Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 145496, 13 pages
http://dx.doi.org/10.1155/2013/145496

Research Article

Optimal Planning of Distributed Sensor Layouts for Collaborative Surveillance

Thomas A. Wettergren and Russell Costa

Naval Undersea Warfare Center, 1176 Howell Street, Newport, RI 02841, USA

Correspondence should be addressed to Thomas A. Wettergren; t.a.wettergren@ieee.org

Received 27 September 2012; Accepted 13 March 2013

Academic Editor: Nadjib Achir

The use of a spatially distributed set of sensors has become a cost-effective approach to achieve surveillance coverage against moving targets. As more sensors are utilized in a collaborative manner, the optimal placement of sensors becomes critical to achieve the most efficient coverage. In this paper, we develop a numerical optimization approach to place distributed sets of sensors to perform surveillance against moving targets over extended areas. In particular, we develop a genetic algorithm solution to find spatial sensor density functions that maximize effectiveness against moving targets, where the surveillance performance of individual sensors is dependent on their absolute position in the region as well as their relative position to both the expected target(s) and any asset that is being protected. The density function representation of optimal sensor locations is shown to provide a computationally efficient method for determining sensor asset location planning. We illustrate the effective performance of this method on numerical examples based on problems of general area surveillance and risk-based surveillance in protection of an asset.

1. Introduction

Target surveillance in large areas is a difficult problem with many challenges; however, due to its importance for military operations it is one that has been studied extensively [1]. In the future, the importance of this problem will only grow as technical advances worldwide create more numerous and capable adversaries. This challenge has created more areas of the world where surveillance assets (sensors) must operate to achieve mission goals of varying scales. The descriptor "large area" is relative to the sensing capability of available (individual) sensors deployed in a specific region against specified targets of interest. A surveillance problem is deemed large area if the sensing capability of an individual sensor is small relative to the area to be searched (covered), in a fixed time scale. For example, problems can be defined in hours, days, weeks, and so forth, depending on tactical mission, and ultimately this will determine scale such as number of required sensors. Military surveillance problems may take the form of covering a bounded region against any intruders (the coverage problem), or may be more specific to covering a region around an asset of interest in order to protect the asset. In both of these situations, the selection of the best from a limited predefined set of surveillance configuration options is the standard practice [2].

Advances in sensor technology have made distributed sensor networks [3–5] a viable candidate technology for performing the military surveillance mission. In order for distributed sensor networks to achieve reasonable surveillance goals, some forms of collaboration must exist amongst the sensors. Historically, this collaboration has been managed in one of two ways. One method has been to partition the search region in such a way so that individual sensors are responsible for their own portion of the region of interest. This partitioning is done *a priori* using algorithms or human judgment to attempt to optimally split up the search effort among available sensors. The collaboration in this approach is limited to occasional reports (amongst sensors or to a central authority) which lead to suboptimal surveillance performance. The other common approach is to again partition the space, but with the emphasis on post processing of detection events. This approach focuses on the reactionary part of the problem (i.e., conditional on the presence and initial detection of a target) and thus, once again is suboptimal in its use of collaboration among sensors. In this work a methodology is developed to plan deployment of distributed sensors which includes a functional dependence on collaboration, as well as an explicit dependence on spatial variation in sensor performance.

With recent technological improvements in automation and communications networking capabilities, there has been an increase in the utilization of collaboration among sensors to perform target surveillance over large areas [6]. The focus of these studies for distributed sensor surveillance has been to spread out a number of sensors and use the spatial distribution of the individual sensors to cover a larger area (much larger than the coverage of any individual sensor) to monitor against intruders [7, 8]. These studies have been primarily for use in networks of sensors that are simple and autonomous in nature but have led to a fresh look at distributed surveillance particularly in the form of postdetection data fusion [9]. Other advances have used sensor repositioning after deployment to improve coverage, such as the use of virtual force algorithms [10] to move randomly deployed sensors to improve the coverage of the sensor network. While related, the problem of sensor network detection and classification algorithm design [11, 12] follows from the positioning of the sensors. We hold that the optimal placement of sensors will benefit from any further improvements gained from the detection and classification process. Similarly, the ability of the surveillance system to track any target of interest is critical to mission performance. Previous efforts in sensor network configuration have examined the positioning of sensors for target tracking applications [13], and it is well recognized that the target tracking performance of adaptively managed sensor networks is heavily dependent on the spatial deployment pattern [14]. In contrast to those efforts, the current paper is focused on the prior problem of maximizing the ability of gaining the initial detection for the surveillance application alone.

In this paper, we optimize distributed sensor configurations to achieve optimal surveillance performance. We utilize objectives based on a prescribed level of collaboration among sensors such that optimization of these objectives results in sensor placement that is optimal with respect to collaboration as well as individual sensor performance. This approach scales well with the number of sensors and, thus, is applicable to the large scale sensor network topologies down to the tactical scales more commonly found in current surveillance (search) problems. It is this latter scale that is the focus of this paper. In this distributed sensing objective, the sensors independently perform target detection and target detection decisions are made by comparing multiple non-collocated detections to check for kinematic consistency, as a form of target classification. If the individual detections are consistent (in spatiotemporal relation) with the anticipated target behavior, then the multiple detections corroborate and the collaborative sensors declare a target present.

The motivation of this work is to utilize a collaboration framework in a formal manner so that with modern computing resources, tactical decision aids can be developed to facilitate the command decisions with respect to collaborative sensors. With the formulation of a numerical objective, more target hypotheses can be considered than notional examination on which current approaches rely. To improve the performance of such a distributed sensor surveillance system, we consider the problem of determining the optimal layout of a group of sensors. Rather than optimizing all of the parameters of a system design, we focus instead on the key component of optimizing the opportunities for multiple sensor detections over time. Such opportunities are a critical first step in the many approaches to collaborative sensing. We consider this optimization of geometrical opportunities for collaboration as a fundamental goal of sensor layout planning, and secondary goals that are particular to a specific form of collaboration are beyond our scope. Our goal is the development of a computationally efficient numerical method that accounts for the geometric and environmental complexity of the problem, while maintaining enough generality to be useful in a variety of scenarios.

In the following, we refer generically to the platform that performs the sensing function as a "sensor," with the implied interpretation that all sensors have an underlying platform that holds them, be it a large manned asset or simply the device's housing. Thus a sensor may be as large as a manned radar or sonar platform, or as small as a simple proximity measurement device. The characterization of these devices, for our application, is given by their expected detection performance against the target of interest, which is presumed to be known from a prior model.

Given a fixed number of sensors, an expected distribution of target behavior, and a model of the sensors' detection performance in the region of interest, we develop an optimization framework that provides a sensor layout (set of sensor positions) for optimal surveillance protection of a region of interest under varying levels of collaboration. In particular, we consider three surveillance problems: (1) the detection of targets that are transiting throughout the region (typical area surveillance), (2) the detection of targets in the region that are far enough away from a high-valued unit (HVU) to provide reaction, and (3) the detection of objects that are weighted by their relative risk to the HVU. In the next section, we provide a mathematical model that accounts for all three of these distributed sensor surveillance problems. This common model has forms for both the cases of independent and of collaborative sensing and, thus provides a framework to study the implications of collaboration in the optimal positioning of sensors. The following develops a genetic algorithm-based optimization framework for optimizing sensor placement under this model. Finally, we conclude with some examples of the optimization to provide a comparison of the sensor layout patterns for various scenarios.

2. Mathematical Model of Distributed Surveillance

A crucial element in utilizing mathematical modeling to find practical solutions to problems such as optimal placement of distributed sensors (distributed assets) is in the formulation and numerical representation of the underlying objective. The formulation of the objective should be an accurate model of the problem that captures all parametric dependencies. In particular, any dependence on tactical parameters such as target behaviors and environmental characteristics requires a method that allows these parameters to be accounted for with varying levels of uncertainty. The numerical calculation of this objective should be a suitable approximation while being as efficient as possible to allow practical use in optimization approaches. The approach we follow is to model all of these dependencies in an integral formulation of expected performance over the search space. This integral form is then integrated with respect to any particular spatial distribution of sensors to arrive at probabilities representing expected surveillance field performance.

