

Robust Deployment of Dynamic Sensor Networks for Cooperative Track Detection

Kelli A. C. Baumgartner, *Member, IEEE*, Silvia Ferrari, *Senior Member, IEEE*, and Thomas A. Wettergren, *Senior Member, IEEE*

Abstract—The problem of cooperative track detection by a dynamic sensor network arises in many applications, including security and surveillance, and tracking of endangered species. Several authors have recently shown that the quality-of-service of these networks can be statically optimized by placing the sensors in the region of interest (ROI) via mathematical programming. However, if the sensors are subject to external forcing, such as winds or currents, they may be rapidly displaced, and their quality-of-service may be significantly deteriorated over time. The novel approach presented in this paper consists of placing the sensors in the ROI based on their future displacement, which can be estimated from environmental forecasts and sensor dynamic models. The sensor network deployment is viewed as a new problem in dynamic computational geometry, in which the initial positions of a family of circles with time-varying radii and positions are to be optimized subject to sets of algebraic and differential equations. When these equations are nonlinear and time-varying, the optimization problem does not have an exact solution, or global optimum, but can be approximated as a finite-dimensional nonlinear program by discretizing the quality-of-service and the dynamic models with respect to time. Then, a near-optimal solution for the initial sensor positions is sought by means of sequential quadratic programming. The numerical results show that this approach can improve quality-of-service by up to a factor of five compared to existing techniques, and its performance is robust to propagated modeling and deployment errors.

Index Terms—Cooperative, coverage, current, deployment, detection, network, ocean, optimization, sensors, sonobuoy, target, track, tracking, velocity field.

I. INTRODUCTION

THE problem of cooperative track detection by a dynamic sensor network arises in many applications, such as, monitoring of oceanic [1] or atmospheric features [2], tracking of endangered species [3] or marine fish [4], and in tracking-and-detecting potential security threats, such as underwater vehicles or divers [5]. Distributed fields of passive sensors can surveil large regions of interest at moderate cost and for prolonged

periods of time [6]–[8]. However, it has long been recognized in practice that if the sensors are deployed in a dynamic environment, such as a water body or the atmosphere, they can be rapidly displaced by currents or winds, and experience a significant loss in coverage and tracking performance [9], [10]. This paper presents a novel sensor deployment approach in which the quality-of-service of the sensor network over a desired period of time is formulated as an integral objective function of the sensors' positions and ranges, and is optimized subject to the sensor network dynamics using a direct-shooting method. By this approach, the quality-of-service of nonmaneuverable sensor networks can be significantly improved and their operability prolonged by several days, without any changes to the sensor design or instrumentation.

The ability of a sensor network to detect target tracks depends on its area coverage and on its track coverage, both of which have received considerable attention in the literature [11]–[17]. *Area coverage* is defined as the union of the areas representing the sensors' fields-of-view (FOV), divided by the area of the region of interest (ROI) [15]–[17]. *Track coverage* refers to the network's ability to detect target tracks in the ROI by obtaining multiple independent detections at different moments in time [11]–[14]. The problem of *track detection*, which was first considered in [11], is of interest in cost-effective collaborative sensor networks (e.g., low-cost, passive sensors) that are deployed to search for a moving target. Each sensor is designed with a limited autonomous detection capability, and is subject to frequent false alarms. Therefore, in order to report reliable target detections, the information from multiple independent detections is combined with a spatio-temporal model of the target track, in an approach known as track-before-detect [18]. In this approach, a confident network-level detection decision is made only after a target track is estimated and, subsequently, tracking information is provided concurrently with detection reports. By choosing a suitably sized ROI, targets can be assumed to move at a constant speed and heading, and tracks can be modeled as straight lines [11]–[14]. Then, *track coverage* is defined as a measure of the tracks detected by the sensor network, divided by a measure of all possible tracks through the ROI [13].

Previous studies considered the problem of placing or repositioning a sensor network performing cooperative track detection in the ROI by static optimization [11], [13]–[15], [19]. However, when the sensor network is dynamic its quality-of-service is a function of time, and must be optimized subject to the sensor network's equation of motion. This paper shows that by formulating area and track coverage as instantaneous functions of the

Manuscript received November 04, 2008; revised February 19, 2009 and April 14, 2009; accepted April 27, 2009. Current version published July 29, 2009. This work was supported in part by the Office of Naval Research Young Investigator Program (Code 321). The associate editor coordinating the review of this paper and approving it for publication was Dr. Ricardo Gutierrez-Osuan.

K. A. C. Baumgartner was with the Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC 27708 USA. She is now with the Mission Analysis Branch, Analex Corporation, Subsidiary of QinetiQ North America, Brook Park, OH 44142 USA (e-mail: kacb@alumni.duke.edu).

S. Ferrari is with the Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC 27708 USA (e-mail: sferrari@duke.edu).

T. Wettergren is with the Naval Undersea Warfare Center, Newport, RI 02841 USA (e-mail: t.a.wettergren@ieee.org).

Digital Object Identifier 10.1109/JSEN.2009.2025836

sensors' ranges and positions the network's quality-of-service can be integrated with respect to time. Then, the sensor network deployment can be viewed as a new problem in dynamic computational geometry in which the initial positions of a family of circles with time-varying radii and positions are to be optimized subject to static and dynamic constraints comprised of sets of nonlinear algebraic and differential equations, respectively.

The optimal deployment (OD) approach is presented in Section IV, and is applied to an ocean sensor network comprised of drifting sonobuoys equipped with passive acoustic sensors in Section VI. Sonobuoys provide a popular means for collection of remote ocean data over large regions of interest and, possibly, over large time scales [20]. However, current-induced drifting has been shown to rapidly displace networks of sonobuoys, deteriorating their quality-of-service especially near coastlines or in regions of strong ocean currents [9], [20]. Recently, a model was developed and experimentally validated in [9] to simulate the drift of free-floating sonobuoys within a deployed network in the ocean from the current velocity field. In Section VI-A, this sonobuoy model is combined with a model of the current velocity field forecast to obtain a nonlinear time-varying equation of motion for the sensor network. In Section VI-B, a model of effective sensor range as a function of time and position is obtained by geoacoustic inversion, using a Bayesian network model trained with an underwater acoustic range-dependent parabolic equation [21].

The effectiveness of the OD approach is demonstrated by simulating networks of sonobuoys that are placed in the ROI near the New Jersey coast, using real current measurements obtained by a Coastal Ocean Dynamics Applications Radar (CODAR) at the Coastal Ocean Observation Lab of Rutgers University (COOL) [22]. The robustness of the OD solutions is investigated via Monte Carlo (MC) simulation, using the stochastic robustness analysis (SRA) approach reviewed in Section V. The numerical results in Section VII show that the quality-of-service of drifting sensor networks can be significantly improved compared to static optimization [13], grid [23], and random [24], [25] deployments. Additionally, the optimal sensor network performance is robust to propagated uncertainties, such as, deployment errors, and errors in the current velocity field and effective sensor range models.

II. BACKGROUND ON TRACK COVERAGE AND GEOMETRIC TRANSVERSALS

The quality-of-service of sensor networks performing cooperative track detection, referred to as track coverage, was recently formulated as a geometric transversals problem in [13] and [26]. In many tracking applications, the FOV of each sensor can be modeled as a circle, referred to as binary omnidirectional sensor model, and targets can be assumed to move at constant speed and heading along straight tracks, inside a suitably sized ROI [11], [18], [27], [28]. Under these assumptions, a target track that is cooperatively detected by k sensors in a network of n sensors is a line transversal of the family of n circles representing the sensors' FOV. A set of geometric objects in \mathbb{R}^d is said to have a j -transversal when all the objects are simultaneously intersected by a j -dimensional flat or translate of a linear subspace [29]. A line transversal with $j = 1$, $d = 2$, and $k \geq 1$,

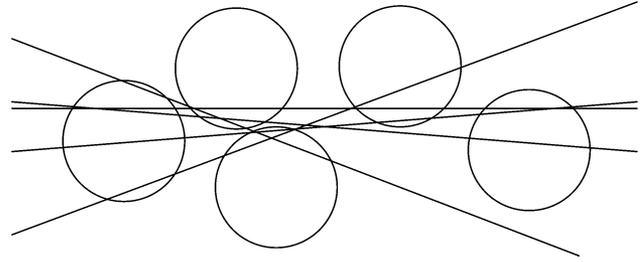


Fig. 1. Example of line transversals for a family of $n = 5$ circles and $k = 3$ (adapted from [29, p. 182]).

also referred to as *stabber*, is a straight line that intersects at least k members of a family of objects. For example, stabbers of a family of five circles, with $k = 3$, are shown in Fig. 1. While considerable attention has been given to establishing the necessary and sufficient conditions for the existence of transversals, algorithms for finding j -transversals or for constructing a space of transversals have been obtained only in a few special cases [29]. In [30], an algebraic decision tree methodology was developed to find a single stabber for a translates family of n line segments in \mathbb{R}^2 , or n equal circles in \mathbb{R}^2 . In [13] convex analysis was used to construct cone representations of sets of stabbers for a nontranslates family of n circles in \mathbb{R}^2 .

Using the cone representation, a Lebesgue measure corresponding to the cone's opening angle can be assigned to a set of stabbers, and computed as a function of the circle's radius and position [13]. By assuming that each target moves at constant speed and heading and may be detected at any time inside the ROI, the temporal nature of the target motion can be ignored. Then, given a set of n static sensors, a measure of the set of stabbers or tracks detected by the sensor network can be obtained and, possibly, optimized with respect to their positions in the ROI [13]. It was also shown in [13] that the measure of all possible tracks through the ROI is a constant that can be determined from the ROI's dimensions and, thus, may be ignored in the optimization. In applications where there is no prior knowledge of the target tracks, track coverage is directly proportional to the probability of track detection, through a known constant [13]. Thus, when the track's parameters are uniformly distributed over their ranges, optimizing track coverage is equivalent to optimizing the probability of track detection in the ROI, as proven in [13]. In the next section, the problem of track detection for mobile sensor networks is formulated by adjoining the sensors' equation of motion to the optimization of the area coverage and track coverage integrated over time.

