Optimal Control-Based Strategy for Sensor Deployment

Ahmad Ababnah, *Student Member, IEEE*, and Balasubramaniam Natarajan, *Senior Member, IEEE*

***Abstract*—In sensor network-based detection/surveillance, one of the first challenges to address is the optimal deployment of sensors such that detection requirements are satisfied in a given area. Specifically, we pose the following question: Given a finite number of sensors, what is the best way to deploy these sensors in order to minimize the squared difference between achieved and required detection/miss probabilities? In this paper, we develop a novel optimal control theory based formulation of this sensor deployment problem. Exploiting similarities between the problem at hand and the linear quadratic regulator, an analytical solution is derived and tested. Unlike prior efforts that rely purely on heuristics, the proposed optimal control framework provides a theoretical basis for the resulting solution. As the complexity of the optimal control based solution is high, we develop a low-complexity approximation called Max_Deficiency algorithm. Using simulation results, we show that the proposed algorithms outperform existing methods by using 10% to 30% fewer number of sensors to satisfy detection requirements.**

***Index Terms*—Detection systems, linear quadratic regulator, optimal control, sensor deployment.**

I. Introduction

SENSOR networks have emerged as a viable solution for many detection and surveillance applications [1]–[3]. The idea is to distribute sensors in an area of interest and have these sensors detect targets or phenomena of interest with high accuracy. Remedial actions can be taken once a target/phenomena has been detected. The detection performance of such sensor networks strongly depends on many factors such as the number of sensors available, the environment where sensors are deployed, and the sensor deployment strategy.

The strategy for deploying sensors becomes even more critical in applications where the detection requirements are not uniform. For example, within a given area there may be highly sensitive subregions where one might demand a higher probability of detection relative to other subregions. In this case, the question that needs to be addressed is the following. Given a finite number of sensors, what is the optimal deployment strategy that will make use of the available sensors in the most effective manner? In other words, what are the optimal locations to place the available sensors such that the difference between achieved and required detection (or equivalently miss) probabilities is minimized? Throughout this paper, any reference to optimality is with respect to the squared difference (i.e., squared error) between achieved and required detection/miss probabilities.

The sensor deployment problem can be formulated and studied in different ways depending on the application for which the sensor network is used. In [4]–[6], for instance, sensor deployment is studied in the context of distribution networks (e.g., water, air, etc.). In [4], two sensor deployment problems are investigated. In the first problem, the goal is to deploy a fixed number of sensors in order to minimize a contaminant's detection time. This problem is shown to be equivalent to the asymmetric k-center problem [7]. In the second problem, the authors attempt to minimize the number of sensors needed to detect a contaminant within a given time limit. This was shown to be equivalent to the dominating set problem. However, [4] assumes that sensors detection model is deterministic (i.e., sensors always detect contaminants perfectly regardless of their concentration), which may not be realistic in general. Furthermore, sensors are assumed to be point sensors (i.e., with no effective area of coverage). Adopting the same assumptions (i.e., deterministic point sensors), [5] considers the problem of deploying a fixed number of sensors in order to minimize a contaminant's net effect. The deployment problem is modeled as a mixed integer linear programming (MILP) problem. The work of [5] was extended in [6], in which the contaminant's effect on all locations of interest was incorporated in the optimization problem. However, solving an MILP problem can be computationally expensive, especially for large problems. The primary focus of other prior efforts [8]–[11] was on estimating the minimum number of sensors required to satisfy certain detection requirements. In [8] and [9], the authors assume a binary sensing model for the sensors. As a result, the sensor deployment problem reduces to a coverage problem. The binary sensing model assumption limits the impact of this work as the sensing model does not accurately reflect realistic sensing characteristics. An exponential sensing model proposed has been accepted as a more realistic model for sensors. In [10], the authors assume an exponential sensing model and they also introduce the Minn_Miss algorithm, which is a heuristic algorithm that is used to deploy sensors in an area with uniform detection requirements. In [11], the deployment problem is represented by a linear shift-invariant (LSI) system. The sensor deployment is then described as an integer programming (IP) problem that is in general NP hard. Due to the complexity of an optimal solution for the IP problem, a heuristic algorithm, called Diff_Deploy, is introduced. It is shown that Diff_Deploy uses a fewer number of sensors than the Minn_Miss to satisfy the same detection requirements. However, since Diff_Deploy is a heuristic algorithm, it is difficult to guarantee optimality.

Manuscript received August 10, 2008; revised August 21, 2009; accepted February 1, 2010. Date of publication July 8, 2010; date of current version November 10, 2010. This work was supported in part through the award of a contract from the Marine Corps Systems Command (MCSC) to M2 Technologies Inc. This paper was recommended by Editor W. Pedrycz.

The authors are with the WiCom Group, Electrical and Computer Engineering Department, Kansas State University, Manhattan, KS 66506 USA (e-mail: ababnah@ksu.edu; bala@ksu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCA.2010.2049992

Therefore, there is a strong need to bring theoretical rigor to the analysis of the sensor deployment problem. We attempt to address this need through the use of the optimal control theory.

In this paper, we propose a novel optimal control-based sensor deployment algorithm. Here, we consider K sensors each with an exponentially decaying sensing model. Given these K sensors, we illustrate that the deployment of a single sensor at a time can be regarded as one step in a linear quadratic regulator (LQR) [12]. The control vectors at each discrete step in the evolution of the system corresponds to the position of the sensor to be deployed. The optimal solution (control vector) is then derived as the vector satisfying the Karush–Kuhn–Tucker (KKT) conditions. This solution is determined using the sweep method [13] that is commonly used to solve LQR problems. Using simulation results, we illustrate that the optimal control-based deployment strategy uses a smaller number of sensors to satisfy detection requirements relative to Diff_Deploy algorithm. Additionally, unlike Diff_Deploy and other prior efforts based purely on heuristics, the optimal control theory framework provides a theoretical foundation to sensor deployment. Finally, we also propose a low-complexity alternative to the optimal control-based algorithm called the Max_Deficiency algorithm. Here, the difference between the required and achieved detection probability is computed after deploying a single sensor. Then, a sensor is placed at the location that corresponds to the highest difference (deficiency). Using simulations, we illustrate that the low-complexity algorithm is comparable to the optimal control-based strategy in terms of detection performance and both these proposed approaches employ 10% to 30% fewer number of sensors relative to Diff_Deploy.