The first component in this model of distributed surveillance is the model for target motion (behavior). This model should allow varying levels of constraints on target motion to be of general use in a wide variety of problems. The model developed in this paper assumes the target motion to be Markovian in nature, such that its behavior can be decomposed into a sequence of short time behaviors. This assumption implies that the target motion path $\mathbf{y}(t)$ is effectively modeled by the sequence of intervals $\{[t_0, t_1), [t_1, t_2), \ldots\}$ and the path on the ith interval is $\mathbf{y}_i(t)$. The union of the collection of paths gives the total target path

$$\mathbf{y}(t) = \{\mathbf{y}_i(t) : t_i \leq t < t_{i+1}\}, \quad (1)$$

where each path $\mathbf{y}_i(t) = \mathbf{y}_{\tau,i} + v_i \cdot (t - t_i) \cdot [\cos(\theta_i), \sin(\theta_i)]^T$ represents a path of constant velocity target motion v_i in direction θ_i. Furthermore, each interval has motion parameters $\mathbf{p}_i = (\mathbf{y}_{\tau,i}, v_i, \theta_i)$ which are sampled from known distributions, and the specific values are only dependent on the previous time step, $\mathbf{p}_{i+1} = f(\mathbf{p}_i)$, as opposed to depending on the entire history (this is the Markov assumption). This Markov motion model is regularly assumed in modeling nonreactive targets [15] and is the basis of many Monte Carlo simulation approaches to target modeling [16]. We utilize the model to limit our analysis to optimizing the performance over a fixed, finite time step, where the motion of the target follows constant velocity during the interval of interest and the probability distributions of the motion parameters are all known *a priori*.

For a given interval $[t_i, t_{i+1})$, the probability of collaborative detection is a function of the random variables that describe target motion, as well as the location of the sensing assets and their detection performance. Consider a single given target motion track over this chosen interval. Assume that all of the sensors and the entire target track are contained within the surveillance region $\mathcal{S} \subseteq \mathbb{R}^2$ (we assume the region is large enough that edge effects are negligible). The probability of detecting this track by N_D individual sensor detections (the probability of successfully surveilling the track) is written as [17]

$$P_{\text{ST}}(N_D \geq k) = 1 - \exp(-NP_D\phi) \sum_{m=0}^{k-1} \frac{(NP_D\phi)^m}{m!}, \quad (2)$$

where k is the minimum number of assets (sensors) independently detecting the target required for a collaborative detection of the target, while N is the total number of assets (sensors) in the surveillance region $\mathcal{S}$. The parameter P_D is the detection probability of an individual sensor, defined as constant within a given detection radius R_D. Note that the parameter k is used to define the level of collaboration in this framework. The variable ϕ represents the likelihood of a sensor being within distance R_D of a particular target track path to have an opportunity to detect the target (i.e., being within range of the target at some point during the track history). Explicitly, it is given by

$$\phi(\mathbf{y}_T, v, \theta) = \int_{\Omega_T(\mathbf{y}_T, v, \theta)} f(\mathbf{x})\, d\mathbf{x}, \quad (3)$$

where $f(\mathbf{x})$ is the distribution of sensors in the space, and the region Ω_T is the two dimensional region (defined by a specific target track) comprised of the subset of $\mathcal{S}$ that is within detection radius R_D of the track.

The relationship between target path and the probability of successful search criteria (for $k = 2$) is illustrated in Figure 1. Figure 1(a) shows a notional path through a rectangular search region in the presence of deployed passive sensors with detection circles as shown. A subset of this path is highlighted and magnified in Figure 1(b) to show a constant velocity segment of this path (from the given Markov parameters) and the subset of the sensors which detect this segment (and subsequently the given path). Note that one can view the detection process from a sensor frame of reference, that is, a detection occurs when two or more sensor circles contain the target segment, or through a target frame of reference, where the center of the sensor circles must be within the pill-shaped region (shaded area in Figure 1(b)) to detect the target. It is through this target frame of reference that we efficiently calculate P_{ST} given a distribution on sensor position. In particular, the expressions in (2) and (3) represent the random search [18] of a moving target "seeking" the fixed sensors when the problem is viewed from a target frame of reference. The resultant probability of successful search, P_{SS}, for this time interval is given by marginalizing the probability $P_{\text{ST}}(N_D \geq k)$ over the uncertainty description of the target track as

$$\begin{aligned} P_{\text{SS}}(N_D \geq k) &= \int_0^{2\pi} \int_{v_{\min}}^{v_{\max}} \int_{\mathcal{S}} P_{\text{ST}}(N_D \geq k)\, f_T(\mathbf{y}_T) \\ &\quad \times f_v(v)\, f_\theta(\theta)\, d\mathbf{y}_T\, dv\, d\theta, \end{aligned} \quad (4)$$

where the functions $f_T(\mathbf{y}_T)$, $f_v(v)$, and $f_\theta(\theta)$ are probability density functions (PDF's) for target motion parameters of position, speed, and course, respectively. In addition, by increasing (or decreasing) k, we subsequently increase (decrease) the required level of collaboration among the distributed sensors.

In practice, a target track is successfully found by a collaborative sensor system based on sensors sharing detection information. Thus, it is not only dependent on the requisite

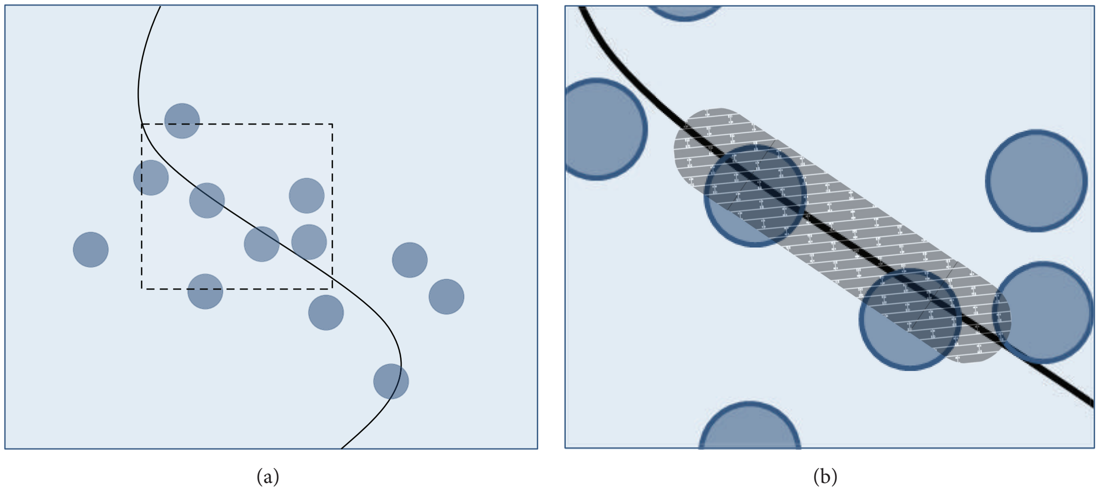


Figure 1: Notional example showing the connection between sensor placement and track coverage. (a) shows the target track path and the location of sensors with their coverage region. (b) is a blowup of the square box drawn in (a), showing a nominal "target pill" region representing a finite-time segment of the target track path.