III. PROBLEM FORMULATION

This paper presents a novel sensor deployment problem in which the area and track coverage of a drifting sensor network are maximized over time by determining the optimal initial positions of the sensors in the ROI. The sensor network is deployed to cooperatively detect moving targets in the ROI, $\mathcal{A} \subset \mathbb{R}^2$, represented by a fixed rectangle $[0, L_1] \times [0, L_2]$, over a fixed time interval $\Delta T = [t_0, t_f]$. It is assumed that L_1 , L_2 , and ΔT are chosen such that the targets move at a constant speed and heading in \mathcal{A} . Then, every target track in \mathcal{A} can be represented by a ray or half line $\mathcal{R}_\theta(b_y)$ with heading θ , and y -intercept b_y .

Since there is no prior knowledge of the target tracks, the track parameters θ and b_y have a uniform probability distribution over their range. Thus, assuming all targets traverse \mathcal{A} during the time interval ΔT , the track coverage of the network at any time $t \in \Delta T$ is directly proportional to its probability of track detection, as shown in [13]. Let the FOV of the i th sensor in the network, located at $\mathbf{x}_i(t) = [x_i(t) \ y_i(t)]^T \in \mathcal{A}$ at time t , be a closed and bounded subset of the ROI, denoted by $C_i(t) \subset \mathcal{A}$. The geometry and dimensions of C_i depend on the physical principles underlying the measurement process, and on the local environmental conditions. In this paper, $C_i(t) = C_i[\mathbf{x}_i(t), r_i(t)]$ is assumed to be a circle of time-varying radius $r_i = g(\mathbf{x}_i)$, centered at \mathbf{x}_i . Then, the i th sensor in the network detects a target track $\mathcal{R}_\theta(b_y)$ at time t if and only if the track intersects its FOV, i.e., $\mathcal{R}_\theta(b_y) \cap C_i(t) \neq \emptyset$, and $\mathcal{R}_\theta(b_y)$ is a stabber of $C_i(t)$ [29]. Let the FOVs of all n sensors in the network be denoted by the set $S(t) = \{C_1(t), \dots, C_n(t)\}$. Then, a target track that is cooperatively detected by k sensors in the network during the time interval ΔT is a stabber of $S(t)$. Since the target tracks are uniformly distributed and may intersect members of $S(t)$ at any time during ΔT , the set of all stabbers of $S(t)$ for all $t \in \Delta T$ represents the set of all tracks detected by the network.

Then, as shown in Section IV, the quality-of-service of a network performing cooperative track detection can be expressed as an integral objective function

$$J = \phi[\mathbf{x}(t_f)] + \int_{t_0}^{t_f} \mathcal{L}[\mathbf{x}(t), \mathbf{r}(t), t] dt \quad (1)$$

that represents the weighted sum of track and area coverage objectives over time, as a function of the sensors' positions, $\mathbf{x} \equiv [\mathbf{x}_1^T \dots \mathbf{x}_n^T]^T$, and ranges, $\mathbf{r}(t) \equiv [r_1(t) \dots r_n(t)]^T$. The terminal cost

$$\phi(\mathbf{x}(t_f)) = W_\phi [\mathbf{x}_f - \mathbf{x}(t_f)]^T [\mathbf{x}_f - \mathbf{x}(t_f)] \quad (2)$$

can be used to specify desired final sensor positions, \mathbf{x}_f . When the sensors are placed in a water body or in the atmosphere, their positions change over time due to the current or wind velocity field in \mathcal{A} . Using environmental models and forecasts, the sensors' dynamics can be modeled by an equation of motion comprised of n independent systems of ordinary differential equations

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} \mathbf{f}_1[\mathbf{x}_1(t), t; \mathbf{x}_{1_0}] \\ \vdots \\ \mathbf{f}_n[\mathbf{x}_n(t), t; \mathbf{x}_{n_0}] \end{bmatrix} \equiv \mathbf{f}[\mathbf{x}(t), t; \mathbf{x}_0] \quad (3)$$

where \mathbf{x} is the sensor network state, and $\mathbf{x}(t_0) = \mathbf{x}_0 \equiv [\mathbf{x}_{1_0}^T \dots \mathbf{x}_{n_0}^T]^T$ denotes the sensor network's initial conditions. In this paper, the sensor network's equation of motion (3) is nonlinear and time-variant (NLTV), and is obtained using the methodology in Section VI-A. Since the sensors have no on-board actuators, their trajectories and performance in \mathcal{A} depend solely on their initial positions, and \mathbf{x}_0 constitutes the network *decision vector*. Therefore, J is to be optimized with respect to \mathbf{x}_0 , subject to the network dynamics (3), and the model of effective sensor range $r_i = g(\mathbf{x}_i)$.

The resulting dynamic optimization problem constitutes a new problem in dynamic computational geometry ([31, p. 1117],) pertaining to a family of circles with time-varying radii that move in \mathcal{A} according to an NLTV equation of motion (3). As shown in the next section, an approximate solution can be determined by transcribing the optimization of the integral objective function (1) into a nonlinear program (NLP) that is solved by sequential quadratic programming (SQP) to determine the optimal decision vector, \mathbf{x}_0^* . The methodology, presented in the next section, is applied to an ocean sensor network in Section VI. Since the models and the execution of the sensors' deployment are both subject to errors, the robustness of the OD approach presented in the next section is verified in Section VII-B using the method reviewed in Section V.

IV. METHODOLOGY

The quality-of-service of sensor networks performing cooperative track detection, represented by track and area coverage, has received considerable attention in the literature [13]–[17]. These previous studies considered the problem of placing a static sensor network in \mathcal{A} such that its ability of obtaining k independent detections is maximized. However, if the sensor network is dynamic, e.g., is subject to displacement or drift in its environment, after the initial placement its coverage can be significantly decreased over time (Section VII). The methodology presented in this paper overcomes this limitation by optimizing an integral objective function, representing the total quality-of-service of the network over the time interval ΔT , subject to the equation of motion (3). The objective function is defined as a weighted sum of instantaneous track coverage, $\mathcal{T}_{\mathcal{A}}^k$, and instantaneous area coverage, A_C , integrated over ΔT

$$J = \phi[\mathbf{x}(t_f)] + \int_{t_0}^{t_f} \{W_{\mathcal{T}} \mathcal{T}_{\mathcal{A}}^k[\mathbf{x}(t), \mathbf{r}(t)] + W_A A_C[\mathbf{x}(t), \mathbf{r}(t)]\} dt \quad (4)$$

where $W_{\mathcal{T}}$ and W_A are user-defined constants used to weigh the two objectives. The instantaneous track and area coverage functions for a moving sensor network are derived in this section, and the integral objective function (4) is optimized using the direct-shooting method presented in Section IV-A.

The geometric-transversals approach presented in [13], and reviewed in Section II, is adopted here to derive $\mathcal{T}_{\mathcal{A}}^k$ as a function of time. This approach leads to a track coverage measure that is a function of the sensors' effective ranges and positions and, therefore, can be integrated with respect to time and optimized subject to a dynamic constraint, (3). Let the inertial xy -frame of reference be placed along two sides of \mathcal{A} , with the origin $(0, 0)_{xy}$ at the lower left corner, such that all target tracks traverse \mathcal{A} in the positive orthant \mathbb{R}_+^2 . As shown in Appendix I, the set of stabbers of a circle $C_i(t) = C_i[\mathbf{x}_i(t), r_i(t)]$ in \mathbb{R}^2 , with y -intercept b_y , can be represented by the cone that is finitely generated by the following unit vectors

$$\begin{aligned} \hat{\mathbf{h}}_i(b_y, t) &= \begin{bmatrix} \cos \alpha_i(t) & -\sin \alpha_i(t) \\ \sin \alpha_i(t) & \cos \alpha_i(t) \end{bmatrix} \frac{\mathbf{v}_i(t)}{\|\mathbf{v}_i(t)\|} \\ &= \mathbf{Q}_i^+[\alpha_i(t)] \hat{\mathbf{v}}_i(t) \end{aligned} \quad (5)$$

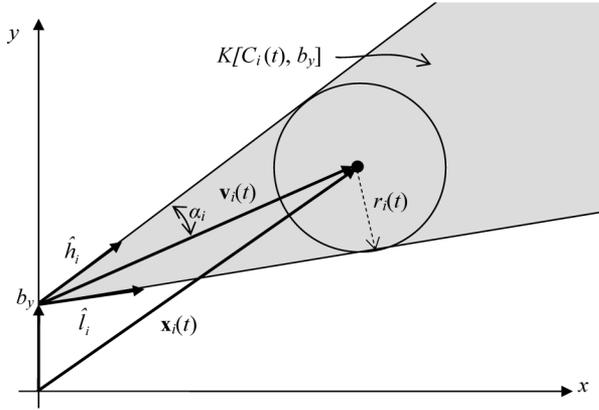


Fig. 2. Coverage cone $K[C_i(t), b_y]$ of a sensor located at \mathbf{x}_i , and generated by the unit vectors $\hat{\mathbf{h}}_i$ and $\hat{\mathbf{h}}_i$ at time t .

and

$$\begin{aligned} \hat{\mathbf{h}}_i(b_y, t) &= \begin{bmatrix} \cos \alpha_i(t) & \sin \alpha_i(t) \\ -\sin \alpha_i(t) & \cos \alpha_i(t) \end{bmatrix} \frac{\mathbf{v}_i(t)}{\|\mathbf{v}_i(t)\|} \\ &= \mathbf{Q}_i^-[\alpha_i(t)] \hat{\mathbf{v}}_i(t) \end{aligned} \quad (6)$$

where $\mathbf{v}_i(t) \equiv \mathbf{x}_i(t) - [0 \ b_y]^T$, and $\alpha_i(t) = \sin^{-1}(r_i(t)/\|\mathbf{v}_i(t)\|)$. A cone $K \subset \mathbb{R}^2$ is said to be finitely generated by two unit vectors when it contains all of their linear combinations [32]. Thus, the *coverage cone* of the i th sensor with origin at the intercept b_y is defined as

$$\begin{aligned} K[C_i(t), b_y] &\equiv \text{cone}(\hat{\mathbf{h}}_i, \hat{\mathbf{h}}_i) \\ &= \{\mathbf{u} \mid \mathbf{u} = c_1 \hat{\mathbf{h}}_i(b_y, t) + c_2 \hat{\mathbf{h}}_i(b_y, t), c_1, c_2 \geq 0\} \end{aligned} \quad (7)$$

and is illustrated in Fig. 2. Like the unit vectors $\hat{\mathbf{h}}_i$ and $\hat{\mathbf{h}}_i$, $K[C_i(t), b_y]$ is a function of the intercept b_y , and of the i th sensor's position and range at time t .

