar to background radiation. This makes it hard for the algorithm to distinguish between radiation from a source and that from the background.

We evaluated the algorithm in a scenario with three sources. The three sources are located at (87, 89), (37, 14), (55, 51), and the background radiation is 5 CPM. The results in a three-source simulation scenario are similar to the two-source case as shown in Fig. 5. However, the algorithm requires more time steps to produce accurate results. Especially with the 4 μCi source, the algorithm takes 9 time steps before producing accurate location estimates. In terms of false positives and negatives, similar trends are observed as the two-source scenario.

We have also evaluated our algorithm with four different levels of background radiation: 0, 5, 10, and 50 CPM. The results in Fig. 6 show that our algorithm can tolerate above-average background radiation than in typical environments. Higher background radiation only affects the first few time steps of the algorithm. The results also show no impact on the false positives and negatives by the higher background radiation.

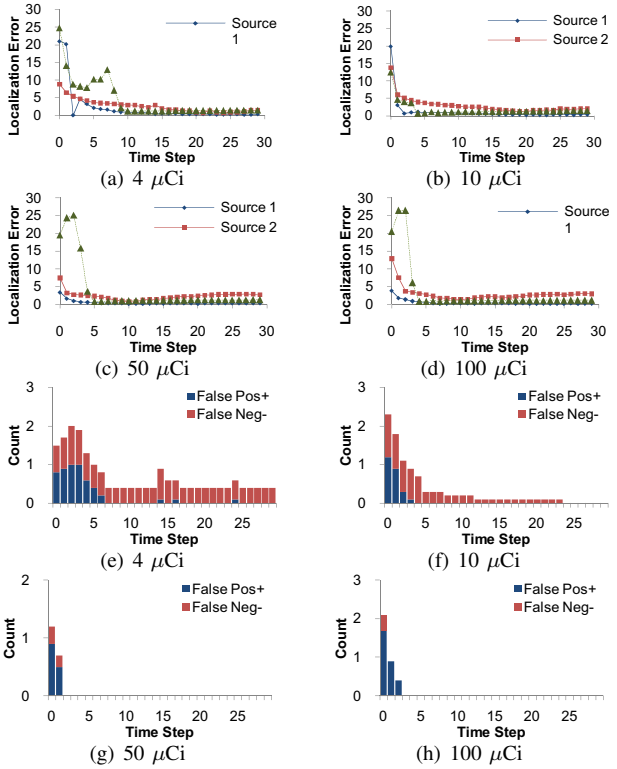


Fig. 5. Localization error (a–d) and false positives/negatives (e–h) with three sources of various strengths (without obstacles). The sources are located at (87, 89), (37, 14), (55, 51). Background radiation is 5 CPM.

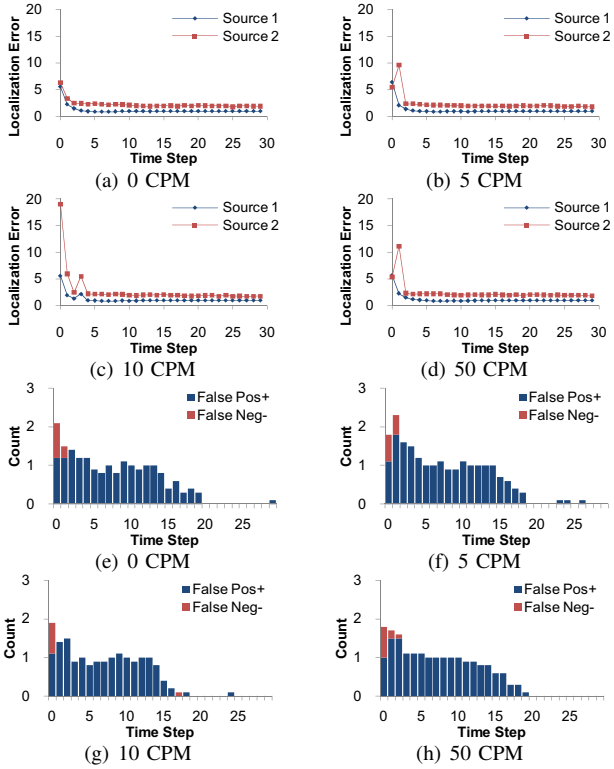


Fig. 6. Localization error (a–d) and false positives/negatives (e-h) of two sources under various background radiation levels (without obstacles). Two sources are located at (47, 71) and (81, 42). Source strengths are 10 μCi.