number of sensors performing successful detections, but it also requires those sensors to communicate their results to other neighboring sensors. In the absence of full communication connectivity amongst the sensors, a graph model of the network node locations is used to assess the overall connectivity of the network in a probabilistic sense. Let $P_{\mathrm{CON}}(f(\mathbf{x}), N, r_c)$ represent the probability of connectivity of a sensor network of N nodes of communication range r_c that are spatially distributed according to the distribution $f(\mathbf{x})$. The probability of connectivity for the network is the probability that there exists some multihop path between any two nodes of the network, thus it is the probability that any network node can communicate with the rest of the network. The computation of $P_{\mathrm{CON}}(f(\mathbf{x}), N, r_c)$ for a fixed number N of nodes can be performed by known methods [19, 20]. Now the operational success of a search operation is the conditional probability of successful search conditioned on network connectivity. Mathematically, this is given by the joint probability expression $P_{\mathrm{SS}} \cdot P_{\mathrm{CON}}$ where P_{SS} is as given in (4). We note that both terms in this joint probability have a dependence on the sensor placement distribution $f(\mathbf{x})$. However, we assume that the sensors in our applications are densely spaced with respect to communications, such that every sensor can communicate reliably with all other nodes within the network. Such an assumption is common in passive detection systems, where the passive detection radius is often much smaller than the reliable communication distance. Thus, for the remainder of this paper, we consider only the case of completely connected networks (i.e., $P_{\mathrm{CON}} = 1$) and, therefore, the objective corresponding to an operational success of search is given by P_{SS} alone. The extension of our optimization technique to problems with limited connectivity is a subject of future work.

In order to numerically calculate the objective (4) over a variety of distributions from flat (uniform within search region $\mathcal{S}$) to highly nonuniform, we represent the sensor density function as a mixture of N_G circular Gaussian functions, as

$$f(\mathbf{x};\mathbf{w}) = \frac{1}{2\pi\sigma^2}\sum_{j=1}^{N_G} w_j \exp\left(-\frac{1}{2\sigma^2}\left(\mathbf{x}-\mathbf{x}_j\right)^T\left(\mathbf{x}-\mathbf{x}_j\right)\right), \quad (5)$$

with modal weights w_j, constant modal variance σ^2, and fixed (spaced equidistant in $\mathcal{S}$) positions $\mathbf{x}_j$, which has been shown to be a useful model for density approximation in many applications [21]. The number (and subsequent spacing) of the Gaussian modes required, as well as the value of the variance parameter σ, are chosen using a heuristic rule. The heuristic [22] is based on the flexibility of the overall density representation; that is, the relationship between the width of a Gaussian mode and the modal spacing; should be such that a wide variety of function forms of $f(\mathbf{x})$ can be represented. After experimentation on many test problems, we determined an appropriate relationship to be

$$N_G = \left(\left\lceil \frac{L}{3R_D} \right\rceil\right)^2, \quad (6)$$

where L is the length of the search region $\mathcal{S}$ along one dimension (assuming that $\mathcal{S}$ is square). The variance parameter is given as $\sigma = 2R_D$ for $R_D \ll L$. As R_D increases relative to L, this parameter should be made smaller relative to R_D; however, for the scale in this paper the given heuristic is applicable.

We note that P_{SS} provides a measure on the ability of multiple collaborative sensors to detect the target in a manner consistent with the spatiotemporal relationship of target motion and sensor position [17]. This is commonly referred to as track-before-detect and is an effective technique for reducing false alarms in distributed detection applications through collaboration defined by the aforementioned spatiotemporal relationship. It is also a method of multisensor filtering of contacts that is commonly used within many data fusion methods. Thus P_{SS} is, in general, a measure of track coverage in a surveillance region. However, as the length of the time interval tends to zero, P_{SS} becomes the more familiar metric of area coverage, that is, coverage independent of target motion. In that context, the expression in (2) is simply the composite area coverage provided by a set of independent sensors provided that their locations are randomly sampled from a common spatial distribution function $f(\mathbf{x})$.

3. Numerical Model

To optimize the placement of assets, the integrals in (3) and (4) must be evaluated with respect to changes in the sensor density function $f(\mathbf{x})$. In both cases these integrals do not have closed form solutions, and thus, must be evaluated numerically. Note first that the evaluation of the integral in (3) is significantly simplified by the representation of sensor density function defined in (5). Namely, through the fixed position and circular variance, the integral can be separated by mode (as a sum of the integrals of each mode) and in dimension (the two-dimensional integral can be separated by independence into the product of two one-dimensional integrals) independent of the modal weights. The latter property allows much of the computation to be done once, prior to entering the optimization, simplifying subsequent objective evaluations. This improvement in efficiency makes the optimization practical on standard desktop computers with no special coding requirements. Further simplification can be made by noting that for constant R_D (sensor performance independent of position in $\mathcal{S}$) the region of integration Ω_T given a target trajectory is a pill-shaped region with area $2R_D v\tau + \pi R_D^2$. This region can be well approximated by a rectangle of equal area for $v\tau \gg R_D$ which allows the integral in (3) to be evaluated using standard error functions commonly used for evaluation of integrals involving Gaussian functions [22]. The implementation utilized in this paper allows for spatial variability of R_D by including an additional step in which the equivalent rectangle is replaced by a series of rectangles (a partitioning) which approximate the track-dependent region (within which sensors have an opportunity to detect the target) by interpolation of an underlying R_D function. The number of segments that each track is partitioned (i.e., the number of rectangles) is determined *a priori* and depends on the extent of the spatial variability of R_D in the search region $\mathcal{S}$.

The mathematical detail required to evaluate (3) consists of the following. Consider an arbitrary target track of fixed length as defined above. Define an arbitrary point along this track $\mathbf{y}_{t_i}$ and a set $A_{t_i} = \{\mathbf{x} : \|\mathbf{y}_{t_i} - \mathbf{x}\| \le R_D(\mathbf{x})\}$, which is the set of all points in $\mathcal{S}$ from which an arbitrary sensor can detect a target at the specified point along the target track (with probability P_D). Then the general form of the "pill" shaped region of integration can be written as $\Omega_T = \bigcup_{i=1}^{\infty} A_{t_i}$. Next, define an approximation to this region as $\Omega^m = \bigcup_{j=1}^{m} A_{t_j}$ where t_j, $j = 1, \ldots, m$ refer to m equally spaced points spanning the length of the arbitrary track and construct disjoint sets B_{t_j} from sets A_{t_j} by the recursion $B_{t_j} = A_{t_j} - U^{j-1}$ given $U^k = \bigcup_{j=1}^{k} A_{t_j}$ and $U^0 = \emptyset$ (the empty set). Then the integral in (3) can be approximated as

$$\begin{aligned}\int_{\Omega_T} f(\mathbf{x})\,d\mathbf{x} &= \int_{\mathcal{S}} I_{\Omega_T}(\mathbf{x})\,f(\mathbf{x})\,d\mathbf{x}\\ &\approx \int_{\mathcal{S}} I_{\Omega^m}(\mathbf{x})\,f(\mathbf{x})\,d\mathbf{x}\\ &= \sum_{j=1}^{m}\int_{\mathcal{S}} I_{B_{t_j}}(\mathbf{x})\,f(\mathbf{x})\,d\mathbf{x},\end{aligned} \tag{7}$$

where $I_A(\mathbf{x})$ is the set indicator function of the set A.

Assuming that the integral in (3) is well approximated by the above and that the function P_{ST} (as in (2)) changes slowly over the target track parameters in $\mathcal{S}$, the numerical evaluation of the integral in (4) can be done simply, provided that the PDFs of the target track parameters are continuously differentiable and slowly changing over their support. Since this is true for the examples in this paper, then in this work the integral is evaluated by gridding the track parameters (and associated weights) evenly over the track parameter space. This allows the triple integral in (4) to be well approximated by a triple sum weighted by the product of the corresponding values of the PDFs over the target parameter grid. The numerical evaluation of the sum as shown above provides a robust computation of the required integral of $f(\mathbf{x})$ over the region Ω_T.