The rest of the paper is organized as follows: In Section II, the system model and the problem definition are presented. In Section III, the deployment problem is formulated as an LQR problem and a dynamic optimization-based solution of LQR is discussed. We present the optimal control-based deployment algorithm in Section IV. In Section V, the Max_Deficiency algorithm is introduced. Simulation results in Section VI show the performance of the proposed methods relative to the Diff_Deploy algorithm. Finally, conclusions are outlined in Section VII.

II. System Model

The system under consideration consists of an area of interest where region-wise detection requirements are provided by the end user. We model the area of interest as a grid $\mathcal{G}$ of $N_x \times N_y$ points. The detection/miss requirements at every point on the grid are ordered in two $N_xN_y \times 1$ vector $\mathbf{p}_d^{\mathrm{req}}/\mathbf{p}_m^{\mathrm{req}}$. Additionally, the sensing model and the number of sensors available serve as inputs to our sensor deployment algorithm. Given these inputs, the objective of this work is to determine the optimal sensor placement that would minimize the square difference between achieved and required detection/miss probabilities. It is important to note that we assume a simple detection model in which a target is declared to be detected if at least a single sensor in the network is able to detect it. Results from this simple framework can be extended to other schemes like distributed detection. We first examine the sensing model and later describe the relationship between the sensors' detection model, their positions, and the resulting detection performance of the network.

There are two common sensing models found in literature: the binary detection model and the exponential detection model. Both models share the assumption that the detection capability of a sensor depends on the distance between the sensor and the phenomena (target) to be detected. In the binary detection model, the probability of detection $p_d(t,s)$ is given as [14], [15]

$$p_d(t,s) = \begin{cases} 1 & \text{if } d(t,s) \le r_d \\ 0 & \text{if } d(t,s) > r_d \end{cases} \tag{1}$$

where r_d is the detection radius and $d(t,s)$ is the distance between the target's position "t" and the sensor location "s" on a plane. If the distance between the sensor and target is greater than r_d, then the target is not detectable by that sensor. However, if the target is within the detection radius, it will be always detected. This is a simple model, but it is not very realistic. The exponential model is a more realistic model to adopt where the probability of detection [16], [17] corresponds to

$$p_d(t,s) = \begin{cases} e^{-\alpha d(t,s)} & \text{if } d(t,s) \le r_d \\ 0 & \text{if } d(t,s) > r_d \end{cases} \tag{2}$$

where α is a decay parameter that is related to the quality of a sensor or the surrounding environment. In the exponential model of (2), even if a target is within the detection radius, there is a probability that it will not be detected (i.e., it will be missed). A wide range of practical sensors [18] (e.g., infrared, ultrasound) fit this general model. Therefore, we assume an exponential sensing model throughout the rest of this paper. It is important to note that the choice of the detection model does not affect the basic formulation of the algorithms proposed in Sections III and IV.

Following the LSI model as in [11], the process of linking individual sensors' detection characteristic to the overall probability of detection requirements on the grid is mathematically quantified using miss probabilities p_{miss} ($p_{\mathrm{miss}} = 1 - p_d$, where p_d is the probability of detection). The probability of a target being detected by any sensor on the grid is the complement of the target being missed by all the sensors on the grid. The overall miss probability $M(x,y)$ corresponds to the probability that a target at point (x,y) will be missed by all sensors, i.e.,

$$M(x,y) = \prod_{(i,j)\in\mathcal{G}} p_{\mathrm{miss}}\left((x,y),(i,j)\right)^{u(i,j)}. \tag{3}$$

Here, $u(i,j)$ represents the presence or absence of a sensor at the location (i,j) on the grid, and corresponds to

$$u(i,j) = \begin{cases} 1, & \text{if there is a sensor at } (i,j) \\ 0, & \text{if there is no sensor at } (i,j). \end{cases} \tag{4}$$

Taking the natural logarithm of both sides in (3) results in

$$m(x,y) = \sum_{(i,j)\in\mathcal{G}} u(i,j) \ln p_{\mathrm{miss}}\left((x,y),(i,j)\right) \tag{5}$$

where $m(x,y)$ is called the overall logarithmic miss probability at point (x,y) [11]. Let us define the function $b(x,y)$ as follows:

$$b(x,y) = \begin{cases} \ln p_{\text{miss}}\left((x,y),(0,0)\right), & d\left((x,y),(0,0)\right) \le r_d \\ 0, & d\left((x,y),(0,0)\right) > r_d. \end{cases} \tag{6}$$

The overall logarithmic miss probabilities for all points on the grid can be arranged in a vector $\mathbf{m} = [m(x,y), \forall (x,y) \in \mathcal{G}]^T$ of dimension $N_xN_y \times 1$ that corresponds to

$$\mathbf{m} = \mathbf{B}\mathbf{u}. \tag{7}$$

Here, $\mathbf{u} = [u(i,j), \forall (i,j) \in \mathcal{G}]^T$ is the deployment vector of dimension $N_xN_y \times 1$. The $((i-1)N_y + j)$-th element of $\mathbf{u}$ indicates the number of sensors deployed at point (i,j) on the grid. The matrix $\mathbf{B}$ is of dimension $N_xN_y \times N_xN_y$, and it contains $\{b(x-i, y-j), \forall (x,y) \in \mathcal{G}, (i,j) \in \mathcal{G}\}$. $b(x-i,y-j)$ corresponds to the (r,c)-th entry of $\mathbf{B}$, where $r = (x-1)N_y + y$ and $c = (i-1)N_y + j$. Essentially, $b(x-i,y-j)$ quantifies the effect of placing a sensor at point (i,j) on the logarithmic miss probability at point (x,y) on the grid. The logarithmic miss probabilities can be easily converted to detection probabilities at a later stage.