B. Results with obstacles

To evaluate how obstacles affect the performance of our algorithm, we repeat the two-source simulations above, but with an added U-shape obstacle in the middle of the surveillance area as shown in Fig. 8(a). The thickness of the obstacle is 2 length units, and the attenuation coefficient is $\mu = 0.0693$. This μ value corresponds to halving the radiation intensity with every 10 units of thickness. It is selected such that the obstacle does not completely block the radiation, but allows some radiation to penetrate through it.

The simulation results in Fig. 9(a) show that the obstacles *improve* the accuracy of our algorithm by 24.5% for source 1, but degrade it by 2.4% for source 2. The shielding by the obstacle, although partial, reduces the interference between the two sources, causing the sensors to provide more accurate readings for the individual sources. The false postives and negatives (not shown) are, however, not significantly different than without the obstacle. We conclude that obstacles in the environment may have positive effects, but not significant negative effects, in our algorithm. We will revisit the effects of obstacles in a large-scale network in Section VI-C.

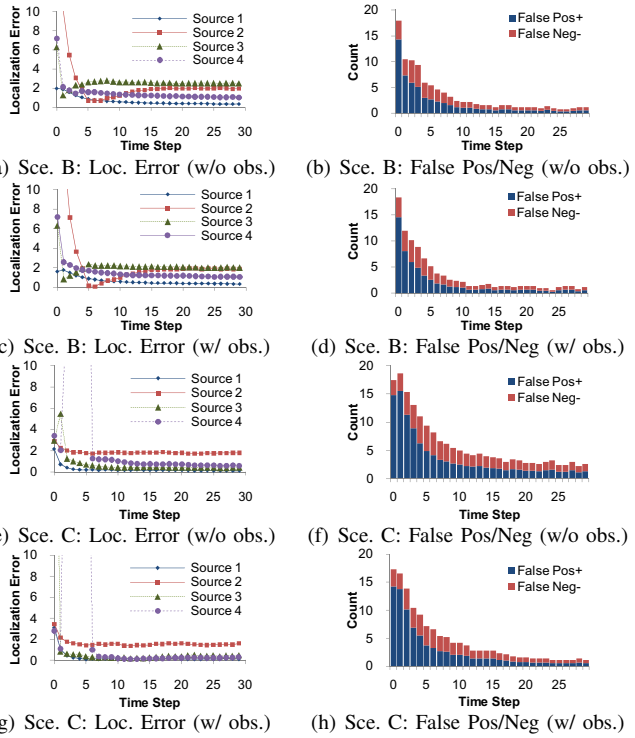
Existing work that localizes multiple radiation sources typically requires modeling all the sources and their interactions. In many cases, ambiguities cannot be resolved because a strong source can give a similar signature as multiple weak sources. Obstacles in the surveillance area pose additional challenges for the existing algorithms. On the other hand, our algorithm can exploit the shielding effect of obstacles to improve the localization accuracy.

C. Large network results

In this section, we illustrate the scalability of our algorithm by simulating a scenario with 196 sensors placed in a grid and 9 radiation sources of non-uniform strengths (between 10–100 μCi) randomly placed in the surveillance area. This scenario has 3$\times$ more sensors and sources compared with that in [2]. In addition, three obstacles located in the area have uneven thickness. The layout of this scenario is given in Fig. 8(b).