The same unit vectors are used to determine the stabbers of families of k nontranslate circles, representing the FOVs of k distinct sensors. We order all unit vectors in \mathbb{R}^2 based on the orientation of the xy -frame. Here, two vectors $\mathbf{u}_i, \mathbf{u}_j \in \mathbb{R}^2$ are ordered as $\mathbf{u}_i \prec \mathbf{u}_j$ if when these vectors are translated such that their origins coincide, and \mathbf{u}_i is rotated through the smallest possible angle to meet \mathbf{u}_j , this orientation is in the same direction as the orientation of the xy -frame [33]. Then, the set of stabbers with y -intercept b_y can be obtained for a family of k circles in \mathbb{R}^2 , as shown by the following result:

Proposition 4.1: The set of all stabbers of a family of k circles $S_k(t) = \{C_1(t), \dots, C_k(t)\}$, through $y = b_y$, is contained by the finitely generated cone

$$K_k[S_k(t), b_y] = \text{cone}(\hat{\mathbf{I}}^*, \hat{\mathbf{h}}^*) \quad (8)$$

where

$$\begin{aligned} (\hat{\mathbf{I}}^*, \hat{\mathbf{h}}^*) &= (\hat{\mathbf{h}}_i, \hat{\mathbf{h}}_j) \ \iota, j \in I_{S_k}, \text{ such that} \\ \hat{\mathbf{h}}_i &\succeq \hat{\mathbf{h}}_i(b_y, t), \hat{\mathbf{h}}_j \preceq \hat{\mathbf{h}}_j(b_y, t), \hat{\mathbf{h}}_i \prec \hat{\mathbf{h}}_j, \quad \forall i, j \in I_{S_k} \end{aligned} \quad (9)$$

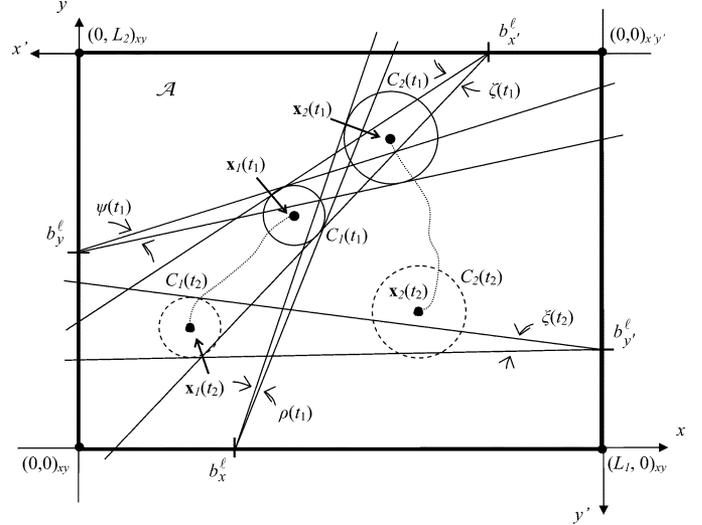


Fig. 3. The $k = 2$ -coverage cones of a family of $n = 2$ sensors are shown at two moments in time, t_1 and t_2 , for a sample intercept (indexed by l) on each axis.

and I_{S_k} denotes the index set of $S_k(t)$. If $\hat{\mathbf{h}}_i \succeq \hat{\mathbf{h}}_j$, then $K_k[S_k(t), b_y] = \emptyset$.

A proof is provided in Appendix II. The cone $K_k[S_k(t), b_y]$ contains the set of tracks detected by a family of k sensors at time t , and is referred to as k -coverage cone. The opening angle ψ of the k -coverage cone is a Lebesgue measure over the set of line stabbers of $S_k(t)$ [13], and is obtained by the cross product

$$\psi[S_k(t), b_y] = \sin^{-1} \|\hat{\mathbf{I}}^* \times \hat{\mathbf{h}}^*\| = H(D_{\iota j}) \sin^{-1}(D_{\iota j}) \quad (10)$$

with

$$\begin{aligned} D_{\iota j} &\equiv \frac{1}{w_i^2 w_j^2} \{ [x_i p_i + (y_i - b_y) r_i] [x_j r_j + (y_j - b_y) p_j] \\ &\quad - [x_j p_j - (y_j - b_y) r_j] [(y_i - b_y) p_i - x_i r_i] \} \end{aligned} \quad (11)$$

where $w_i(t) \equiv \|\mathbf{v}_i(t)\| = \sqrt{x_i^2(t) + [y_i(t) - b_y]^2}$ and $p_i(t) \equiv \sqrt{w_i^2(t) - r_i^2(t)}$ for $i = \iota, j$. In the above equations, the time argument is omitted for brevity, and the indices ι and j are obtained from (9). The Heaviside function $H(\cdot)$ guarantees that if $\hat{\mathbf{I}}^* \succ \hat{\mathbf{h}}^*$, the opening angle of the coverage cone is equal to zero.

By placing a second inertial frame of reference $x'y'$ along the remaining sides of \mathcal{A} , Proposition 4.1 can be applied to stabbers that intercept the x, y' , and x' axis at b_x, b_y , and $b_{x'}$, respectively, as shown in Fig. 3. The opening angles of the corresponding k -coverage cones are denoted by ρ, ξ , and ζ , respectively, and are illustrated in Fig. 3 for $n = k = 2$. The set of tracks traversing \mathcal{A} and intersecting at least k circles in $S(t)$ is approximated by the union of the k -coverage cones over a finite set of intercept values that are indexed by the superscript l . The intercept values are obtained by discretizing the perimeter of the ROI, $\partial\mathcal{A}$, using a constant interval, δb . Then, as shown in Appendix III, the following track coverage function

is a Lebesgue measure on the set of tracks that intersect at least k circles in $S(t)$ at time t

$$\begin{aligned}
 \mathcal{T}_{\mathcal{A}}^k[\mathbf{x}(t), \mathbf{r}(t)] &= \frac{1}{2} \sum_{\ell=1}^{L_2/\delta b} \sum_{j=1}^q (-1)^{j+1} \sum_{1 \leq i_1 < \dots < i_j \leq q} \{ \psi [S_p^{i_1, j}(t), b_y^\ell] \\
 &+ \xi [S_p^{i_1, j}(t), b_{y'}^\ell] \} \\
 &+ \frac{1}{2} \sum_{\ell=0}^{(L_1/\delta b)-1} \sum_{j=1}^q (-1)^{j+1} \sum_{1 \leq i_1 < \dots < i_j \leq q} \{ \rho [S_p^{i_1, j}(t), b_x^\ell] \\
 &+ \zeta [S_p^{i_1, j}(t), b_{x'}^\ell] \} \quad (12)
 \end{aligned}$$

where q is the binomial coefficient n choose k (Appendix III), the summation $\sum_{1 \leq i_1 < \dots < i_j \leq q}$ is the sum over all $[q!/(q-j)!j!]$ distinct integer j -tuples (i_1, \dots, i_j) satisfying $1 \leq i_1 < i_j \leq q$, and $S_p^{i_1}(t)$ denotes the i_1 th p -subset of $S(t)$, at time t [34]. The proof is based on the principle of inclusion-exclusion [13], and is shown in Appendix III. The authors recently showed that the computational complexity of the Lebesgue measure (12) for static sensors is $O(n_\delta q(k + \log q))$, where $n_\delta = L_2/\delta b$ [35]. Since integration is a linear operation, the computational complexity of integrating (12) numerically over time is $O(N n_\delta q(k + \log q))$, where N is the number of collocation points (Section IV-A). Thus, the computation time increases significantly as the number of sensors n , and the parameter q increase.