The question we attempt to address in this paper is the following: Given a number of sensors, where can the sensors be placed to minimize the squared error between achieved and required detection/miss probabilities? Once again, the squared error (SE) between required and achieved detection probabilities at all points in the grid can be mapped to the SE between required and achieved miss probabilities. However, one should note that after the sensors have been deployed, the achieved detection probability at some of the grid points will meet/exceed the detection requirements at these points. Therefore, we emphasize on minimizing the squared error at the points for which detection/miss requirements are not met.

Let $\mathbf{m}_{\text{req}}$ be the required miss probability vector and let $\mathbf{m}_k$ be the achieved miss probability vector resulting from deploying k sensors to the grid. Our sensor deployment problem can be mathematically formulated as follows:

$$\arg\min_{\mathbf{u}} \sum_{j:p_d^K(j) < p_d^{\text{req}}(j)} \left(\mathbf{m}_K(j) - \mathbf{m}^{\text{req}}(j)\right)^2$$
$$\text{subject to } \{\mathbf{1}^T\mathbf{u} = K \tag{8}$$

where $p_d^K(j)$ is the detection probability at the jth grid point after K sensors have been deployed to the grid. $\mathbf{1}^T$ indicates the transpose of an $N_xN_y \times 1$ vector, with all entries set to 1. K is the total number of available sensors.

III. Optimal Control Formulation

The problem of minimizing the square error between achieved and required detection probabilities can be viewed as minimizing the square difference between achieved and required miss or overall logarithmic probabilities. Define $\mathbf{x}_k$ to be the difference between the required log miss probability vector $(\mathbf{m}_{\text{req}})$ and the log miss probability vector achieved after deploying k sensors $(\mathbf{m}_k)$, i.e., $\mathbf{x}_k = \mathbf{m}_k - \mathbf{m}_{\text{req}}$. The system described in (7) can be written in terms of the dynamic model

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{B}\mathbf{u}_k \tag{9}$$

where $\mathbf{u}_k$ is the deployment vector at the kth step. In typical control problems, the index k indicates the time index describing time evolution of the system. In our case, we assume the sequential placing of sensors (i.e., sensors are placed one at a time) with k representing the kth step in this process.

In terms of the dynamic state vector $\mathbf{x}_k$, we can define a weighted SE cost function J as

$$J = \frac{1}{2}\mathbf{x}_K^T\mathbf{Q}_f\mathbf{x}_K + \frac{1}{2}\sum_{k=0}^{K-1}\left(\mathbf{x}_k^T\mathbf{Q}\mathbf{x}_k + \mathbf{u}_k^T\mathbf{R}\mathbf{u}_k\right). \tag{10}$$

Here, $\mathbf{u}_k$ is the deployment vector for the kth sensor. $\mathbf{Q}$, $\mathbf{Q_f}$ and $\mathbf{R}$ are positive-definite diagonal weighting matrices with dimension $N_xN_y \times N_xN_y$ that are chosen by the designer. In our problem, a good choice of $\mathbf{Q}$, $\mathbf{Q_f}$, and $\mathbf{R}$ is one that reflects the detection requirements on the grid. That is, if the detection requirement at a certain point is relatively large compared to other points, then the entries in the matrices $\mathbf{Q}$, $\mathbf{Q_f}$, and $\mathbf{R}$ that correspond to that point should be in such a way that the resulting solution will be biased toward satisfying that point before other points. One choice that fits well with the above reasoning is the following: $\mathbf{R}(i) = (\mathbf{m}_{\text{req}}(i)/\mathbf{1}^T\mathbf{m}_{\text{req}})^{-1}$, where $\mathbf{R}(i)$ is the ith diagonal element of $\mathbf{R}$ and $\mathbf{m}_{\text{req}}(i)$ is the overall logarithmic miss requirement at point i on the grid. The ith diagonal elements of $\mathbf{Q}$ and $\mathbf{Q}_f$ are given as $\mathbf{Q}(i) = \mathbf{Q}_f(i) = (\mathbf{R}(i))^{-1}$, where $(\cdot)^{-1}$ denotes the inverse operation. The goal of the control problem is to determine the sequence of control vectors $\{\mathbf{u}_k, k = 0, 1, \ldots, K-1\}$ that would minimize the cost function J. The squared error cost function penalizes both positive and negative deviations from the required detection probability profile. To avoid incurring a penalty for satisfying/exceeding detection requirements, the error terms corresponding to a satisfied point is set to zero in J. Therefore, after each sensor deployment, the cost function to be minimized is the squared error evaluated at the points where detection/miss requirements are not satisfied. We refer to this squared error cost function as the effective SE. Therefore, the effective SE corresponds to

$$\text{eSE}(k) = \sum_{j:p_d^k(j) < p_d^{\text{req}}(j)} x_k(j)^2 \tag{11}$$

where $p_d^k(j)$ is the achieved detection probability at the jth grid point after k sensors have been deployed in the grid. The formulation of the problem discussed above is known as a linear-quadratic regulator problem in control theory literature [12], [19]. Therefore, we can employ the dynamic optimization techniques used for solving LQR problems to solve the sensor deployment problem as described in the next subsection.

A. Dynamic Optimization

The objective of dynamic optimization [20] in general is to find a series of control vectors that would minimize a cost function that is related to the system under consideration. Suppose that a discrete system model is given as

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k), \quad k = 0, 1, \ldots, K-1 \tag{12}$$

with an initial condition $\mathbf{x}_0 = \mathbf{x}_{initial}$. Furthermore, suppose that it is desired to minimize a cost function related to the system as

$$J = L(\mathbf{x}_K) + \sum_{k=1}^{K-1} V_k(\mathbf{x}_k, \mathbf{u}_k). \tag{13}$$