4. Formulation of Optimization

The problem of optimum deployment of assets for collaborative multisensor surveillance, restricted to the mathematical model in (4), is one of maximizing search effectiveness in a fixed region. From an optimal planning perspective, the problem is one of maximizing the likelihood of achieving the surveillance mission, where the mission can take on various forms. We represent this as a minimization problem of the form

$$\min_{f} P_{MF}, \tag{8}$$

where P_{MF} is the probability of mission failure and $f = f(\mathbf{x})$ is a function representing the sensor distribution over the region $\mathcal{S}$. In practice, this mission failure may take a variety of forms, but we are primarily concerned with the joint probability of not detecting a target within our surveillance region, combined with the risk associated with that target's presence. In particular, for the small time interval of interest which determines the path interval of interest, we have

$$\begin{aligned}P_{MF} &= \int_0^{2\pi}\int_{v_{\min}}^{v_{\max}}\int_{\mathcal{S}} (1 - P_{ST})\,\psi(\mathbf{y}_T)\,f_T(\mathbf{y}_T)\\ &\qquad \times f_v(v)\,f_\theta(\theta)\,d\mathbf{y}_T\,dv\,d\theta\\ &= \int_{\mathcal{S}} f_T(\mathbf{y}_T)\,\psi(\mathbf{y}_T)\,d\mathbf{y}_T\end{aligned}$$

$$- \int_0^{2\pi} \int_{v_{\min}}^{v_{\max}} \int_{\mathcal{S}} P_{\mathrm{ST}}(\mathbf{y}_T) \cdot [f_T(\mathbf{y}_T) \psi(\mathbf{y}_T)] \times f_v(v) f_\theta(\theta) \, d\mathbf{y}_T \, dv \, d\theta, \tag{9}$$

where $\psi(\mathbf{y}_T)$ is a consequence (risk) function. The consequence function is dependent on the location of target track $\mathbf{y}_T$ and is defined to measure the relative risk posed by various tracks over that of others (such as those in proximity to an HVU, if that is the intention of the surveillance region). The first integral in (9) does not depend on the choice of the sensor location density $f(\mathbf{x})$, so it does not impact the optimization leading to an effective minimization objective of

$$J = - \int_0^{2\pi} \int_{v_{\min}}^{v_{\max}} \int_{\mathcal{S}} P_{\mathrm{ST}}(\mathbf{y}_T) \cdot [f_T(\mathbf{y}_T) \psi(\mathbf{y}_T)] f_v(v) f_\theta(\theta) \, d\mathbf{y}_T \, dv \, d\theta. \tag{10}$$

The optimization problem of (8) is now given in the form

$$\min_f J(f) \tag{11}$$

for the objective functional J given in (10).

If all target locations and tracks are equally important, then the consequence function $\psi(\mathbf{y}_T)$ is necessarily equal to unity, leading to $J = -P_{\mathrm{SS}}$ (see (4)). In such cases, the optimization problem of (11) is equivalent to

$$\max_f P_{\mathrm{SS}}. \tag{12}$$

We seek the $f(\mathbf{x})$ which maximizes the probability of successful search, leading to a density function from which sensors will then be placed [22]. When our goal is more specific, that is for protection of an HVU, the consequence function $\psi(\mathbf{y}_T)$ is used to represent the relative risk of various target tracks, and the solution of the same optimization problem ((10) and (11)) yields the solution of minimizing the expected risk to the HVU. Thus, the optimization problem of (10) and (11) is generically utilized as the asset layout optimization problem, with the understanding that a variety of specific problems are addressed by varying the form of the consequence function.

To compute the objective functional J as shown in (10), the target motion distribution parameters must be known, as well as the effective sensor performance P_D and R_D, which are generally functions of location in the region. In this work, we adopt a notional model for the spatial dependence of sensor performance in a rectangular region of interest, depicted in Figure 2. As in the previously defined model, sensor performance is measured by a spatially dependent radius of detection $R_D(\mathbf{x})$ which corresponds to a fixed probability of detection P_D (note that in Figure 2 we show $R_D(\mathbf{x})$ normalized to the size of the region L, where the R_D corresponds to a value of $P_D = 0.9$). In practice, sensor performance predictions such as these can be formulated using historical information on the environment and are

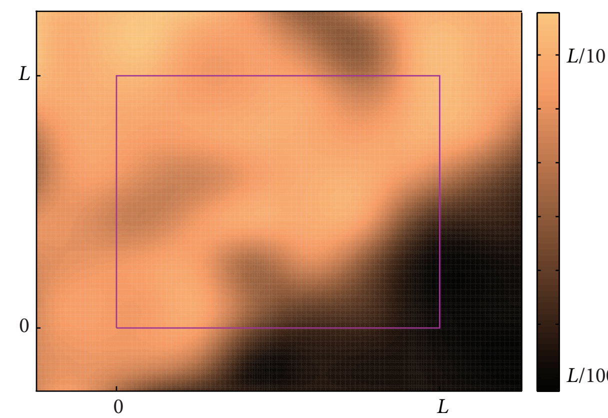


Figure 2: Sensor detection range map for the example problems. The region drawn in the center represents the surveillance region for sensor placement.

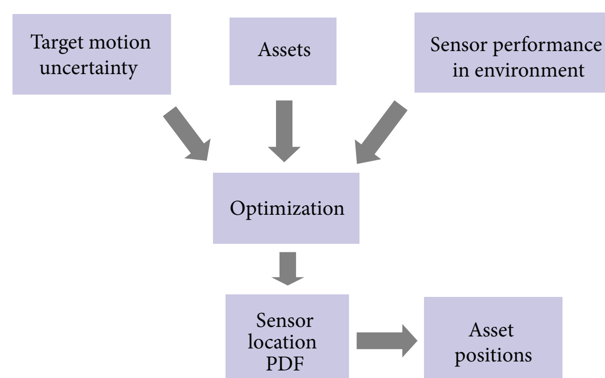


Figure 3: Sensor optimization algorithm flow diagram.

common in passive acoustic sensor applications, both in air [23] and undersea [24] domains. The sensor performance and number of sensors provide the necessary inputs for computation of P_{ST} (as in (2)) for any specified target track. When combined with the target motion distribution parameters and consequence function, they provide a complete description of the objective J for any distribution $f(\mathbf{x})$ of sensor locations. Figure 3 shows a functional description of the overall approach, where it becomes clear that the "inputs" to the optimization, that is *a priori* knowledge of target, environment, and asset availability, are utilized to find the optimal distribution of assets. The specific placement of the individual sensor assets from the distribution is done using a sampling procedure from the resulting distribution, leading to a placement map for the surveillance region.

5. Numerical Procedure for Optimization

Recall from (5) that the modal positions representing the sensor distribution are fixed and thus the optimal distribution with respect to the surveillance objective (9) is parameterized only through the modal weights. Thus the numerical objective for optimization becomes

$$\min_{\mathbf{w}} J(\mathbf{w})$$
$$\text{subject to} \sum_{j}^{N_G} w_j = 1, \quad 0 \leq w_j \leq 1, \forall j. \tag{13}$$

We implement a genetic algorithm to perform the optimization defined in (13). The genetic algorithm cannot be run to any guarantee of convergence [25] but is rather run to a prescribed number of generations (iterations). If a theoretically optimal result is desired, the result of the genetic algorithm may be used as the starting point for a nonlinear program (NLP). These stages are complementary in that the genetic algorithm is insensitive to its start and will make significant progress toward a global solution but is devoid of satisfactory stopping criteria (i.e., no guaranteed final convergence). The NLP on the other hand can be quite sensitive to its starting solution but theoretically proven to converge to a local maximum [26]. Thus one goal in the design of this approach is for the potential use of the genetic algorithm to initialize an NLP in the neighborhood of a globally optimal solution, and thus we can attain convergence to a global maximum, if desired.