The area coverage of a sensor network is defined as the union of the areas representing the sensors' FOVs divided by the area of the ROI, i.e.,

$$A_C = \frac{1}{L_1 L_2} \bigcup_{i=1}^n A_i \quad (13)$$

where A_i is the area covered by the i th sensor [17]. If the sensors' positions, \mathbf{x} , and effective ranges, \mathbf{r} , are known and fixed, (13) can be computed from their FOV using planar geometry. When the sensors move subject to drift, however, \mathbf{x} and \mathbf{r} are functions of time that obey the dynamic and equality constraints, (3) and (45). Thus, a measure of area coverage is derived here as a function of \mathbf{x} and \mathbf{r} , such that it can be integrated and optimized over time. Let A_0 denote the instantaneous area coverage of the sensors in \mathcal{A} at time t provided none of the FOV intersect each other or $\partial\mathcal{A}$, i.e.,

$$\begin{aligned}
 A_0(t) &= \sum_{i=1}^n H[x_i(t) - r_i(t)] H[y_i(t) - r_i(t)] \\
 &\times H[L_1 - x_i(t) - r_i(t)] \\
 &\times H[L_2 - y_i(t) - r_i(t)] \pi r_i^2(t) \quad (14)
 \end{aligned}$$

where the Heaviside function, $H(\cdot)$, ensures that only sensors that lie entirely in \mathcal{A} contribute to A_0 . Now, let A_s denote the instantaneous area-coverage reduction due to intersecting circular segments in \mathcal{A} , and A_p denote the instantaneous area-coverage reduction due to circular segments that intersect $\partial\mathcal{A}$, as shown

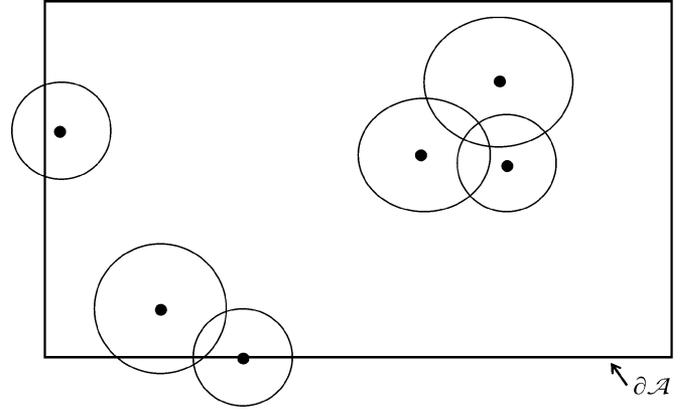


Fig. 4. Sensors with reduced area coverage due to intersections between FOVs or the perimeter, $\partial\mathcal{A}$.

by the example in Fig. 4. Then, the effective area coverage of the sensor network is

$$A_C(t) = A_0(t) - A_s(t) - A_p(t) \quad (15)$$

where A_0, A_s and A_p are all functions of \mathbf{x} and \mathbf{r} , as shown in the remainder of this section and, thus, $A_C(t) = A_C[\mathbf{x}(t), \mathbf{r}(t)]$.

A circular segment is the portion of a circle C_i that is delimited by an arc, s_i , and a chord, c_i , with a central angle $\theta_i < \pi$ that obeys $\theta_i = s_i/r_i$, as shown in Fig. 5. Let $\theta_{ij}(t)$ denote the central angle of the circular segment formed by the intersection of $C_i(t)$ and $C_j(t)$, at time t , as shown in Fig. 6. From the properties of sectors [36], the area-coverage reduction experienced by the i th sensor due to an intersection with the j th sensor (Fig. 6) is given by

$$A_{ij}^c(t) = \frac{1}{2} r_i^2(t) [\theta_{ij}(t) - \sin \theta_{ij}(t)] \quad (16)$$

where

$$\theta_{ij}(t) = 2 \cos^{-1} \left\{ \sin \left[\frac{1}{2} \cos^{-1} \left(\frac{r_i^2(t) + r_j^2(t) - \|\mathbf{x}_i(t) - \mathbf{x}_j(t)\|^2}{2r_i(t)r_j(t)} \right) \right] \right\} \quad (17)$$

and all quantities are defined as in Figs. 5–7. Similarly, the area-coverage reduction experienced by the j th sensor, denoted by A_{ji}^c , is obtained by exchanging the subscripts i and j in the above equations. As shown in Fig. 6, each sensor experiences a different reduction in coverage and, therefore, the total reduction is obtained by considering all n sensors pairwise

$$\begin{aligned}
 A_s(t) &= \sum_{i=1}^{n-1} \sum_{j=2}^n \left[1 + \frac{r_i^2(t)}{r_j^2(t)} \right] \\
 &\times H[r_i(t) + r_j(t) - \|\mathbf{x}_i(t) - \mathbf{x}_j(t)\|] A_{ij}^c(t). \quad (18)
 \end{aligned}$$

See Appendix IV for a detailed proof.

The instantaneous area-coverage reduction due to sensors that intersect $\partial\mathcal{A}$ is obtained by computing the area of each cir-

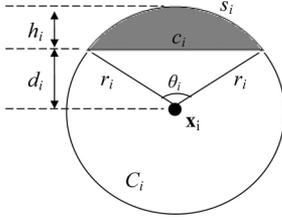


Fig. 5. Geometry and notation of a circular segment for the i th sensor.

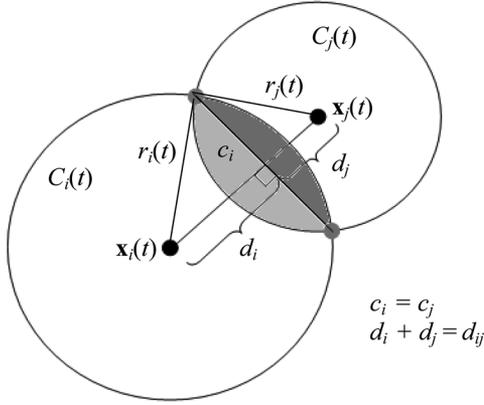


Fig. 6. Circular segments representing the area-coverage reductions $A_{ij}^c(t)$ (dark gray) and $A_{ji}^c(t)$ (light gray) experienced at time t by sensors i and j , respectively.

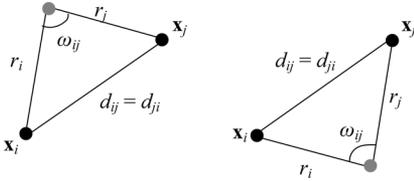


Fig. 7. Geometry and notation of circular segments in Fig. 6.

circular segment that lies outside of \mathcal{A} from the distance between \mathbf{x}_i and the chord of the circular segment. Then, as shown in Appendix IV, the total area-coverage reduction is

$$A_p(t) = \sum_{i=1}^n \sum_k r_i^2(t) H[r_i(t) - d_{ik}(t)] \times \left\{ \cos^{-1} \left[\frac{d_{ik}(t)}{r_i(t)} \right] - \frac{1}{2} \sin \left[2 \cos^{-1} \left(\frac{d_{ik}(t)}{r_i(t)} \right) \right] \right\} \quad (19)$$

where $k = x, y, x', y'$, and $d_{ix} = y_i, d_{iy} = x_i, d_{ix'} = (L_2 - y_i), d_{iy'} = (L_1 - x_i)$ denote the distances between the i th sensor and the four axis in Fig. 3, which are indexed by k .

By substituting (14), (18), and (19) into (15), the derivation of the area coverage function is complete, and the integral objective function (4), representing the sensor network's quality-of-service over ΔT , is obtained with respect to \mathbf{x} and \mathbf{r} . Then, using the methodology reviewed in the next subsection, (4) can be optimized with respect to the initial deployment \mathbf{x}_0 , subject to the equation of motion and the effective-range model.

A. Optimal Deployment Solution via Direct-Shooting Method

In this section, a methodology based on the direct-shooting method is presented for computing the optimal sensor network deployment, \mathbf{x}_0^* , that maximizes the quality-of-service (4), subject to equality and dynamic constraints. When the equation of motion (3) is linear, a closed-form solution to \mathbf{x}_0^* can be obtained from the systems' transition matrix [37]. Typically, however, the current velocity field and, consequently, the equation of motion (3) is NLTV, and \mathbf{x}_0^* must be determined numerically. Direct shooting is a numerical method that has been developed and proven effective for solving optimal control problems in which an integral objective function is to be optimized subject to NLTV dynamic constraints [38]. The sensor-network deployment problem considered in this paper, however, is not an optimal control problem because the sensors have no on-board actuators, and their state trajectories depend solely on their initial positions. Therefore, in this section, direct shooting is modified to determine \mathbf{x}_0^* numerically, by evaluating the equation of motion, equality constraints, and objective function at discrete points in time, known as *collocation points*, and by transcribing the dynamic optimization problem into a NLP in \mathbf{x}_0 . Between collocation points, the equation of motion (3) is integrated by Euler integration ([39], p. 77). Although higher order numerical-integration routines, such as Runge-Kutta integration [39], could potentially be applied, Euler integration was found to be adequate for integrating (3), and more computationally efficient (Section VII).

For N equally spaced collocation points, let $t_k = t_0 + k\Delta t$ denote a discrete time index, where $k = 0, \dots, (N-1)$, and $\Delta t = (t_f - t_0)/N$ is the discretization interval. Then, the optimization of an integral objective function subject to continuous-time dynamics can be transformed into a finite-dimensional NLP by discretizing the objective function (4) with respect to time

$$J(\mathbf{x}_0) = \phi(\mathbf{x}_N) + \Delta t \sum_{k=0}^{N-1} [W_T T_A^k(\mathbf{x}_k, \mathbf{r}_k) + W_A A_C(\mathbf{x}_k, \mathbf{r}_k)] \quad (20)$$

and by performing an implicit integration of the system dynamics (3), where $\mathbf{x}_k \equiv \mathbf{x}(t = t_k)$, and $\mathbf{r}_k \equiv \mathbf{r}(t = t_k)$. Using Euler integration, the residuals

$$\boldsymbol{\rho}_k(\mathbf{x}_0) = \mathbf{x}_{k+1} - \mathbf{x}_k - \mathbf{f}(\mathbf{x}_k, t_k; \mathbf{x}_0) \Delta t, \quad k = 0, \dots, N-1 \quad (21)$$

are computed at all collocation points, using only one function evaluation per residual, and are driven to zero as part of the optimization process [40]. When the effective sensor-range model is evaluated at every collocation point, t_k , it is transformed into N n -dimensional equality constraints

$$\mathbf{c}_k(\mathbf{x}_0) \equiv \mathbf{r}_k - \begin{bmatrix} g[\mathbf{x}_1(t_k)] \\ \vdots \\ g[\mathbf{x}_n(t_k)] \end{bmatrix} = \mathbf{0}_{n \times 1} \quad (22)$$

where $k = 0, \dots, N-1$, and the sensors' positions and effective ranges over time depend on \mathbf{x}_0 . It follows that the solution \mathbf{x}_0^* of the NLP

$$\text{maximize } J(\mathbf{x}_0) \quad (23)$$

$$\text{subject to } \boldsymbol{\rho}_k(\mathbf{x}_0) = \mathbf{0}_{n \times 1}, \quad k = 0, \dots, N-1 \quad (24)$$

$$\mathbf{c}_k(\mathbf{x}_0) = \mathbf{0}_{n \times 1}, \quad k = 0, \dots, N-1 \quad (25)$$

constitutes a near-optimal solution for the sensor-network dynamic optimization problem in continuous time. The NLP solution can be obtained by SQP [41], [42], and it can be made arbitrarily close to the optimal solution by using a higher order integration rule and by increasing the size of the decision vector, letting $\Delta t \rightarrow 0$ and $N \rightarrow \infty$, at the expense of the computation time [38]. Local minima may be avoided by using multiple random initializations [41].