The minimization can be carried out by solving for the value of $\mathbf{u}_k$ at every instant k. A common and easy way to solve the equations is to use Lagrange multipliers and apply the discrete KKT conditions [12], [13], [20]. The optimal solution satisfies the following conditions:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k) \tag{14}$$

$$\mathbf{x}_0 = \mathbf{x}_{initial} \tag{15}$$

$$\lambda_k = \nabla_{\mathbf{x}_k} f(\mathbf{x}_k, \mathbf{u}_k)^T \lambda_{k+1} + \nabla_{\mathbf{x}_k} V_k \tag{16}$$

$$\lambda_K = \nabla_{\mathbf{x}_K} L \tag{17}$$

$$0 = \nabla_{\mathbf{u}_k} V_k + \lambda_{k+1}^T \nabla_{\mathbf{u}_k} f(\mathbf{x}_k, \mathbf{u}_k). \tag{18}$$

We can now use the conditions (14)–(18) to solve our LQR problem. Our system model is given as

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{B}\mathbf{u}_k. \tag{19}$$

Our objective is to find the (optimal) control vector sequence $\{\mathbf{u}_k, k = 0, 1, \ldots, K-1\}$ that would minimize the cost function J, where J is given as

$$J = \frac{1}{2}\mathbf{x}_K^T \mathbf{Q}_f \mathbf{x}_K + \frac{1}{2}\sum_{k=0}^{K-1}\left(\mathbf{x}_k^T \mathbf{Q}\mathbf{x}_k + \mathbf{u}_k^T \mathbf{R}\mathbf{u}_k\right). \tag{20}$$

Applying the discrete KKT conditions (14)–(18) on the above system model will result in the following optimality conditions:

$$\mathbf{x}_0 = \mathbf{x}_{initial} \tag{21}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{B}\mathbf{u}_k \tag{22}$$

$$\boldsymbol{\lambda}_k = \boldsymbol{\lambda}_{k+1} + \mathbf{Q}\mathbf{x}_k \tag{23}$$

$$\boldsymbol{\lambda}_K = \mathbf{Q}_f \mathbf{x}_K \tag{24}$$

$$\mathbf{u}_k = -\mathbf{R}^{-1}\mathbf{B}^T \boldsymbol{\lambda}_{k+1}. \tag{25}$$

The optimal $\mathbf{u}_k$ in the above equations can be found using a method known as the sweep method [13] and [21]. In the sweep method, the Lagrange multiplier of step k is calculated as

$$\boldsymbol{\lambda}_k = \mathbf{P}_k \mathbf{x}_k \tag{26}$$

where

$$\mathbf{P}_k = \left(\mathbf{P}_{k+1} - \mathbf{P}_{k+1}\mathbf{B}\mathbf{S}_k^{-1}\mathbf{B}^T\mathbf{P}_{k+1}\right) + \mathbf{Q} \tag{27}$$

$$\mathbf{S}_k = \mathbf{R} + \mathbf{B}^T \mathbf{P}_{k+1}\mathbf{B}. \tag{28}$$

Let

$$\mathbf{G}_k = \mathbf{S}_k^{-1}\mathbf{B}^T\mathbf{P}_{k+1}. \tag{29}$$

We can solve for $\mathbf{u}_k$ in terms of $\mathbf{G}_k$ and $\mathbf{x}_k$ as

$$\mathbf{u}_k = -\mathbf{G}_k \mathbf{x}_k. \tag{30}$$

The calculation of $\mathbf{S}_k$, $\mathbf{G}_k$, and $\mathbf{P}_k$ is performed in reverse order starting from $k = K-1$ to $k = 0$. The final conditions that are used in the initial step of the sweep method are $\mathbf{P}_K = \mathbf{Q}_f$, $\mathbf{G}_K = \mathbf{0}$, and $\mathbf{S}_K = \mathbf{0}$. Note that matrices $\mathbf{P}_k$, $\mathbf{S}_k$, and $\mathbf{G}_k$ are all of dimension $N_x N_y \times N_x N_y$.

The steps of the sweep method are summarized in Algorithm 1 [20].

Algorithm 1 Sweep method

Initialization $\mathbf{P}_K = \mathbf{Q}_f$, $\mathbf{G}_K = \mathbf{0}$, and $\mathbf{S}_K = \mathbf{0}$.
for $k = K-1, K-2, \ldots, 0$ **do**
  Calculate $\mathbf{S}_k$ from (28).
  Calculate $\mathbf{G}_k$ from (29).
  Store $\mathbf{G}_k$.
  Calculate $\mathbf{P}_k$ from (27).
end for
for $k = 0, 1, \ldots, K-1$ **do**
  Calculate $\mathbf{u}_k$ from (30)
end for

An application of the discrete KKT conditions, along with the sweep method, to the sensor deployment problem results in the optimal solution to the sensor deployment problem. Although the proposed approach offers the optimal solution, it also adds significant computational complexity that grows with K. The complexity is due to the need to evaluate matrices $\mathbf{S}_k$, $\mathbf{G}_k$, and $\mathbf{P}_k$ for $k = K-1, \ldots, 0$ as well as the necessity to store all instances of matrix $\mathbf{G}$. There is another important point to note: The resulting optimal solution vector $\mathbf{u}_k$ usually consists of $N_x N_y$ real continuous values and not binary values. In order to have a binary integer solution, a new vector $\mathbf{u}_k^o$ is constructed by placing a 1 at the index where $\mathbf{u}_k$ has its maximum value and placing a 0 in place of the remaining entries. Intuitively, this implies the following: We can interpret the real values in the optimal control vector $\mathbf{u}_k$ obtained from the sweep method to indicate the desirability of placing a sensor in a particular position on the grid. Therefore, when we sequentially place sensors, the index of $\mathbf{u}_k$ corresponding to the maximum value is the location on the grid where placing the next sensor will have its maximum impact on the cost function J. This observation can be further exploited to develop a low-complexity alternate to the optimal algorithm, which we will discuss next.

Though the proposed algorithm is initially designed for minimizing the effective SE between achieved and required detection/miss probabilities, it can be modified in order to find the number of sensors needed to satisfy the required detection/miss profile. This is done by initially setting K to a large value ($\leq N_x N_y$) in the algorithm for sequential deployment. After each sensor deployment, the resulting detection/miss probability profile is compared against the required profile. The algorithm terminates when the detection/miss requirement is met at each point. The number of sensors deployed at that stage provides an estimate of the minimum number of sensors required to satisfy detection/miss requirements.