Genetic algorithms operate on a discrete set of parameters in the form of a binary string. The parameters in this problem are the weights $\{w_j\}_{j=1}^{N_G}$ representing the sensor distribution $f(\mathbf{x})$ in $\mathcal{S}$. In the numerical implementation, each weight parameter w_j is represented by a four-bit binary string, with N_G individual Gaussian modes for the representation in (5). Thus, the string length is $4N_G$.

A genetic algorithm starts with some random values of the parameters of interest represented in the form of a binary string as described above. A set of these strings is produced which is referred to as a *population*. This type of algorithm is an iterative search where iterations are referred to as *generations*. At each generation (iteration), the binary strings which make up the population undergo a series of operations. Thus, starting with a randomly generated population, each string is evaluated by the objective function J returning a value corresponding to each string. Typically, the value of the objective is mapped into a more convenient form (to improve scaling) referred to as fitness [27]. However, in this implementation fitness is set to the evaluated objective J, as this quantity is well scaled.

A standard form of genetic algorithm [27] was implemented with each generation consisting of three genetic operations defined in the evolutionary vernacular as *selection*, *mating*, and *mutation*. These operations utilize the fitness associated with each binary string in the population to pseudorandomly select the best (with respect to the objective J) parameter combinations, randomly combine the selected strings, and apply some random perturbations to the resulting strings, respectively. Specifically, the selection approach utilized, referred to as "roulette," selects binary strings by first scaling the fitness of the population members to sum to unity. Next, the cumulative sum of the fitness is calculated, creating an interval $(0, 1)$, with subintervals proportional to the fitness of each binary string. A random uniform number is then generated and the subinterval in which the number falls determines the string that is selected. Thus, a string (population member) with high fitness, relative to other strings, will be selected with high probability while one with low fitness will be selected with low probability. In this implementation, the string with the highest fitness (at each generation) is kept as a *survivor*; that is, the best string gets passed on to the next generation unchanged. Therefore, $N_{\text{pop}} - 1$ strings are selected to pass to the next generation (where N_{pop} is the fixed number of strings in a population), and these strings make up what is referred to as the *mating pool*. In the next phase, strings in the mating pool are randomly (without regard to fitness) paired up and then randomly combined to create new parameter strings. This operation is called *crossover*. Crossover consists of randomly breaking two strings (at the same point) and then combining the leading part of one with the trailing part of the other. Finally, each of these newly formed strings is passed to the *mutation* operation which flips bits (i.e., change 0 to 1, or vice versa) within each string randomly at some specified (*a priori*) probability. This is essentially a random perturbation of the parameters meant to avoid premature convergence to local minima. Once these operations are complete, the new strings are grouped with the survivor string and these strings become the new population passed on to the next generation (iteration). This process is repeated for some predefined set of generations. From numerical experimentation on a variety of problems, a population size of 100 run over 200 generations was suitable for producing meaningful results for the numerical examples in this paper.

On completion of the genetic algorithm, the optimal sensor density is obtained and the sensors are then placed using a numerical sampling procedure. The procedure consists of a sequential (conditional) sampling where an asset location is selected (among a grid of possible locations) which maximizes the relative entropy between the prior form of the PDF $f(\mathbf{x})$ (discretized and normalized to sum to unity, in order to convert to a probability mass function) and the posterior probability mass function (PMF) calculated by selecting the asset. The relative entropy between two PMFs is written as [28]

$$D(p_1 || p_0) = \sum_{\mathbf{s} \in \mathcal{S}} p_1(\mathbf{s}) \log\left(\frac{p_1(\mathbf{s})}{p_0(\mathbf{s})}\right) \tag{14}$$

and represents a measure of divergence of one PMF relative to the other. The conditional sampling procedure used to place sensors from $f(\mathbf{x})$ treats individual sensor placement as a Bayes recursion where a unique posterior is generated by a positional-dependent likelihood update, defined as corresponding to a possible sensor location. The procedure starts with the definition of two grids (uniformly spaced points in $\mathcal{S}$), written as $\mathbf{z}_i$, $i = 1, \ldots, m$ and $\mathbf{v}_j$, $j = 1, \ldots, n$ where $\mathbf{z}_i$ represents discretely sampled points of $f(\mathbf{x})$, and $\mathbf{v}_j$ represents all possible sensor locations for placement. Next, the prior is calculated from the final form (after optimization) of $f(\mathbf{x})$ as

$$p_0(\mathbf{z}_i) = \frac{I_{\mathbf{z}_i}(\mathbf{x}) f(\mathbf{x})}{\sum_{i=1}^{m} I_{\mathbf{z}_i}(\mathbf{x}) f(\mathbf{x})}, \quad (15)$$

where

$$I_{\mathbf{z}_i}(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} = \mathbf{z}_i \\ 0, & \mathbf{x} \neq \mathbf{z}_i \end{cases} \quad (16)$$

is the indicator function. The posterior probability resulting from selecting (placing) a sensor at position $\mathbf{v}_j$ is defined as

$$\pi^i(\mathbf{z}) = \begin{cases} \dfrac{p_0(\mathbf{z})}{\psi_j}, & \mathbf{z} \in \overline{B}_j \\ \dfrac{\alpha \cdot p_0(\mathbf{z})}{\psi_j}, & \mathbf{z} \in B_j, \end{cases} \quad (17)$$

where $\mathcal{S} = B_j \cup \overline{B}_j$, $B_j \cap \overline{B}_j = \emptyset$ (i.e., a disjoint partitioning of $\mathcal{S}$ with respect to sensor position $\mathbf{v}_j$) and $B_j = \{\mathbf{z} : \|\mathbf{z} - \mathbf{v}_j\|_2 \leq R_D(\mathbf{v}_j)\}$ is a ball of spatially dependent radius R_D (with respect to constant P_D) centered at point $\mathbf{v}_j$. The sensor coefficient $\alpha = 1 - P_D$ plays the role of decreasing mass within the radius of the placed sensor, while the sensor-dependent normalizing constant is written as

$$\psi_j = \sum_{\mathbf{z} \in \overline{B}_j} p_0(\mathbf{z}) + \sum_{\mathbf{z} \in B_j} \alpha \cdot p_0(\mathbf{z}). \quad (18)$$

This normalizing constant is required so that each posterior probability is a proper PMF (i.e., sums to unity over its support). The posterior with respect to all possible sensor grid points is calculated as in (15), and a sensor is placed at a specified position, by choosing the posterior which maximizes the relative entropy with respect to the prior. This is formalized as

$$\pi^* = \arg\max_j D\left(\pi^j \| p_0\right), \quad (19)$$

where this process is repeated in a sequential fashion to place all N sensors. Upon the placement of each sensor, the posterior with respect to the chosen location acts as the prior for placing the next sensor.

6. Numerical Examples

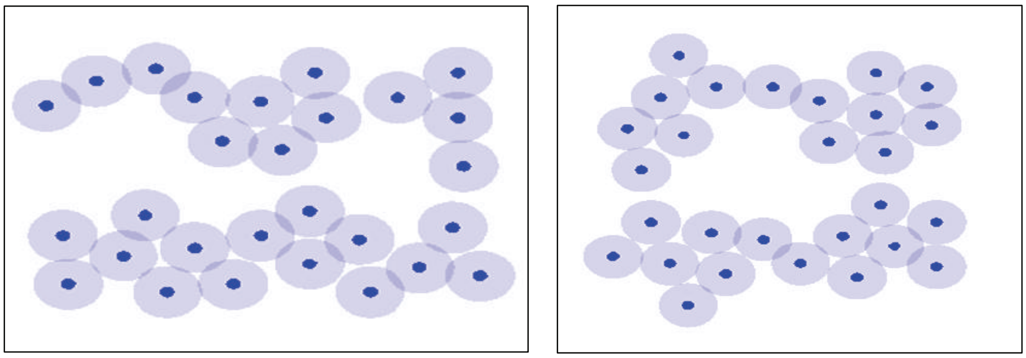
The problem of sensor placement as defined above depends on many factors. These factors can be primarily sensor dependencies from environmental variability [29] or can be dominated by other factors such as target behavior [30]. To illustrate these dependencies, we present several numerical examples. Throughout the examples, we consider the number of available sensor assets N to be fixed. The planning problem is to place these sensors optimally in a square planar region $\mathcal{S}$ of size $L \times L$. For these examples, the optimality criteria are to maximize the probability of surveillance mission success, corresponding to minimization of the probability of mission failure P_{MF}.