The effectiveness of the OD method presented in this section is demonstrated through numerical simulations in Section VII. Since this method utilizes approximate models of the sensors' measurements and dynamics, presented in Section VI, the robustness of its performance to various sources of uncertainty is analyzed in Section VII-B, using the approach reviewed in the next section.

V. STOCHASTIC ROBUSTNESS ANALYSIS

The deployment method presented in the previous section exploits models of the system's components that influence the sensor network's performance over time, in order to optimize its ability to perform cooperative track detection. Since models are always subject to errors, robustness analysis is used here to determine the possibility of inadequate performance in the face of uncertainty [43], [44]. Unlike deterministic approaches, such as singular-value analysis [45] and parameter-space methods [46], the SRA approach presented in [43], [44] can be applied to sensor network deployment because it relies on a statistical description of parameter uncertainty, and does not require the system's dynamics and objectives to be linear and quadratic, respectively. In a nonlinear dynamical system, such as (3), the effects of uncertainty on the state \mathbf{x} are propagated stochastically over time, and \mathbf{x} can be considered as a random variable with a time-varying probability density function (PDF) whose higher moments (or variability) typically increase over time [39]. MC simulations are a flexible and effective approach for evaluating the effects of uncertainty propagation on the system performance, such as response envelopes and confidence levels, for one or more uncertain parameters with known PDFs.

The PDF models of parameter uncertainties are specified by the user, based on system's knowledge, experiments, and heuristics. In ocean sensor networks, for example, the main sources of uncertainty are the execution of the initial placement, \mathbf{x}_0^* , the current velocity field, and the environmental conditions that influence the effective sensor range. Once \mathbf{x}_0^* is computed by the methodology in Section IV, the sensors are deployed by a surface or air vehicle and, thus, are subject to random errors $\mathbf{n}_\epsilon \in \mathbb{R}^{2n}$ caused by navigation devices, and to random bias errors $\mathbf{d}_\epsilon \in \mathbb{R}^{2n}$ caused, for example, by winds or human errors. Current velocity forecasts and oceanographic field estimates that are used to derive the equation of motion (3), as

shown in Section VI-A, are subject both to modeling errors and to measurement errors in assimilated data. In turn, these errors cause random Gaussian errors in the current velocity field and effective range that are denoted by \mathbf{v}_ϵ and \mathbf{r}_ϵ , respectively. Thus, the actual initial positions, current velocities, and effective sensor ranges are considered as random variables defined as

$$\hat{\mathbf{x}}_0 = \mathbf{x}_0^* + \mathbf{d}_\epsilon + \mathbf{n}_\epsilon \quad (26)$$

$$\hat{\mathbf{v}}(t) = [\mathbf{v}_1^b(t) \cdots \mathbf{v}_n^b(t)]^T + \mathbf{v}_\epsilon \quad (27)$$

$$\hat{\mathbf{r}}(t) = \mathbf{r}(t) + \mathbf{r}_\epsilon \quad (28)$$

respectively, in terms of the nominal variables, \mathbf{x}_0^* , $\mathbf{r}(t)$, and $\mathbf{v}_i^b(t)$ (representing the surface currents experienced by the i th sensor), where (27) and (28) depend on the sensors' positions over time, $\hat{\mathbf{x}}(t)$, which are propagated stochastically in the presence of these uncertainties using a MC simulation. Every component of the error vectors is independently and identically sampled from a multivariate normal PDF with a mean and covariance matrix specified by the user (as shown in Section VII-B).

In the MC simulation, the errors in (26) and (27) are propagated through time by integrating the sensor network's dynamics (3) with sampled initial conditions $\hat{\mathbf{x}}_0$ and current velocities $\hat{\mathbf{v}}$. In each MC evaluation, the stochastic state history $\hat{\mathbf{x}}(t)$ and objective function \hat{J} are computed. Thus, given a sufficient number of evaluations M , the probability of adequate coverage

$$\mathbb{P} = \Pr\{\hat{J}/J \geq \varrho\}, \quad 0 \leq \varrho \leq 1 \quad (29)$$

is estimated as the ratio between the number of evaluations with adequate coverage over the total number of evaluations (M) [43], [47]. A confidence interval bounds the expected error of this estimate, such that the true value (29) lies in an interval $[L, U]$ with probability

$$\Pr\{L \leq \mathbb{P} \leq U\} = 1 - \alpha_c \quad (30)$$

where α_c is the confidence coefficient [43], [48]. For example, \mathbb{P} lies in $[L, U]$ with $100(1 - \alpha_c)\%$ confidence, and a 95% confidence interval implies that in 95% of the MC evaluations, \mathbb{P} lies in $[L, U]$, i.e.,

$$\Pr\left\{\hat{\mu}_J - 1.95 \frac{\hat{\sigma}_J}{\sqrt{M}} \leq \mathbb{P} \leq \hat{\mu}_J + 1.95 \frac{\hat{\sigma}_J}{\sqrt{M}}\right\} = 0.95 \quad (31)$$

where $\hat{\mu}_J$ and $\hat{\sigma}_J$ are the estimated mean and standard deviation of the objective function, respectively. Therefore, $\hat{\mathbb{P}}$ and $[L, U]$ can be considered as performance robustness criteria. The narrower is the confidence interval, the more precise is the estimate $\hat{\mathbb{P}}$, and the higher is $\hat{\mathbb{P}}$, the more robust is the system performance.

In the next section, the OD methodology presented in Section IV is applied to an ocean sensor network comprised of sonobuoys, and deployed to cooperatively detect passive targets in a large ROI. The effectiveness and robustness of the methodology are illustrated through numerical simulations in Section VII.

VI. APPLICATION TO OCEAN SENSOR NETWORKS

Drifting surface sonobuoys provide a means for collection of remote ocean data over large regions of interest and, possibly,

over large time scales, at moderate cost [20]. Although many sonobuoys designs have been developed over the years, drogued drifting buoys have been shown to follow the path of a true Lagrangian drifter and, therefore, are typically implemented for ocean sensing [49]. Because their cost is relatively low, sonobuoys are often considered expendable, and are launched from surface vessels or air platforms at desired locations [10]. The most frequently used sonobuoys are omnidirectional and operate at a fixed detection threshold that is decided *a priori* by an operator, based on the environmental conditions and the sensing objectives. The position of omnidirectional sonobuoys in the ROI is crucial both to track detection algorithms [18] and to the overall performance of the sensor network, and is often assumed constant for simplicity [10]. However, current-induced drifting has been shown to rapidly displace networks of sonobuoys, especially near coastlines and regions of strong ocean currents. Recently, a model was developed and experimentally validated in [9] to simulate the drift of free-floating sonobuoys within a deployed network in the ocean. This sonobuoys field drift model (SFDM) uses the current velocity field as an input, and solves the sonobuoy equilibrium equations updating its position recursively over time.

In Section VI-A, a simplified version of SFDM is combined with a model of the current velocity field forecast in order to derive the sensor network's equation of motion (3). A model of effective sensor range is obtained in Section VI-B from a range-dependent acoustic model and oceanographic field estimates, specifying the equality constraints (22). Then, the methodology presented in Section IV is used to determine the optimal sensor deployment, such that the ability of the network to perform cooperative detections in \mathcal{A} is maximized over the time interval ΔT . The results, presented in Section VII, show that this methodology improves track and area coverage by up to a factor of five compared to existing techniques.

A. Equation of Motion of Lagrangian Drifting Sonobuoys

The sonobuoy equation of motion is based on the theory of hydrodynamic array models, which was extended to horizontal line sensor arrays in [50]. The buoy and the drogue are approximated by two spheres that are connected by a flexible cable (or tether) of negligible mass, through their centers of mass, as schematized in Fig. 8(a). The total drag on a sphere is given by the steady-state solution to Stokes' Problem along the local current velocity vector

$$f_D = \frac{1}{2}\rho C_d A V^2 \quad (32)$$

where ρ is the water density, C_d is the sphere's drag coefficient, A is its cross-sectional area, and V is the magnitude of the relative velocity vector. Experimental results have shown that the vertical current velocity profile can be assumed to take the form in Fig. 8(a), where each sphere experiences a constant but different current denoted by a superscript $(\cdot)^b$ for the buoy, and $(\cdot)^d$ for the drogue [9]. The local current velocity vectors experienced by the buoy and the drogue of the i th sensor can be resolved into the x and y components in the plane [Fig. 8(b)],

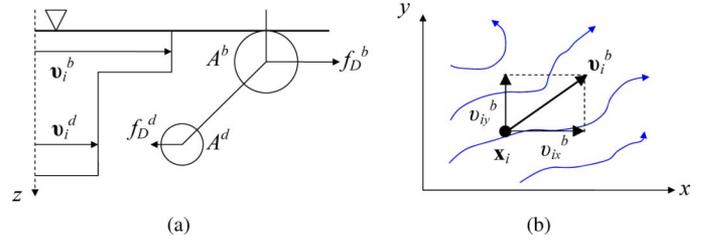


Fig. 8. Modeling of a drogued drift buoy (a) positioned at \mathbf{x}_i (b), and subject to local ocean current velocities, v_i^b and v_i^d .

such that $\mathbf{v}_i = [v_{i_x} \ v_{i_y}]^T$. Therefore, the relative velocity vector is defined as

$$\mathbf{v}_i[\mathbf{x}_i(t), t] \equiv \dot{\mathbf{x}}_i(t) - \mathbf{v}_i[\mathbf{x}_i(t), t] \quad (33)$$

where $\dot{\mathbf{x}}_i(t) = [\dot{x}_i(t) \ \dot{y}_i(t)]^T$ is the sonobuoy's velocity in inertial frame, and the current velocity vector \mathbf{v}_i is a function of time and position.