IV. Optimal Control-Based Deployment Algorithm

In this section, we introduce the deployment algorithm, which is based on the LQR formulation of the sensor deployment problem. Algorithm 2 illustrates the steps of the optimal control-based sensor deployment algorithm.

Given the total number of sensors, K, and the $\mathbf{B}$ matrix, the algorithm evaluates the feedback gain matrix (i.e., $\mathbf{G}_k, k = K-1 : -1 : 0$) using the sweep method discussed earlier. Sensors are deployed sequentially until all available sensors have been deployed or when detection requirements at all points on the grid have been satisfied (i.e., effective SE equals 0). In the kth iteration, the set of points for which the detection/miss requirements are satisfied is determined and the entries in the vector $\mathbf{x}_{k-1}$ corresponding to these points are set to 0. Afterward, the kth deployment vector is calculated as in (30). However, since the deployment vector can only have $\{0, 1\}$ entries, the entry in the deployment vector $\mathbf{u}$ corresponding to the largest entry (with index $j_{\max}$) in the kth deployment vector (i.e., $\mathbf{u}_k$) is set to 1. After updating the deployment vector, the resulting overall logarithmic miss can be calculated as in (7). It is also possible to calculate the achieved detection probability vector as $\mathbf{p}_d^k = \mathbf{1} - \exp(\mathbf{m}_k)$, where $\exp(\mathbf{m}_k)$ indicates the exponential of each entry of $\mathbf{m}_k$.

Algorithm 2 Optimal Control Based Algorithm

```
Input: p_d^req (detection requirement), K (number of available sensors), and B
Outputs: u (deployment vector).
for k = K − 1 : −1 : 0 do
    Evaluate G_k as in Algorithm 1.
end for
Initialization: k = 0, u = 0
while k ≤ K or p_d^req ⋡ p_d^k (i.e., eSE ≠ 0) do
    Find set of grid points with unsatisfied detection requirements {i : p_d^k(i) ≥ p_d^req(i)}.
    Set x_{k−1}(i) = 0 (i.e., Only an error at unsatisfied points is considered).
    Calculate the control vector u_k = −G_k x_k.
    Find index j_max, where j_max = max_index(u_k). (The function max_index(u_k) returns the index of the largest entry in vector u_k.)
    Update the deployment vector (i.e., u(j_max) = 1).
    Calculate m_k = Bu.
    Evaluate achieved detection profile p_d^k = 1 − exp(m_k).
    Calculate x_k.
    Increment number of sensors in the grid: k = k + 1
end while
```

V. Max_Deficiency Deployment Algorithm

Due to the computational cost associated with the optimal control solution presented earlier, it is advantageous to develop a low-complexity algorithm that is relatively simple to implement. In this section, we introduce a new algorithm, which we call the Max_Deficiency algorithm.

We assume that given K sensors, we will be deploying them sequentially, until all sensors have been deployed or the detection requirements have been met at all the grid points. In the kth iteration, the Max_Deficiency algorithm calculates the difference $\mathbf{p}_\delta$ between the required $\mathbf{p}_d^{\text{req}}$ and achieved detection probabilities $\mathbf{p}_d^{k-1}$ and then deploys the kth sensor to the point $j_{\max}$ on the grid where $\mathbf{p}_\delta$ is maximum. The deployment vector $\mathbf{u}$ is updated by placing a 1 instead of 0 at its $j_{\max}$ entry. Employing (7), we calculate the resulting logarithmic miss probability $\mathbf{m}_k$. The resulting detection probability vector $\mathbf{p}_d^k$ is calculated as $\mathbf{p}_d^k = \mathbf{1} - \exp(\mathbf{m}_k)$. In other words, at each step in the deployment algorithm, we identify the point on the grid that is most deficient in terms of meeting the detection requirements and we place a sensor in that position, calculate its effect, and then repeat the process. Identifying the point with the maximum deficiency is similar to identifying the location on the grid that will have the maximum impact on the cost function J.

The Max_Deficiency algorithm is illustrated in Algorithm 3.

Algorithm 3 Max_Deficiency Algorithm

```
Inputs: p_d^req, K
Outputs: u (i.e., deployment vector)
Initialization: p_d^0 = 0, p_δ = p_d^req, and k = 0
while k ≤ K or p_δ ≻ 0 do
    Calculate p_δ = p_d^req − p_d^k.
    Find index j_max, where j_max = max_index(p_δ). (The function max_index(p_δ) returns the index of the largest entry in vector p_δ.)
    Place a sensor at position j_max (u(j_max) = 1).
    Calculate m_k = Bu.
    Evaluate achieved detection profile p_{d,k} = 1 − exp(m_k).
    Increment number of sensors in the grid: k = k + 1
end while
```

VI. Simulation Results

In this section, the performance of both the optimal control-based and the Max_Deficiency algorithm is compared to that of the Diff_Deploy algorithm [11].

In the first experiment, we compare the number of sensors needed by the three algorithms to meet the detection requirements as we vary the decay parameter α. The area of interest is modeled as a grid of 25×25 points. The area consists of three subregions, each with its own detection requirement as is shown in Fig. 1. We assume that all sensors employ a detection radius of $r_d = 5$. Table I presents the number of sensors needed by the three algorithms as α varies. The stopping criteria in both the Diff_Deploy and Max_Deficiency algorithm is meeting the detection/miss requirements at all grid points. The optimal control-based algorithm employs the same criteria, but in terms of the effective SE (i.e., deployment terminates when the effective SE equals 0). As expected, the optimal control-based algorithm outperforms the Diff_Deploy and the Max_Deficiency in terms of the number of sensors needed to satisfy the detection requirements. This is because the cost function used in the design of the optimal control algorithm

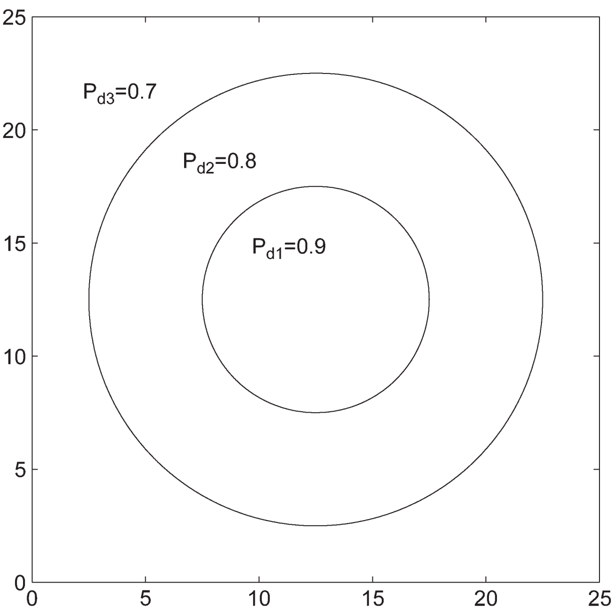

Fig. 1. Required detection probability profile.