As an introductory example, and to demonstrate the utility of the optimization approach, we define a nominal environment (constant detection range given by $R_D = L/15$) with target parameters for which intuitive solutions exist. We seek to optimally place $N = 28$ sensors, such that we obtain the maximum P_{SS} (corresponding to minimizing P_{MF}) with a requirement of at least two sensor reports during a time interval τ. The target is assumed to be traveling in a known fixed heading (assumed north) at constant speed v over the fixed time interval τ (where $v\tau = L/2$) with a start position randomly distributed within the search region. For this problem, the expected optimal placement is a "barrier" formation perpendicular to the target course [22]. In particular, due to random starting positions, we should observe a two-line barrier perpendicular to the target course. Figure 4(a) illustrates the optimization results from this problem, where we see that the barrier structure results, as expected. A second nominal example considers a similar problem but with target heading defined as random. In this case, the optimization result, shown in Figure 4(b), produces a sensor layout in a "box-like" structure, which may not be intuitively obvious but has been shown to be optimal [22]. These nominal examples show the dependence that the target behavior has on the optimal sensor layouts.

In a typical approach to deployment of sensors under limited knowledge of the environment, it is reasonable to consider some nominal sensor detection performance. However, given current environmental modeling capabilities, we assume that sensor detection performance can be provided to some acceptable level of fidelity. Figure 2 shows a $1.5L \times 1.5L$ region containing the region of interest, defined by the inner box. The underlying color map depicts sensor coverage as a function of position within the region. The sensor performance is limited by environmental factors that are beyond our control, and the optimization seeks to maximize mission performance (minimizing P_{MF}) in surveillance of the given region with a limited number of sensors. In particular, for the region in Figure 2, the lower right part of the region exhibits a sharp dropoff in individual sensor coverage.

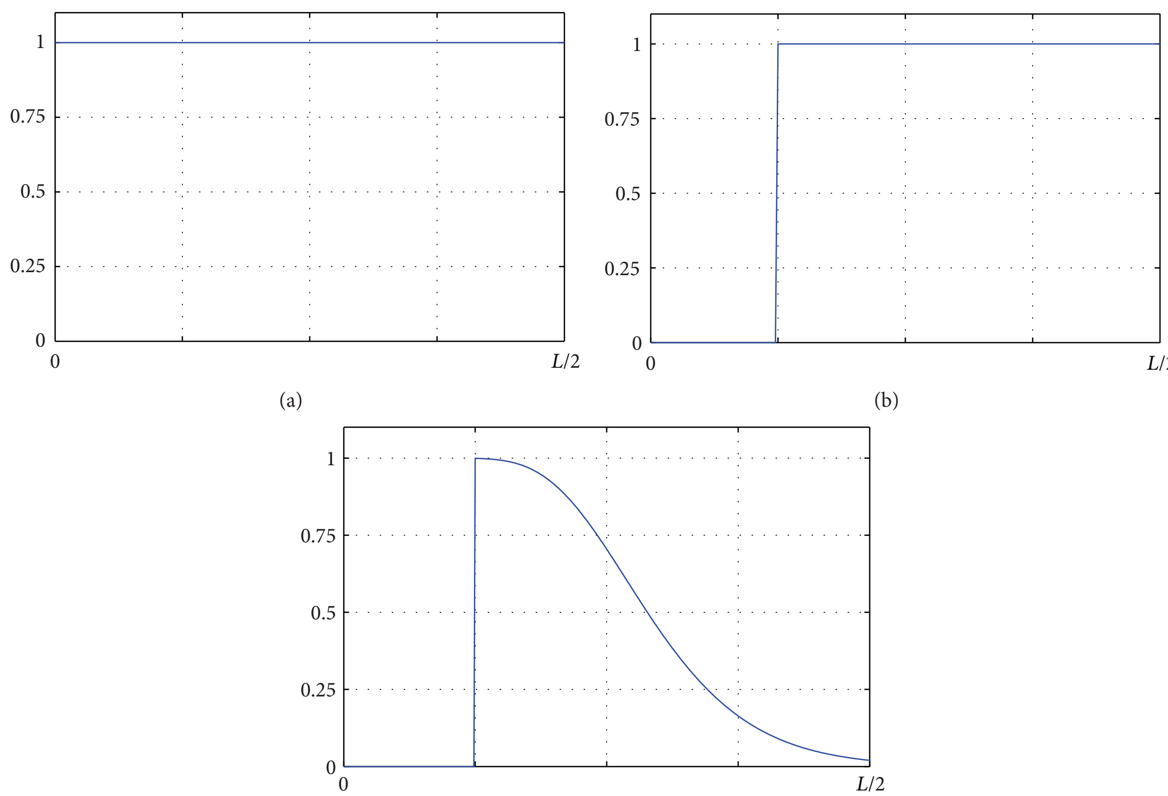
An additional input to the optimization is the characterization of target behavior. The numerical examples that follow were produced assuming that target position and heading are uniformly random within the search region $\mathcal{S}$. That is, all reference track positions (previously defined as $\mathbf{y}_T$) are equally likely. Furthermore, assume that the target of interest travels at a fixed speed v over time intervals of length τ. This defines a track length of $v\tau$, which is given as $v\tau = L/8$ for these numerical examples. The track length is scalable over varying combinations of speed and time (as it is simply the product of the two) and represents *a priori* knowledge of the target of interest, which will result in increased surveillance performance over that of situations where there is very little known (and thus can be assumed) of the anticipated target behavior.

In Figure 5, we illustrate three example consequence (risk) functions $\psi(\mathbf{y}_T)$ of interest. The first function shown in Figure 5(a) is the nominal unity function that is equivalent to the problem of optimizing cumulative probability of detection (see (12) and the surrounding discussion). The second



(a) (b)

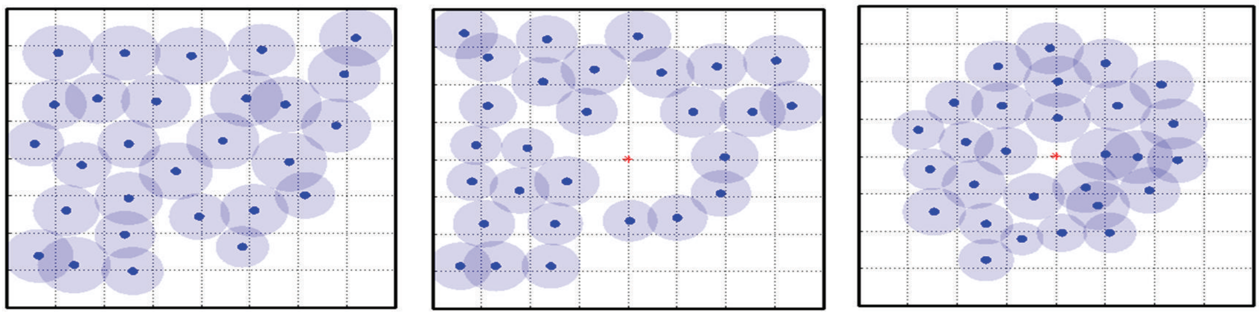
Figure 4: Optimal sensor placement for surveillance of fixed speed target in nominal environment. (a) is for a target with a known course; (b) is for a target with an unknown course.