Assuming the sonobuoy is at equilibrium along its entire trajectory, the external force on the drogue must be equal and opposite to that on the buoy at all times, i.e., $f_D^b(t) = f_D^d(t)$. Since (32)–(33) apply to both the buoy and the drogue, the force balance equation can be written as

$$C_d^b A^b \nu_{i_x}^b[\mathbf{x}_i(t), t]^2 = C_d^d A^d \nu_{i_x}^d[\mathbf{x}_i(t), t]^2 \quad (34)$$

$$C_d^b A^b \nu_{i_y}^b[\mathbf{x}_i(t), t]^2 = C_d^d A^d \nu_{i_y}^d[\mathbf{x}_i(t), t]^2 \quad (35)$$

by resolving the relative velocities in their x and y components. Then, the equation of motion of the i th sonobuoy is obtained by a few simple manipulations. Letting $\beta \equiv \sqrt{(C_d^d A^d)/(C_d^b A^b)}$, (34) and (35) can be written as $\nu_{i_x}^b[\mathbf{x}_i(t), t] = \beta \nu_{i_x}^d[\mathbf{x}_i(t), t]$ and $\nu_{i_y}^b[\mathbf{x}_i(t), t] = \beta \nu_{i_y}^d[\mathbf{x}_i(t), t]$, respectively, and from (33) the velocity of the sonobuoy in inertial frame can be written as

$$\dot{\mathbf{x}}_i(t) = \left[\begin{array}{c} \frac{1}{1+\beta} v_{i_x}^b[\mathbf{x}_i(t), t] + \frac{\beta}{1+\beta} v_{i_x}^d[\mathbf{x}_i(t), t] \\ \frac{1}{1+\beta} v_{i_y}^b[\mathbf{x}_i(t), t] + \frac{\beta}{1+\beta} v_{i_y}^d[\mathbf{x}_i(t), t] \end{array} \right], \mathbf{x}_i(t_0) = \mathbf{x}_{i_0} \quad (36)$$

where β is a known constant parameter, and $v_{i_x}^{b,d}$ and $v_{i_y}^{b,d}$ are the current velocity components. In practice, the local current velocity experienced by the drogue is a fraction of the surface current velocity experienced by the buoy and, thus, can be modeled as $\nu_{i_x}^d[\mathbf{x}_i(t), t] = \alpha v_{i_x}^b[\mathbf{x}_i(t), t]$, where the parameter $0 \leq \alpha \leq 1$ is determined from experiments, and the surface current \mathbf{v}_i^b is obtained from a model of the ocean velocity forecast in \mathcal{A} , as shown below. Then, (36) simplifies to

$$\begin{aligned} \dot{\mathbf{x}}_i(t) &= \left[\begin{array}{c} \left(\frac{1+\beta\alpha}{1+\beta} \right) v_{i_x}^b[\mathbf{x}_i(t), t] \\ \left(\frac{1+\beta\alpha}{1+\beta} \right) v_{i_y}^b[\mathbf{x}_i(t), t] \end{array} \right] \\ &= \gamma \left[\begin{array}{c} v_{i_x}^b[\mathbf{x}_i(t), t] \\ v_{i_y}^b[\mathbf{x}_i(t), t] \end{array} \right] \equiv \mathbf{f}_i[\mathbf{x}_i(t), t; \mathbf{x}_{i_0}] \end{aligned} \quad (37)$$

where $\gamma \equiv (1 + \beta\alpha)/(1 + \beta) < 1, i = 1, \dots, n$, and the argument \mathbf{x}_{i_0} represents the dependency on the initial conditions. For simplicity, hereon all sonobuoys are assumed to have the

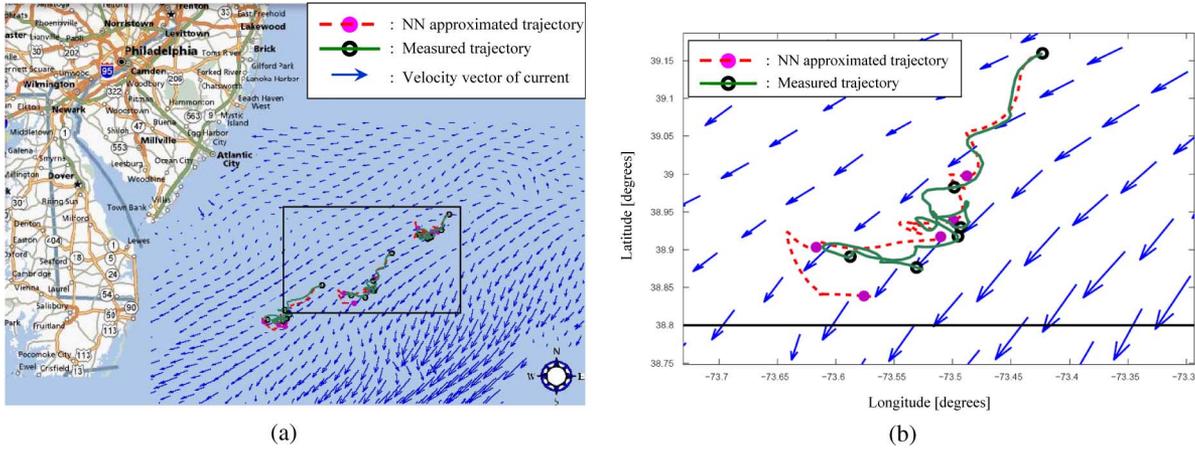


Fig. 9. Point-mass trajectories obtained from the neural network model and from CODAR measurements in an ROI with longitude coordinates $(74.1^\circ\text{W}, 72.7^\circ\text{W})$ and latitude coordinates $(38.6^\circ\text{N}, 39.5^\circ\text{N})$ plotted in a black solid line.

same size and drag coefficients. In order to obtain the sensor network's equation of motion (3), (37) is combined with a neural network model of the velocity forecast in \mathcal{A} .

A forecast of the ocean current velocity field over \mathcal{A} can be obtained by oceanographic models with assimilated data [51]. A forecast is a future prediction that is initialized with present estimates, and is obtained from physical models and data via state-estimation techniques, as reviewed in [51] and [52]. The data consists of past measurements of the ocean's bathymetry, circulation systems, and oceanographic field information, as well as of real-time measurements of the ocean's currents that may be obtained via satellite [53] or via CODAR [22]. Typically, a forecast of the current field consists of a multidimensional array containing estimated values of the components of the current velocity vector, \mathbf{v}_i , at sample points in space and time. The equation of motion (3), however, requires a continuous representation in space and time, i.e., for $\forall \mathbf{x}_i \in \mathcal{A}$ and $\forall t \in \Delta T$. Although tabular data representations can potentially be incorporated by combining them with an interpolation routine, this approach typically leads to problems that are computationally intractable [54]. One approach that has been presented to model the ocean current data and circumvent these problems, is to first process it by open-boundary model analysis [55], and then to represent it by B-splines basis functions [56]. In this paper, a compact functional representation of the current velocity forecast is obtained by means of a neural network function approximator. The advantages of neural networks over B-splines are that they require fewer parameters, and can be obtained in one step, without preprocessing the velocity forecast [57]–[59]. Also, they are found to be very effective at smoothing noisy measurements, and at generalizing in domains with sparse data [60], [61].

Assume there exists a continuously differentiable function $\mathbf{h} : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ that approximates the spatial and temporal characteristics of the current velocity field over \mathcal{A} , during ΔT [56]. Then, if $\mathbf{p}(t) \equiv [\mathbf{x}_i^T(t) t]^T \in \mathcal{P}$, where $\mathcal{P} = \mathcal{A} \times \Delta T$, the buoy's current velocity vector is $\mathbf{v}_i^b(t) = \mathbf{h}[\mathbf{p}(t)]$, at any sensor location $\mathbf{x}_i(t) \in \mathcal{A}$. The five-dimensional array containing the surface-current forecast is used to form a training set $F = \{\mathbf{x}_i^\kappa, \mathbf{v}_i^{\kappa}, t^\kappa\}_{\kappa=1, \dots, p}$, containing the value of \mathbf{v}_i^b at the location \mathbf{x}_i^κ , and at the time instant t^κ , for p sample points in \mathcal{P} .

Then, F can be used to approximate the function $\mathbf{h}[\cdot]$ by means of a feedforward neural network

$$\mathbf{v}_i^b[\mathbf{x}_i(t), t] = \mathbf{W}_2 \Phi[\mathbf{W}_1 \mathbf{p}(t) + \mathbf{b}_1] + \mathbf{b}_2 \quad (38)$$

with two linear output neurons, and one hidden layer of s sigmoidal functions represented by the operator

$$\Phi(\mathbf{n}) \equiv [\sigma(n_1) \cdots \sigma(n_s)]^T \quad (39)$$

where $\mathbf{n} = [n_1 \cdots n_s]^T$ is the input to the hidden layer, and $\sigma(n_i) \equiv 1/(1 + e^{-n_i})$. The adjustable parameters or weights $\mathbf{W}_1 \in \mathbb{R}^{s \times 3}$ and $\mathbf{W}_2 \in \mathbb{R}^{2 \times s}$, and the input and output biases, $\mathbf{b}_1 \in \mathbb{R}^s$ and $\mathbf{b}_2 \in \mathbb{R}^2$, are determined from F by a Bayesian regularization backpropagation algorithm ("trainbr" [62]) that minimizes a linear combination of neural network squared errors and weights, by means of a Levenberg–Marquardt optimization routine [61], [63]. Regularization refers to the Bayesian comparison of alternative models of the data, in this case in the form (38), that are obtained by different choices of parameters. Bayes' rule is used to infer the normalizing constants in the combination of multiple models, and the parameters are determined by maximizing their posterior probability given the data F . As a result, a model (38) with excellent generalization properties can be obtained even in the presence of noisy data [61]–[63].