TABLE I
NUMBER OF SENSOR NEEDED BY EACH ALGORITHM FOR DIFFERENT α'S

α	Diff_Deploy	Max_Deficiency	Optimal Control Based
$\alpha = 0.05$	22	20	20
$\alpha = 0.1$	31	28	27
$\alpha = 0.15$	32	31	28
$\alpha = 0.2$	38	38	34
$\alpha = 0.25$	41	40	39

is the effective SE while there is no clear cost function in the heuristic Diff_Deploy. On the other hand, the effective cost function in the Max_Deficiency algorithm is the error at a single point only, with no regard to the errors at other grid points. In the optimal control deployment algorithm, using the matrix $\mathbf{B}$ along with the sweep method implies that information regarding the effect of each sensor placement on the entire grid is incorporated in the deployment process. This is in contrast to the Max_Deficiency algorithm, which makes its deployment decision based solely on the effect of a sensor at its deployment location. Additionally, as α increases, the detection sensitivity of a single sensor decreases. Therefore, the number of sensors needed to satisfy the detection requirements increases as α increases. This is confirmed in our results, which are presented in Table I.

In order to further compare the performance of the three algorithms, we present the evolution of effective SE between achieved and required detection probability profiles as sensors get deployed in the grid (see Fig. 2). We specifically examine the SE for the points that are yet to be satisfied in terms of the detection/miss requirements, which we call the effective SE. Fig. 2 considers the case of $\alpha = 0.15$ listed in Table I. It is obvious that the effective SE of the optimal control-based algorithm converges faster than that of the Diff_Deploy algorithm and the Max_Deficiency algorithm. The results indicate that we can satisfy the detection/miss requirements with a fewer number of sensors if we employ the proposed optimal control-based algorithm. Fig. 3 shows the achieved detection probability profile resulting from deploying sensors based on the three algorithms. From Fig. 3(c), it is evident that the proposed approach does

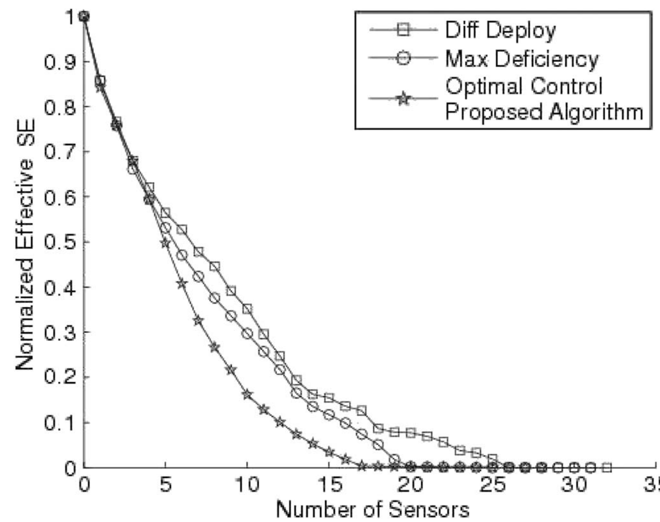

Fig. 2. Effective SE convergence.

not overbudget for satisfying the detection requirements in each subregion. For example, in the outer region of the grid, where the detection requirement is set to 0.7, we note that the achieved detection probability in that subregion is around 0.7. This is in contrast to the Diff_Deploy algorithm, where the minimum achieved detection probability is close to 0.86.

In the second experiment, the performance of the Diff_Deploy is compared to that of the Max_Deficiency algorithm. The grid size is 50 $\times$ 50. The detection requirements profile is shown in Fig. 4. The numerical values of $p1$, $p2$, and $p3$ along with the number of sensors needed to satisfy the detection requirements are listed in Table II. As evident from Table II, the Max_Deficiency algorithm always uses a smaller number of sensors than the Diff_Deploy algorithm. The reduction in the number of sensors depends on the required detection profile, and in our simulations, it ranges from about 10% to 30%.

In the third experiment, the performance of the Max_Deficiency algorithm versus that of the Diff_Deploy algorithm is examined in the presence of obstacles. The obstacle positions as well as the required detection probabilities are shown in Fig. 5. When an obstacle is present between a sensor and a point on the grid that lies within the detection radius of the sensor, the sensor would not be capable of detecting a target at that point (i.e., $p_{\text{miss}} = 1$). The effect of an obstacle being present between a sensor placed at point (i, j) and a point on the grid (x, y) is captured by modifying the entries of the $\mathbf{B}$. The modification is performed by setting to zero the value of $b(x-i, y-j)$ since $b(x-i, y-j)$ corresponds to the logarithmic miss probability (i.e., $\ln(1) = 0$). Simulation results show that the Diff_Deploy algorithm requires 97 sensors to satisfy the requirements whereas the Max_Deficiency algorithm requires 78 sensors only. This corresponds to a savings of approximately 20% in the number of sensors used by the Max_Deficiency algorithm in comparison to the Diff_Deploy algorithm.