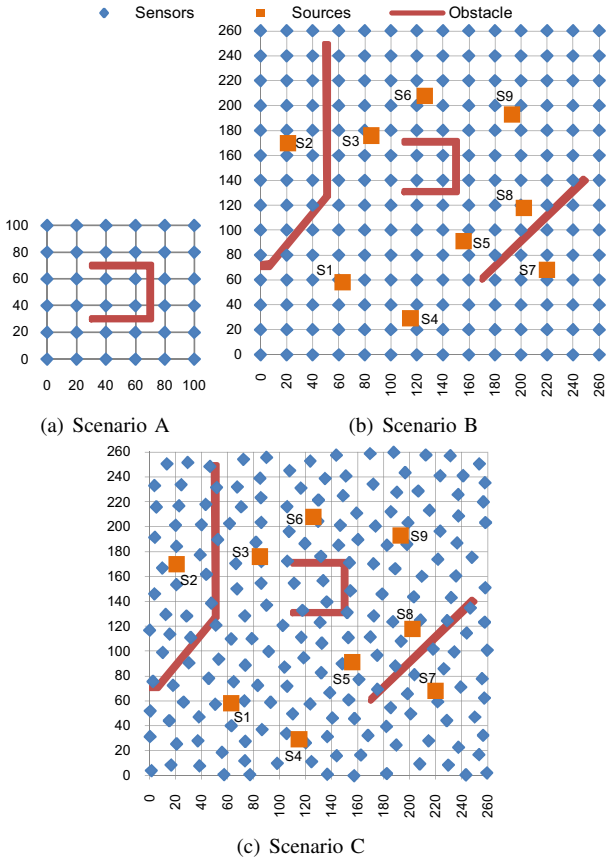
In this simulation, we increase the number of particles to 15000, proportional to the area increase. We evaluate the performance of our algorithm with and without the obstacles. Fig. 7 shows that our algorithm achieves similar localization accuracy in both the large network and the small network, regardless of the number of sources. Similar to the results in Section VI-A, the first few time steps produce a large number of false positives and negatives. In this case, the total number of false positives and negatives has increased by more than 10 times due to the increased number of sources. However, they quickly reduce to around 0.5 on average after several time steps.

To illustrate the effects of obstacles, we compare the normalized localization errors for the experiments with and without obstacles in Fig. 9(b). The results show that some sources benefit from the obstacles more than the others. This is largely determined by the placements of the obstacles relative to the sources and sensors. Fig. 9(c) shows the average improvements in the localization error in time steps 5–29. (The first 5 time steps are omitted in the computation because they



(a) Sce. B: Loc. Error (w/o obs.) (b) Sce. B: False Pos/Neg (w/o obs.)
(c) Sce. B: Loc. Error (w/ obs.) (d) Sce. B: False Pos/Neg (w/ obs.)
(e) Sce. C: Loc. Error (w/o obs.) (f) Sce. C: False Pos/Neg (w/o obs.)
(g) Sce. C: Loc. Error (w/ obs.) (h) Sce. C: False Pos/Neg (w/ obs.)

Fig. 7. Localization error and false positive/negatives in Scenario B and C without obstacles (a–b, e–f) and with obstacles (c–d, g–h) corresponding to the layout in Fig. 8(b). Data for source 4–9 are similar and not shown.



(a) Scenario A (b) Scenario B
(c) Scenario C

Fig. 8. The location of sensors, radiation sources, and obstacles.

are not representative.) The figure shows that 5 sources, S_2, S_3, S_6, S_7, S_9, benefit from the obstacles. All these sources have at least an obstacle located near them. Three sources, S_1, S_4, S_8, have similar localization errors with or without the obstacles. These sources do not have obstacles near them except S_8. Lastly, only one source, S_5, is affected negatively by the obstacle. The localization accuracy of this source drops by as much as 25%.

D. Random sensor placement and out-of-order data delivery

We evaluate the algorithm with another realistic scenario, Scenario C, as shown in Fig. 8(c), where 195 sensors are distributed according to a Poisson point process, and the data are received out-of-order to simulate unpredictable transmission latency. The locations of the source and obstacles remain the same as Scenario B in the previous section.

The results in Fig. 7(e-h) show that Scenario C has similar localization accuracy compared with Scenario B. However, the false positive and false negative rates are increased by 1.6 and 0.9 on average, respectively, in the case without obstacles. In the case with obstacles, the increases are 0.8 and 0.3 on average, respectively. They are mainly due to out-of-order delivery of sensor measurements that causes the particle filter to take longer to converge.