This approach is verified by training the neural network model (38), with $s = 100$, using real CODAR measurements of the ocean currents near the New Jersey coast obtained by COOL [22] at Rutgers University over a period of five days. A snapshot of the current velocity field is shown in Fig. 9 by plotting sample vectors whose length is proportional to the velocity's magnitude, at a representative moment in time. The database of CODAR measurements was sampled to produce a training set F with $p = 12\,525$ samples, and the remaining data was used to validate the neural network model. Fig. 10 shows a comparison between the neural network output, \mathbf{v}_i^b , and the measured velocity components drawn both from training and validation data, at 600 sample locations, and $t = 122$ hr. The neural network approximation of the velocity components $v_{i_x}^b$ and $v_{i_y}^b$ [defined in Fig. 8(b)] as a function of position

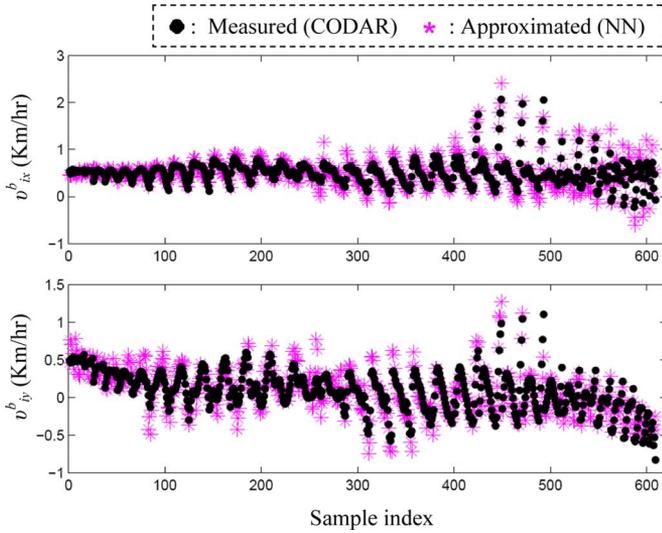


Fig. 10. Comparison of neural network approximation and measured velocity components at 600 sample locations in the ROI, at time $t = 122$ hr.

at different moments in time is plotted in Fig. 11, with the CODAR measurements superimposed on the surface in black dots to illustrate the effectiveness of the approximation. The approach is also validated by comparing the trajectories of point masses placed in the velocity field modeled by the neural network (38), to those obtained using the CODAR database combined with a cubic-spline interpolation routine (“trainbr” [64]). Each point mass is randomly placed in the ROI and allowed to drift according to a second-order Newtonian dynamic model [1] for a period of five days. As shown by the three examples in Fig. 9(a), and by the higher resolution example in Fig. 9(b), the trajectories obtained with the neural network approximator are very close to those obtained from the real CODAR measurements.

Finally, the sensor network’s equation of motion (3) is obtained by combining the sonobuoy model (37) with the current forecast model (38). Then, the effective range of each sensor in the network is modeled as a function of the sensor position \mathbf{x}_i that is obtained from oceanographic field information and an acoustic propagation model, as shown in the next section.

B. Model of Effective Sensor Range

The field-of-view of many low-cost wireless sensors, including passive single-hydrophone omnidirectional sensors that are commonly installed on sonobuoys, can be approximated by a circle centered at the sensor position [25]. The circle’s radius is referred to as effective range, and represents the maximum range of a target that would cause the received signal to exceed the detection threshold. When the i th sensor declares a detection, the target can be assumed to be closest to \mathbf{x}_i than to any other sensor, and the i th sensor receives the highest signal in the network [10]. Range-dependent parabolic equation (PE) models are the most accurate methods for capturing the influence of environmental conditions on acoustic propagation loss and the received signal in the ocean. But, because they rely on numerical integration and known boundary conditions, PE models do not provide estimates of effective range as a function

of spatial coordinates. Instead, underwater acoustic PE models have recently been used in the literature to train neural or Bayesian networks to perform efficient geoacoustic inversion, and obtain the effective-range model, $r_i = g(\mathbf{x}_i)$ [65], [66]. In [66], the Bayesian network (BN) underwater acoustic model was shown to outperform a neural network model trained with the same data in both accuracy and flexibility. Thus, in this subsection, the BN approach presented in [66] and [67] is applied to obtain the effective-range model from the PE solution, the oceanographic fields estimates in \mathcal{A} , and the sonar equation.

A range-dependent PE model based on the split-step Padé approximation solution can be used to obtain a numerical solution to the far-field equation

$$\frac{\partial^2 p}{\partial \lambda^2} + \rho \frac{\partial}{\partial z} \left(\frac{1}{\rho} \frac{\partial p}{\partial z} \right) + k^2 p = 0 \quad (40)$$

which holds in each range-independent region [21], where ρ is the density, and k is the complex wave number, which depends on the attenuation, the circular frequency, and the sound speed [21]. In this model, range dependence is handled accurately by applying an energy-flux conservation correction at the vertical interfaces between regions, as the environmental conditions vary with range. Using the reference value of $1 \mu\text{Pa}$, the propagation loss is defined as

$$p_\ell = -20 \log |p| \quad (41)$$

and is computed from the PE solution in units of dB. From (40) and (41), p_ℓ is a function of the target’s range λ and depth z , and, through the parameters ρ and k , it is influenced by the target frequency f , and by the ocean environmental conditions, such as, the bottom sound speed c_b , bottom density ρ_b , and sea-floor profile s_f [21].

A BN acoustic model obtained from (40) can be used for geoacoustic inversion, for example, by inferring environmental parameters or the target’s range from the propagation loss [65]–[67]. A BN model is a directed acyclic graph that can be learned from data in order to obtain a convenient factorization of the joint multivariate probability mass function (PMF) of the sensor variables [68]. Based on the physical model (40), the set of acoustic sensor variables, referred to as BN universe, is defined as $U = \{p_\ell, \lambda, z, f, c_b, \rho_b, s_f\}$. The sensor variables are discretized using equal-width discretization [69], which has been shown to outperform other approaches in [66]. Then, the BN acoustic data is obtained by solving (40) repeatedly by means of the range-dependent acoustic model [21], under all possible combinations of discrete sensor variables’ values. This data is divided into a training set and a validation set. The training set is used to learn the structure and parameters of the BN, by means of the K2 greedy-search algorithm [70] and the expectation-maximization (EM) algorithm [71], [72], respectively. The validation set, which is not used for learning, is then used to test the effectiveness of the BN model by estimating one or more sensor variables from known values of other variables in U .

The BN acoustic model implemented in this paper was learned from a training set with 30 000 PE solutions, and was validated with 10 000 PE solutions that were not used for

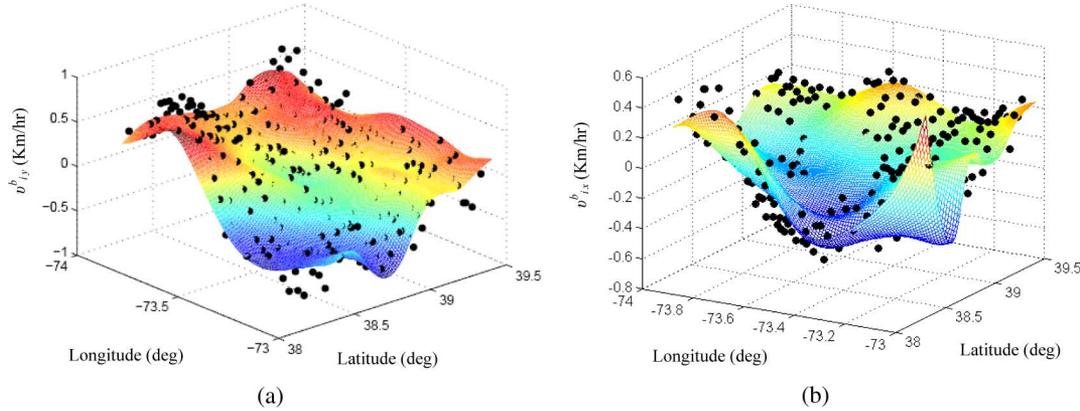


Fig. 11. Neural network approximation of the velocity components as a function of position at (a) $t = 66$ hr and (b) $t = 101$ hr, with available CODAR measurements superimposed in black dots.