(a) (b) (c)

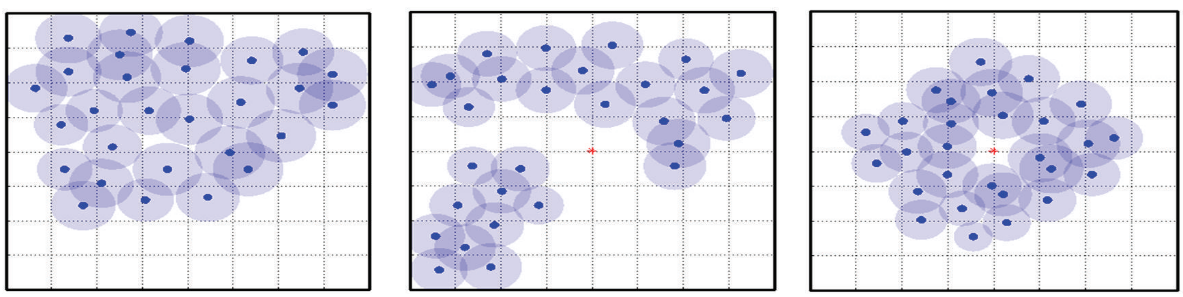
Figure 5: Consequence (risk) function $\psi(\mathbf{y}_T)$ versus range for each of the three example cases.

consequence function shown in Figure 5(b) represents a protection problem for an HVU, whereby any targets within a certain distance to the HVU are too close to provide a surveillance response, and, thus, provide zero surveillance risk, with all others providing nominal risk. This may seem counterintuitive, to have zero risk closest to the HVU, but the point here is to maximize surveillance performance, and this case illustrates the situation in which the surveillance mission is no longer operational when targets are too close to the HVU. Alternatively, if one were to weight the consequence very high near the HVU, an obviously optimal solution is to only try to detect those targets and ignore all targets that are not directly in proximity to the HVU, which is not desired if the risk is already passed. The third consequence function, shown in Figure 5(c), is perhaps the most operationally relevant, in that it incorporates the features of the second case along with degradation in risk for targets further from the HVU. In this case, the risk degradation follows a log-normal function, as described by [31]. Such a consequence is representative of scenarios in which there is a greater



(a) (b) (c)

Figure 6: Optimal sensor distributions for each of the three consequence functions of Figure 5 for scenarios with no cooperation between sensors. Circle size represents the detection range of the sensor (which is a function of position).



(a) (b) (c)

Figure 7: Optimal sensor distributions for each of the three consequence functions of Figure 5 for scenarios with $k = 2$ cooperation between sensors. Circle size represents the detection range of the sensor (which is a function of position).

importance to detect targets closer to the HVU, up to a point at which they are so close that response becomes impractical. Specifically, the log-normal consequence function takes the form

$$\psi(\mathbf{y}_T) = \begin{cases} 0, & \|\mathbf{y}_T - \mathbf{x}_0\| < r_0 \\ \dfrac{1}{2}\left[1 - \operatorname{erf}\left[\dfrac{\ln(\|\mathbf{y}_T - \mathbf{x}_0\|/\alpha)}{\sqrt{2}\beta}\right]\right], & \|\mathbf{y}_T - \mathbf{x}_0\| \geq r_0 \end{cases} \tag{20}$$

for an HVU at location $\mathbf{x} = \mathbf{x}_0$ with a minimal response distance r_0. The parameters α, β in (20) are shape parameters that control the slope and taper of the log-normal consequence function. These three example consequence functions illustrate various applications of the consequence function $\psi(\mathbf{y}_T)$ to show how seemingly different scenarios are solved using the same field optimization approach.

Figure 6 illustrates the results of the optimization process applied to the three consequence functions of Figure 5 for a scenario with $N = 28$ sensors performing noncollaborative surveillance. By noncollaborative surveillance, we consider the sensors to behave completely autonomously ($k = 1$ in (2)), and extended coverage is obtained only through the effective spacing of the individual sensors with respect to the target prior information, that is, since there is no collaboration between sensors, the surveillance relies only upon individual sensors to detect a target if present. In Figure 6, along with the sensor positions, we include opaque circles corresponding to the coverage capability radius R_D of each sensor. The circle size varies according to the local sensing capabilities attributable to the local environmental conditions, as shown in Figure 2. The effect for the first consequence function (for unity risk), as shown in Figure 6(a), is to place the sensors somewhat evenly to best cover the requirement of single sensor coverage in the field. Fewer sensors are located where there is lower detection capability (lower right corner) since the additive coverage is small. For the second consequence function, note that there are no sensors placed near the HVU (see Figure 6(b)), as expected. When compared to the first consequence function, note that the sensors are still spread evenly but now pushed slightly closer to the edges, in order to still cover as much of the area as possible. The third consequence function provides a different type of optimal configuration, as shown in Figure 6(c). In this case, the sensors tend to encircle the HVU in an annulus, as the annular region is the region of highest consequence if detections are missed. This effect appears more significant than the effect of avoiding the low coverage in the lower right corner, and more sensors are added to the lower right section of the annular region to make up for the lower individual sensor coverage. Note that each of these results were created from the same optimization procedure, with the only distinction between the three cases being the specific form of the consequence function $\psi(\mathbf{y}_T)$.

Table 1: Comparison of optimal and nominal values of the objective J for the distributions shown in Figures 6 and 7.

	Objective value for consequence function a		Objective value for consequence function b		Objective value for consequence function c	
	Uniform placement	Optimal placement	Uniform placement	Optimal placement	Uniform placement	Optimal placement
Noncollaborative Sensors	0.46	0.60	0.45	0.55	0.52	0.80
Collaborative Sensors	0.14	0.28	0.14	0.28	0.18	0.60

In practical situations with many sensors, there is performance enhancement opportunity through the use of collaboration [32, 33]. Historically, such problems are solved using optimal processing strategies given a fixed location of sensors [34, 35]. However, the optimization framework developed herein permits the optimization of sensor placements for a given level of collaboration. For instance, the parameter k in (2) may be adjusted to represent the number of sensors that must concurrently detect a target over the time interval of interest τ. Any detections that are spatially or temporally isolated will not count towards the probability P_{ST} used as the performance objective, as they are likely false positives. Recall that the requirement of multiple detections need not occur simultaneously, only over the time interval of interest. Thus, the performance objective cannot be translated into a simple geometrical overlap requirement, that is, a goal in which maximal overlap is sought. In fact, since this objective depends on target track parameters which have spatiotemporal features, there are many scenarios for which non-intuitive patterns of sensors will be optimal. In particular, as complexity (from such factors as environmental sensitivity or higher levels of collaboration) is added, results formed through intuition become less likely to approach optimal, reinforcing the need for an optimization framework which can factor in these complexities.

To illustrate the impact of multiple sensor collaboration on the optimal patterns, the examples of Figure 6 are repeated with a requirement of $k = 2$ detections to occur over the time interval of interest. In this case the goal is to optimally deploy the same sensors in the same variable environment, but we now require two separate sensor detections ($k = 2$) over the previously defined time interval τ. The resulting optimal patterns for the three consequence functions are shown in Figure 7. Comparison with Figure 6 shows that the increased detection requirement coupled with the relatively short target track length results in a more clustered approach to the deployment. For the third consequence function the deployment pattern has only subtle differences compared to Figure 6. This is attributed to the effect of having more than a suitable number of sensors for covering the annular region of primary interest.

In Table 1 we show the numerical values of the performance objectives for each of the scenarios presented in Figures 6 and 7. These objective values are also compared to the equivalent objective values obtained with uniform placement patterns of the assets for each situation. Observe that in each case the optimization approach resulted in better performance in the objective J than for the uniform distribution, as expected. In these examples the $N = 28$ sensors represent a sparse coverage with respect to the search region $\mathcal{S}$, particularly for general surveillance (consequence function a) and for cases requiring multiple detections. This sparsity explains some of the general trends seen in the results. For instance, for both consequence functions a and b there is little or no difference between the results for collaborating and independent sensors. This is because the reduction in the search space due to the presence of the HVU is not significant with respect to the level of sensor coverage sparsity. However, the coverage numbers increase significantly for consequence function c, where the form of the consequence function $\psi(\mathbf{y}_T)$ increases the spatial dependence of the objective with respect to the position of the HVU. Overall the increased coverage due to optimization is much more significant for collaborative sensors than for independent sensors. This is due to the added sensitivity of sensor placement when using collaboration based on spatiotemporal target dependence.