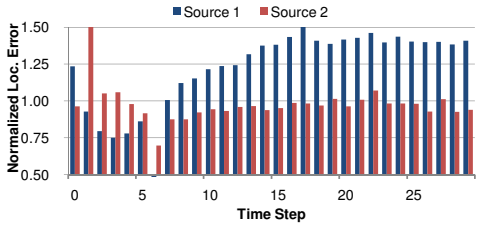
Similarly, obstacles may improve the performance of the algorithm. Fig. 9(c) shows that the localization accuracy is improved when obstacles are present in both Scenario B and C. In addition, as indicated in Fig. 7, the false positive and false negative rates are reduced by 26% and 35%, respectively.

E. Running time

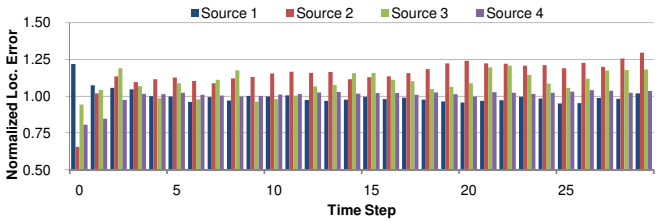
In this section, we demonstrate that our algorithm can take advantage of a multi-core machine for efficiency, which enables it to run quickly even for very large networks. In this experiment, we measure the wall-clock time of the algorithm for each iteration. We run the experiment on two machines. The first machine is a single-socket with Intel Core2 Quad CPU at 2.40 GHz and 2 GB RAM. The second machine is a four-socket with 6-core Intel Xeon CPU E7450 at 2.40 GHz and 24 GB RAM. The average execution times of the algorithm are listed in Table I.

The results show that our algorithm is highly scalable, with average 5× speed up in moving from four cores to 24 cores. The majority of the concurrency is achieved using the mean-shift technique. Manipulating the particles in the predict-and-update steps consume a negligible amount of time because the selective updates of the algorithm effectively discard a large number of particles at the first step of the algorithm.

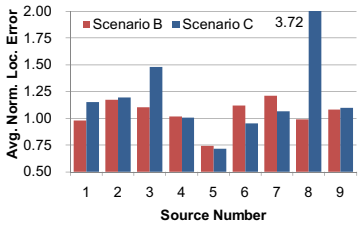
The algorithm takes a fraction of a second to execute on average, even with 196 sensors and 15000 particles. However, we notice that the first few time steps take longer than average because particles are randomly distributed in the beginning. As time proceeds, the particles concentrate at several spots. Therefore, the majority of the iterations only affect a small number of the particles, resulting in shorter execution time.



(a) Scenario A for each time step



(b) Scenario B for each time step (source 5–9 are not shown)



(c) Scenario B & C for each source

Fig. 9. Normalized localization error of a scenario without obstacles to the same scenario with obstacles. Values > 1 imply that obstacles improve the accuracy.

TABLE I
AVERAGE EXECUTION TIME (SECONDS) OF THE ALGORITHM PER ITERATION.

	4-core machine		24-core machine	
# Particles	$N = 36$	$N = 196$	$N = 36$	$N = 196$
2000	0.23	0.22	0.04	0.048
5000	0.54	0.47	0.12	0.129
15000	2.86	1.99	0.50	0.415

VII. CONCLUSION

We have addressed the problem of localizing multiple point radiation sources using a network of sensors with consideration of noise and obstacles in realistic environments. Unlike existing algorithms, the proposed algorithm does not require knowledge about the obstacles, but the presence of obstacles may improve the localization accuracy. The proposed algorithm is robust against unreliable network transmission, out-of-order data delivery, and malfunctioning sensors. The algorithm is scalable to a large network of sensors and many radiation sources, and the computation has significant parallelism that benefits from multi-core processors. Simulation results verified the accuracy of our algorithm in a number of realistic application scenarios.

REFERENCES

[1] M. Morelande, C. Kreucher, and K. Kastella, "A bayesian approach to multiple target detection and tracking," *IEEE Trans. on Signal Proc.*, vol. 55, no. 5, May 2007.

[2] M. Morelande, B. Ristic, and A. Gunatilaka, "Detection and parameter estimation of multiple radioactive sources," in *Proc. of FUSION*, Jul 2007.

[3] X. Sheng and Y.-H. Hu, "Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks," *IEEE Trans. on Signal Processing*, vol. 53, no. 1, 2005.