VII. CONCLUSION

In this paper, we study the sensor deployment problem. Specifically, given a finite number of sensors, we attempt to

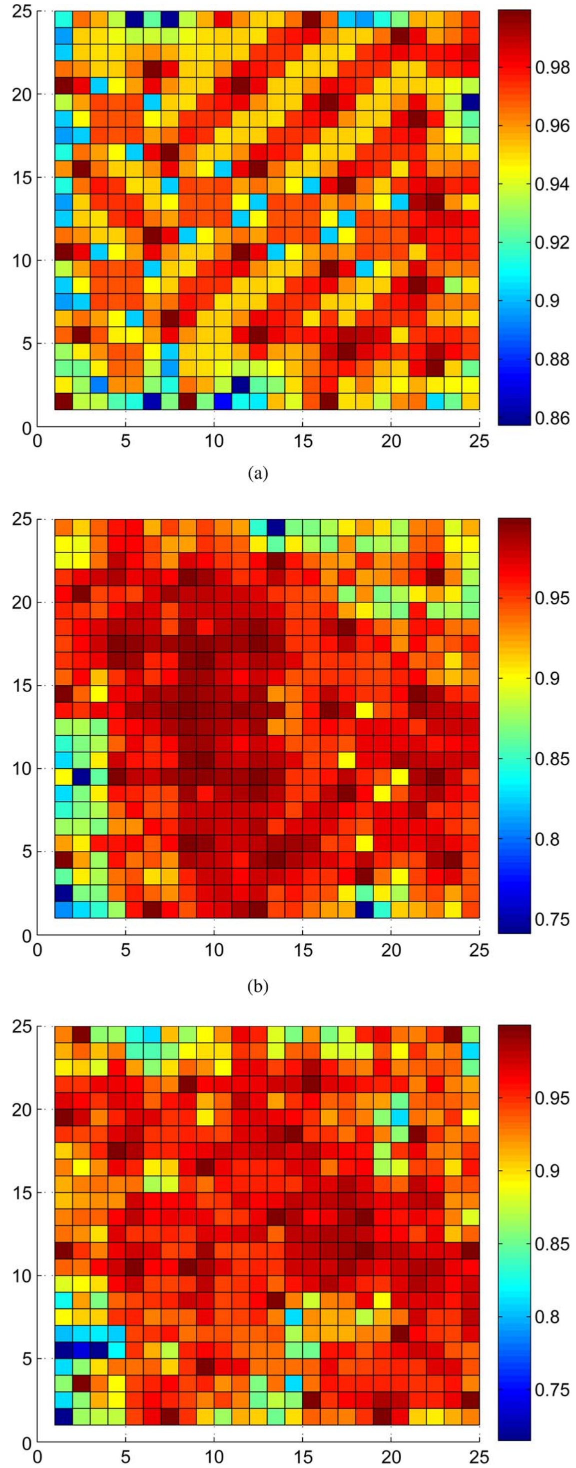

Fig. 3. Achieved detection probability profile. (a) Diff_Deploy. (b) Max_Deficiency. (c) Optimal control based.

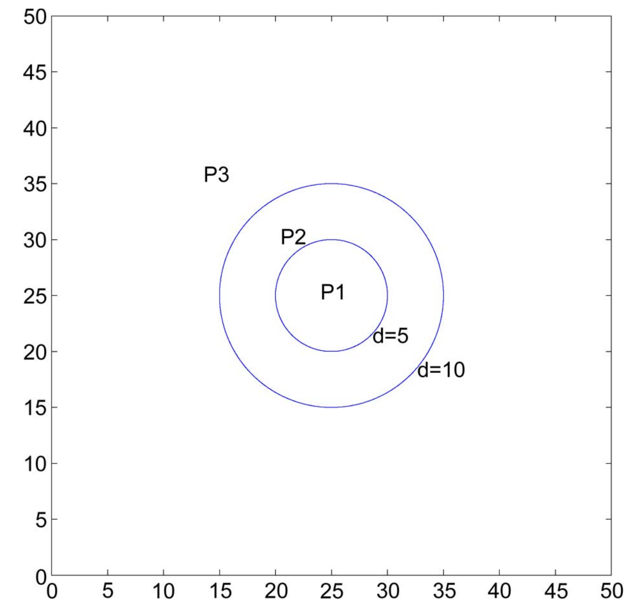

Fig. 4. Required detection probability profile.

TABLE II
SECOND EXPERIMENT RESULTS

Detection Probabilities	Diff_Deploy	Max_Deficiency	%Savings
p1=0.9,p2=0.8,p3=0.6	99	72	27%
p1=0.9,p2=0.7,p3=0.5	83	66	20%
p1=0.8,p2=0.9,p3=0.8	133	121	9%
p1=0.9,p2=0.5,p3=0.7	107	92	14%

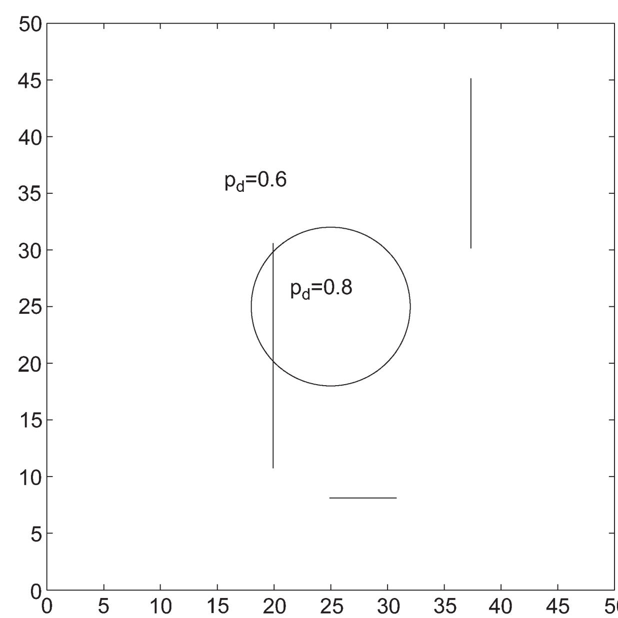

Fig. 5. Required detection probability with obstacles.

determine the locations that the sensors need to be deployed at in order to satisfy the detection requirements in a squared error sense. We express the deployment problem as a dynamical system and formulate the sensor deployment problem as an optimal control problem (linear quadratic regulator). However, the optimal control-based approach is computationally demanding due to the use of the sweep method. Therefore, we introduce a low-complexity alternative called the Max_Deficiency algorithm, which offers comparable performance relative to the optimal control-based approach. Using simulation results, we show that the proposed algorithms outperform existing methods by using 10% to 30% fewer number of sensors to satisfy the detection requirements.