An important byproduct of these numerical results is that for a number of diverse surveillance missions, a common optimization procedure can be utilized for positioning sensors to either meet specific performance criteria, or to get the best performance possible. This can be applied in two ways, the obvious one being as a predeployment tool for positioning sensors for a specific mission, the other being a guide to repositioning sensors to react to a change in mission. In either case these examples show that through proper modeling of the problem, optimal positioning of sensor assets can be achieved, without resorting to costly simulations. In fact, the results attained for these examples were produced with a per case computation time of approximately 20 minutes on a Pentium IV 3 GHz processor with code implemented in MATLAB.

7. Conclusion

We have developed an optimization approach to place distributed sets of sensors to collaboratively perform surveillance against moving targets over extended areas. In particular, a genetic algorithm solution was provided to find the spatial sensor density functions that maximize effectiveness against moving targets. These density function representations provide a computationally efficient method for determining sensor locations for planning and were applied to situations with environmentally induced sensor spatial variability and varying forms of target risk. By illustrating the effective performance of our method on problems of general area surveillance and risk-based surveillance in protection of an asset, we have shown how the general technique applies to seemingly dissimilar problems. The numerical solutions that were obtained were shown to compare favorably against nominal layouts of sensors in the scenarios that were examined. Future work includes the extension of this method to problems with limited network connectivity between the sensor nodes.

Acknowledgments

This work has been supported by the Office of Naval Research code 321MS and by the In-house Laboratory Independent Research program of the Naval Undersea Warfare Center.

References

[1] D. H. Wagner, W. C. Mylander, and T. J. Sanders, *Naval Operations Analysis*, Naval Institute Press, Annapolis, Md, USA, 1999.

[2] S. Olariu and J. V. Nickerson, "Protecting with sensor networks: perimeters and axes," in *Proceedings of the Military Communications Conference (MILCOM '05)*, vol. 3, pp. 1780–1786, October 2005.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[4] D. Estrin, D. Culler, K. Pister, and G. Sukhatme, "Connecting the physical world with pervasive networks," *IEEE Pervasive Computing*, vol. 1, no. 1, pp. 59–69, 2002.

[5] H. Qi, S. S. Iyengar, and K. Chakrabarty, "Distributed sensor networks—a review of recent research," *Journal of the Franklin Institute*, vol. 338, no. 6, pp. 655–668, 2001.

[6] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, and K. K. Saluja, "Sensor deployment strategy for detection of targets traversing a region," *Mobile Networks and Applications*, vol. 8, no. 4, pp. 453–461, 2003.

[7] S.S. Dhillon and K. Chakrabarty, "Sensor placement for effective coverage and surveillance in distributed sensor networks," in *Proceedings of the Wireless Communications and Networking Conference*, vol. 3, pp. 1609–1614, March 2003.

[8] P. K. Biswas and S. Phoha, "Self-organizing sensor networks for integrated target surveillance," *IEEE Transactions on Computers*, vol. 55, no. 8, pp. 1033–1047, 2006.

[9] M. F. Duarte and Y. H. Hu, "Vehicle classification in distributed sensor networks," *Journal of Parallel and Distributed Computing*, vol. 64, no. 7, pp. 826–838, 2004.

[10] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization in distributed sensor networks," *ACM Transactions on Embedded Computer Systems*, vol. 3, no. 1, pp. 61–91, 2004.

[11] E. H. Aboelela and A. H. Khan, "Wireless sensors and neural networks for intruders detection and classification," in *Proceedings of the International Conference on Information Networking (ICOIN '12)*, pp. 138–143, 2012.

[12] A. Arora, P. Dutta, S. Bapat et al., "A line in the sand: a wireless sensor network for target detection, classification, and tracking," *Computer Networks*, vol. 46, no. 5, pp. 605–634, 2004.

[13] S. Martínez and F. Bullo, "Optimal sensor placement and motion coordination for target tracking," *Automatica*, vol. 42, no. 4, pp. 661–668, 2006.

[14] K. Hadi and C. M. Krishna, "Management of target-tracking sensor networks," *International Journal of Sensor Networks*, vol. 8, no. 2, pp. 109–121, 2010.

[15] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms, and Software*, Wiley-Interscience, 2001.

[16] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House, 2004.

[17] T. A. Wettergren, "Performance of search via track-before-detect for distributed sensor networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 1, pp. 314–325, 2008.

[18] B. O. Koopman, *Search and Screening: General Principles with Historical Applications*, Pergamon Press, 1980.

[19] M. Desai and D. Manjunath, "On the connectivity in finite ad hoc networks," *IEEE Communications Letters*, vol. 6, no. 10, pp. 437–439, 2002.

[20] A. Ghasemi and S. Nader-Esfahani, "Exact probability of connectivity in one-dimensional ad hoc wireless networks," *IEEE Communications Letters*, vol. 10, no. 4, pp. 251–253, 2006.

[21] G. McLachlan and D. Peel, *Finite Mixture Models*, John Wiley & Sons, 2000.

[22] T. A. Wettergren and R. Costa, "Optimal placement of distributed sensors against moving targets," *ACM Transactions on Sensor Networks*, vol. 5, no. 3, article 26, pp. 1–25, 2009.

[23] R. J. Kozick, B. M. Sadler, and D. K. Wilson, "Signal processing and propagation for aeroacoustic sensor networks," in *Frontiers in Distributed Sensor Networks*, S. S. Iyengar and R.R. Brooks, Eds., CRC Press LLC, 2004.

[24] C. Ferla and M. B. Porter, "Receiver depth selection for passive sonar systems," *IEEE Journal of Oceanic Engineering*, vol. 16, no. 3, pp. 267–278, 1991.

[25] G. Rudolph, "Convergence analysis of canonical genetic algorithms," *IEEE Transactions on Neural Networks*, vol. 5, no. 1, pp. 96–101, 1994.

[26] R. Fletcher, *Practical Methods of Optimization*, John Wiley & Sons, 2nd edition, 1987.

[27] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley Longman, 1989.

[28] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley-Interscience, 1991.

[29] R. Stolkin, L. Vickers, and J. V. Nickerson, "Using environmental models to optimize sensor placement," *IEEE Sensors Journal*, vol. 7, no. 3, pp. 319–320, 2007.

[30] S. A. Musman, P. E. Lehner, and C. Elsaesser, "Sensor planning for elusive targets," *Mathematical and Computer Modelling*, vol. 25, no. 3, pp. 103–115, 1997.

[31] J. S. Przemieniecki, *Mathematical Methods in Defense Analyses*, American Institute of Aeronautics and Astronautics, Reston, Va, USA, 3rd edition, 2000.

[32] J. F. Chamberland and V. V. Veeravalli, "Decentralized detection in sensor networks," *IEEE Transactions on Signal Processing*, vol. 51, no. 2, pp. 407–416, 2003.

[33] S. Ferrari, "Track coverage in sensor networks," in *Proceedings of the American Control Conference*, pp. 2053–2059, Minneapolis, Minn, USA, June 2006.

[34] R. Niu, P. K. Varshney, and Q. Cheng, "Distributed detection in a large wireless sensor network," *Information Fusion*, vol. 7, no. 4, pp. 380–394, 2006.

[35] D.E. Penny, "Multi-sensor management for passive target tracking in an anti-submarine warfare scenario," in *IEE Colloquium on Target Tracking: Algorithms and Applications*, vol. 3, pp. 1–5, 1999.