[4] N. S. V. Rao, M. Shankar, J.-C. Chin, D. K. Y. Yau, S. Srivathsan, S. S. Iyengar, Y. Yang, and J. C. Hou, "Identification of low-level point radiation sources using a sensor network," in *Proc. of IPSN*, 2008.

[5] J.-C. Chin, D. K. Y. Yau, N. S. V. Rao, Y. Yang, C. Y. T. Ma, and M. Shankar, "Accurate localization of low-level radioactive source under noise and measurement errors," in *Proc. of SenSys*, Nov 2008.

[6] A. Gunatilaka, B. Ristic, A. Skvortsov, and M. Morelande, "Parameter estimation of a continuous chemical plume source," in *FUSION*, 2008.

[7] R. Niu and P. K. Varshney, "Distributed detection and fusion in a large wireless sensor network of random size," *EURASIP Jour. on Wireless Communications and Networking*, vol. 2005, no. 4, 2005.

[8] J. Chen, R. Hudson, and K. Yao, "Maximum-likelihood source localization and unknown sensor location estimation for wideband signals in the near-field," *IEEE Trans. on Signal Processing*, vol. 50, no. 8, Aug 2002.

[9] T. He, S. Krishnamurthy, J. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. Krogh, "Energy-efficient surveillance system using wireless sensor networks," in *Proc. of MobiSys*, 2004.

[10] D. Li, K. Wong, Y. H. Hu, and A. Sayeed, "Detection, classification, and tracking of targets," *IEEE Signal Proc. Mag.*, vol. 19, no. 2, Mar 2002.

[11] J. W. Howse, L. O. Ticknor, and K. R. Muske, "Least squares estimation techniques for position tracking of radioactive sources," *Automatica*, vol. 37, pp. 1727–1737, 2001.

[12] A. Gunatilaka, B. Ristic, and R. Gailis, "On localisation of a radiological point source," *IDC '07: Information, Decision and Control*, Feb 2007.

[13] X. Xu, N. S. V. Rao, and S. Sahni, "A computational geometry method for localization using difference of distances," *IEEE Trans. on Sensor Networks*, 2008.

[14] N. S. V. Rao, M. Shankar, J.-C. Chin, D. K. Y. Yau, Y. Yang, J. C. Hou, X. Xu, and S. Sahni, "Localization under random measurements with application to radiation sources," in *Proc. of Information Fusion*, 2008.

[15] M. Ding and X. Cheng, "Fault-tolerant target tracking in sensor networks," in *Proc. of MobiHoc*, 2009.

[16] Y. Cheng and T. Singh, "Source term estimation using convex optimization," in *Proc. of Information Fusion*, June 2008.

[17] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization in distributed sensor networks," *ACM Transactions on Embedded Computing Systems*, vol. 3, no. 1, February 2004.

[18] B. Ristic, M. Morelande, and A. Gunatilaka, "A controlled search for radioactive point sources," in *Proc. of Information Fusion*, June 2008.

[19] C. Christodoulopoulos, C. Kyriakopoulos, and A. G. Kanatas, "A realistic approach to source localization using a wireless robotic network," in *Proc. of RoboComm*. Piscataway, NJ, USA: IEEE Press, 2007, pp. 1–4.

[20] P. Fabiani, H. H. Gonzlez-Baos, J. C. Latombe, and D. Lin, "Tracking an unpredictable target among occluding obstacles under localization uncertainties," *Robotics and Autonomous Systems*, vol. 38, no. 1, 2002.

[21] J. H. Hubbell, "Photon cross sections, attenuation coefficients, and energy absroption coefficients from 10 kev to 100 gev," *NSRDS-NBS 29*, 1969.

[22] R. Vilim, R. Klann, S. de la Barrera, P. Vilim, and I. Ross, "Tracking of weak radioactive sources in crowded venues," in *IEEE Nuclear Science Symposium Conference Record*, 2009.

[23] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pat. Anal. Mach. Intell.*, vol. 24, no. 5, 2002.

[24] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," in *IEE Proc. F In Radar and Signal Processing*, vol. 140, August 2002, pp. 107–113.

[25] R. A. Muller. (2004, June) The dirty bomb distraction. [Online]. Available: http://muller.lbl.gov/TRessays/29-Dirty_Bombs.htm