REFERENCES

[1] D. Li, K. Wong, Y. H. Hu, and A. Sayeed, "Detection, classification, and tracking of targets," *IEEE Signal Process. Mag.*, vol. 19, no. 2, pp. 17–29, Mar. 2002.

[2] H. Liu, P. Wan, and X. Jia, "Maximal lifetime scheduling for sensor surveillance systems with K sensors to one target," *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 12, pp. 1526–1536, Dec. 2006.

[3] C.-Y. Chong and S. Kumar, "Sensor networks: Evolution, opportunities, and challenges," *Proc. IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug. 2003.

[4] T. Berger-Wolf, W. E. Hart, and J. Saia, "Discrete sensor placement problems in distribution networks," *J. Math. Comput. Model.*, vol. 42, no. 13, pp. 1385–1396, Dec. 2005.

[5] J. Berry, W. E. Hart, C. E. Phillips, J. G. Uber, and J. Watson, "Sensor placement in municipal water networks with temporal integer programming models," *J. Water Resour. Plan. Manage.*, vol. 132, no. 4, pp. 218–224, Jul./Aug. 2006.

[6] D. Hamel, M. Chwastek, B. Farouk, M. Kam, and K. Dandekar, "A computational fluid dynamics approach for optimization of a sensor network," in *Proc. IEEE Int. Workshop Homeland Security, Contraband Detection Pers. Safety*, 2006, pp. 38–42.

[7] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA: Freeman, 1979.

[8] O. Rahman, A. Razzaque, and C. S. Hong, "Probabilistic sensor deployment in wireless sensor network: A new approach," in *Proc. 9th Int. Conf. Adv. Commun. Technol.*, Feb. 12–14, 2007, vol. 2, pp. 1419–1422.

[9] M. Watfa and S. Commuri, "Optimal 3-dimensional sensor deployment strategy," in *Proc. 3rd IEEE CCNC*, Jan. 8–10, 2006, vol. 2, pp. 892–896.

[10] Y. Zou and K. Chakrabarty, "Uncertainty-aware and coverage-oriented deployment for sensor networks," *J. Parallel Distrib. Comput.*, vol. 64, no. 7, pp. 788–798, Jul. 2004.

[11] J. Zhang, T. Yan, and S. H. Son, "Deployment strategies for differentiated detection in wireless sensor networks," in *Proc. 3rd Annu. IEEE Commun. Soc. Sens. Ad Hoc Commun. Netw. SECON*, Sep. 28, 2006, vol. 1, pp. 316–325.

[12] P. Dorato, C. T. Abdallah, and V. Cerone, *Linear Quadratic Control: An Introduction*. Melbourne, FL: Krieger, 2000.

[13] A. Sage and C. White, *Optimum Systems Control*. Englewood Cliffs, NJ: Prentice-Hall, 1977.

[14] K. Chakrabarty, S. Iyengar, H. Qi, and E. Cho, "Grid coverage for surveillance and target location in distributed sensor networks," *IEEE Trans. Comput.*, vol. 51, no. 12, pp. 1448–1453, Dec. 2002.

[15] S. Shakkottai, S. Srikant, and N. Shroff, "Unreliable sensor grids: Coverage, connectivity and diameter," in *Proc. 22nd Annu. Joint Conf. IEEE INFOCOM*, Mar. 30–Apr. 3, 2003, vol. 2, pp. 1073–1083.

[16] S. Dhillon and K. Chakrabarty, "Sensor placement for grid coverage under imprecise detections," in *Proc. 5th Int. Conf. Inf. Fusion*, Jul. 8–11, 2002, vol. 2, pp. 1581–1587.

[17] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," in *Proc. 20th Annu. Joint Conf. IEEE INFOCOM*, Apr. 22–26, 2001, vol. 3, pp. 1380–1387.

[18] A. Elfes, "Occupancy grids: A stochastic spatial representation for active robot perception," in *Proc. 6th Conf. Uncertainty AI*, 1990, pp. 60–70.

[19] B. D. O. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[20] M. Mahmoud and M. Singh, *Discrete Systems: Analysis, Control and Optimization*. Berlin, Germany: Springer-Verlag, 1984.

[21] V. Sima, *Algorithms for Linear-Quadratic Optimization*. New York: Marcel Dekker, 1996.

Ahmad Ababnah (S'09) received the B.S. degree in electrical engineering from Mutah University, Mutah, Jordan, in 2003, and the M.S. degree in electrical engineering from Arizona State University, Tempe, in 2007. He is currently working toward the Ph.D. degree in electrical engineering at Kansas State University, Manhattan.

In 2007, he joined the Department of Electrical and Computer Engineering, Kansas State University. His research interests include distributed sensor networks, optimal control, and sequential Bayesian filtering.

Balasubramaniam (Bala) Natarajan (S'98–M'02–SM'08) received the B.E degree in electrical engineering from Birla Institute of Technology and Science, Pilani, India, in 1997 and the Ph.D. in electrical degree from Colorado State University, Fort Collins, in 2002.

Since Fall 2002, he has been a Faculty Member in the Department of Electrical and Computer Engineering, Kansas State University, Manhattan, where he is currently an Associate Professor and the Director of the Wireless Communication (WiCom) and Information Processing Research Group. He was also involved in telecommunications research at Daimler Benz Research Center, Bangalore, India, in 1997. He also has two pending patents in adaptive/customized precoder design for MIMO systems. He has numerous journal and refereed conference publications in the wireless communications and signal processing arena. He has published a book titled *Multi-carrier Technologies for Wireless Communications* (Kluwer, 2002) and holds a patent on customized spreading sequence design algorithm for CDMA systems. His research interests include spread spectrum communications, multi-carrier CDMA and OFDM, multi-user detection, cognitive radio networks, sensor signal processing, distributed detection, and estimation and antenna array processing.