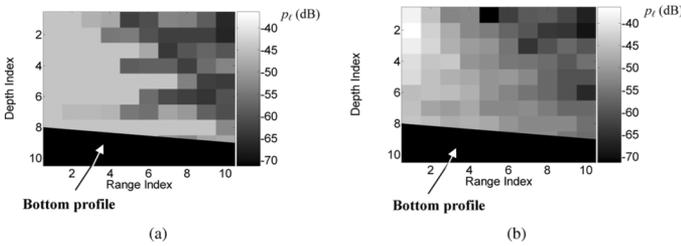


Fig. 12. Acoustic propagation loss p_ℓ estimated by the BN model (b) and by the range-dependent acoustic model (a).

learning (see [66] for more details). The resulting BN acoustic model is found to compute p_ℓ correctly in approximately 95% of solutions in the validation set, and to provide a reasonable approximation of p_ℓ as a function of range and environmental conditions, as shown by the example in Fig. 12. The corresponding BN factorization is

$$\begin{aligned} P(U) &= P(p_\ell, \lambda, z, f, c_b, \rho_b, s_f) \\ &= P(p_\ell | \lambda, z, f, c_b, s_f) P(c_b | \rho_b) \\ &\quad \times P(\rho_b) P(\lambda) P(z) P(f) P(s_f) \end{aligned} \quad (42)$$

where $P(x_1, x_2)$ denotes the joint PMF of x_1 and x_2 , and $P(x_1 | x_2)$ denotes the conditional PMF of x_1 given x_2 . All PMFs in the above factorization are learned from data using the EM algorithm. Subsequently, Bayes' rule is applied to (42) in order to infer λ , by computing its posterior PMF as a function of p_ℓ and of the environmental conditions

$$\begin{aligned} P(\lambda | p_\ell, z, f, c_b, \rho_b, s_f) \\ = \frac{P(p_\ell | \lambda, z, f, c_b, s_f) P(\lambda)}{\sum_\lambda P(p_\ell | \lambda, z, f, c_b, s_f) P(\lambda)} \end{aligned} \quad (43)$$

where \sum_λ denotes marginalization with respect to λ [73].

The effective sensor range of the i th sensor, r_i , is defined as the maximum value of λ that would cause the received signal to exceed ϑ_i . Thus, the corresponding value of propagation loss $p_{\ell_i}^{\max}$ can be obtained from the passive sonar equation [74]

$$p_{\ell_i}^{\max} = F + I_s - (n_l - I) - \vartheta_i \quad (44)$$

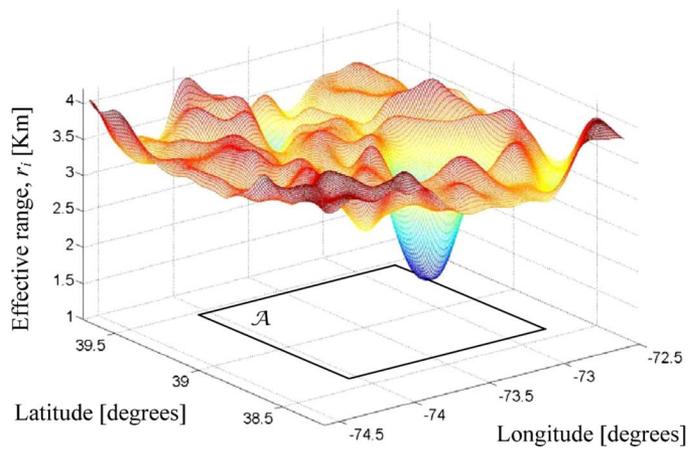


Fig. 13. Effective sensor range as a function of position (taken from [66]).

and substituted in (43), eliminating the range dependency on propagation loss, where the target source level F , depth z , and frequency f are assumed known and constant. The directivity index I , the target-source directivity I_s , and the detection threshold ϑ_i are known sensor parameters that, for simplicity, are assumed to be the same for all sensors. The noise level n_l in \mathcal{A} is assumed to be Gaussian, and can be estimated based on the expected ship traffic [74]. The environmental conditions c_b , ρ_b , and s_f depend on sensor position, and are obtained for $\forall \mathbf{x}_i \in \mathcal{A}$ using field estimates given by the Harvard Ocean Prediction System (HOPS) [75]. Then, the effective range

$$\begin{aligned} r_i &= \arg \max_\lambda \{ P[\lambda | p_{\ell_i}^{\max}, z, f, c_b(\mathbf{x}_i), \rho_b(\mathbf{x}_i), s_f(\mathbf{x}_i)] \} \\ &= g(\mathbf{x}_i) \end{aligned} \quad (45)$$

is estimated for $\forall \mathbf{x}_i \in \mathcal{A}$, using (43)–(44). When necessary, (45) can be filtered to smoothen the effects of sensor-variable discretization and of noisy environmental conditions. The effective range function obtained for the ROI in Fig. 9(a), and implemented in Section VII, is plotted in Fig. 13.

The effective ranges and positions of the n acoustic sensors determine the sensor network performance, because they determine its ability to cooperatively detect targets in \mathcal{A} . Since the sonobuoys move in \mathcal{A} according to the equation of motion

TABLE I
COMPARISON OF SENSOR NETWORK DEPLOYMENT METHODS

$(n, k) :$	Performance Measure	Optimal Deployment (OD)	Static Optimization (OD Improvement)	Grid (OD Improvement)	Random (OD Improvement)
(10,2)	Track Coverage	$1.671 \cdot 10^4$	8,823	4,866	4,951
	Area Coverage	$3.091 \cdot 10^4$	$1.119 \cdot 10^4$	$1.831 \cdot 10^4$	$1.843 \cdot 10^4$
	Objective Function	100.5	42.73 (+135%)	48.09 (+109%)	48.51 (+107%)
(15,3)	Track Coverage	$1.630 \cdot 10^4$	3,145	1,763	2,057
	Area Coverage	$5.368 \cdot 10^4$	8,223	$2.830 \cdot 10^4$	$3.094 \cdot 10^4$
	Objective Function	145.6	23.78 (+512%)	61.17 (+138%)	67.19 (+117%)
(20,3)	Track Coverage	$2.245 \cdot 10^4$	$1.326 \cdot 10^4$	2,800	6,281
	Area Coverage	$7.174 \cdot 10^4$	$2.965 \cdot 10^4$	$3.607 \cdot 10^4$	$3.471 \cdot 10^4$
	Objective Function	196.1	90.14 (+118%)	79.25 (+147%)	84.43 (+132%)
(25,4)	Track Coverage	$1.860 \cdot 10^4$	$1.358 \cdot 10^4$	932.3	2,648
	Area Coverage	$8.892 \cdot 10^4$	$3.211 \cdot 10^4$	$4.163 \cdot 10^4$	$4.494 \cdot 10^4$
	Objective Function	222.0	95.97 (+132%)	86.21 (+158%)	96.82 (+129%)

derived in Section VI-A, their coverage is a function of time and can be maximized over ΔT using the methodology presented in Section IV. As a result, the OD, \mathbf{x}_0^* , is found to significantly outperform other deployment techniques, improving quality-of-service by up to a factor of five, as shown in the next section.

VII. RESULTS

The OD approach presented in the previous sections is demonstrated on networks of sonobuoys that perform cooperative track detection in the ROI, \mathcal{A} , of dimensions $L_1 = 90$ Km and $L_2 = 82.51$ Km [shown in Fig. 9(a)], for a fixed ΔT . The time-varying ocean current field in and around \mathcal{A} is simulated using CODAR data provided by COOL at Rutgers University [22]. The surface current speed and direction is retrieved from the Doppler shift of the backscattered high-frequency signal [22]. The current velocity forecast in \mathcal{A} is represented by the feedforward neural network (38), and the system dynamics are simulated by integrating (3) explicitly using a fourth-order Runge–Kutta routine with variable stepsize (“ode45” [64]). This routine was also used to verify that the Euler integration used by the direct shooting method (Section IV-A) brings about a negligible error. In Section VII-A, the effectiveness of the approach presented in this paper is demonstrated by comparing the quality-of-service of networks that are optimally deployed to that of networks deployed by existing strategies, namely, static optimization [13], grid [23], and random [24], [25] deployments. In Section VII-B, the robustness of the OD method is analyzed using the approach described in Section V.

A. Comparison With Existing Deployment Strategies

Previous techniques devised to deploy sensor networks for cooperative track detection consist of static optimization [13], grid [23], and random deployments [24], [25], [76]. Static optimization via mathematical programming computes the sensors’ positions that optimize area and track coverage in \mathcal{A} , without accounting for the sensors’ dynamics [13], [17]. Grid deployment is popular in many wireless networks’ applications, and consists of placing sensors at equally spaced positions in \mathcal{A} [23], [77], [78]. Random deployment places sensors sequentially at

random, such that their FOV do not intersect, until a desired level of coverage is achieved in the ROI [24]. Random deployment has been shown effective at placing sensor networks to observe a target moving along a specified track in path-exposure problems [76], [79] or at replenishing sensor networks to improve coverage [25]. Recently, static optimization has been shown to improve the coverage of static sensor networks by a factor of two compared to random and grid deployments [13]. Nevertheless, as shown by the simulations in this subsection, if the sensors are subject to external forcing, such as, winds or ocean currents, their coverage deteriorates significantly over time, regardless of their optimal initial performance.

The approach presented in this paper, deploys sensors based on forecasted drift effects, by determining the optimal initial positions \mathbf{x}_0^* that maximize track and area coverage in \mathcal{A} , over a time interval ΔT . In this subsection, the results of different deployment strategies are simulated by integrating (3) explicitly, and by computing the network’s track and area coverage, (12) and (15), based on \mathbf{x}_0 , and on the positions and ranges, $\mathbf{x}(t)$ and $\mathbf{r}(t)$, at every moment in time $t \in \Delta T$. The results of OD, static optimization, grid, and random strategies applied to four networks of sonobuoys are summarized in Table I. The OD strategy is shown to increase both the area and the track coverage of the sensor networks by up to a factor of five compared to other deployment techniques.

For comparison, the initial positions and trajectories of a network with $n = 20$ sonobuoys, and $k = 3$, are simulated in Fig. 14 using all four deployment strategies. For this example, computing the OD strategy required approximately four hours on a Pentium 4 CPU 3.06 GHz computer. The resulting quality-of-service, plotted over time in Fig. 15, illustrates that the OD approach significantly improves performance compared to other deployments, and maintains a high level of coverage at all times. Conversely, sensors deployed by other techniques may drift outside of \mathcal{A} , or cluster into low-coverage configurations, as shown Fig. 14(b)–(d), such that their quality-of-service deteriorates significantly shortly after deployment. As can be expected, networks deployed by static optimization provide a quality-of-service that is initially optimal but that rapidly deteriorates over time, due to drift, eventually approaching

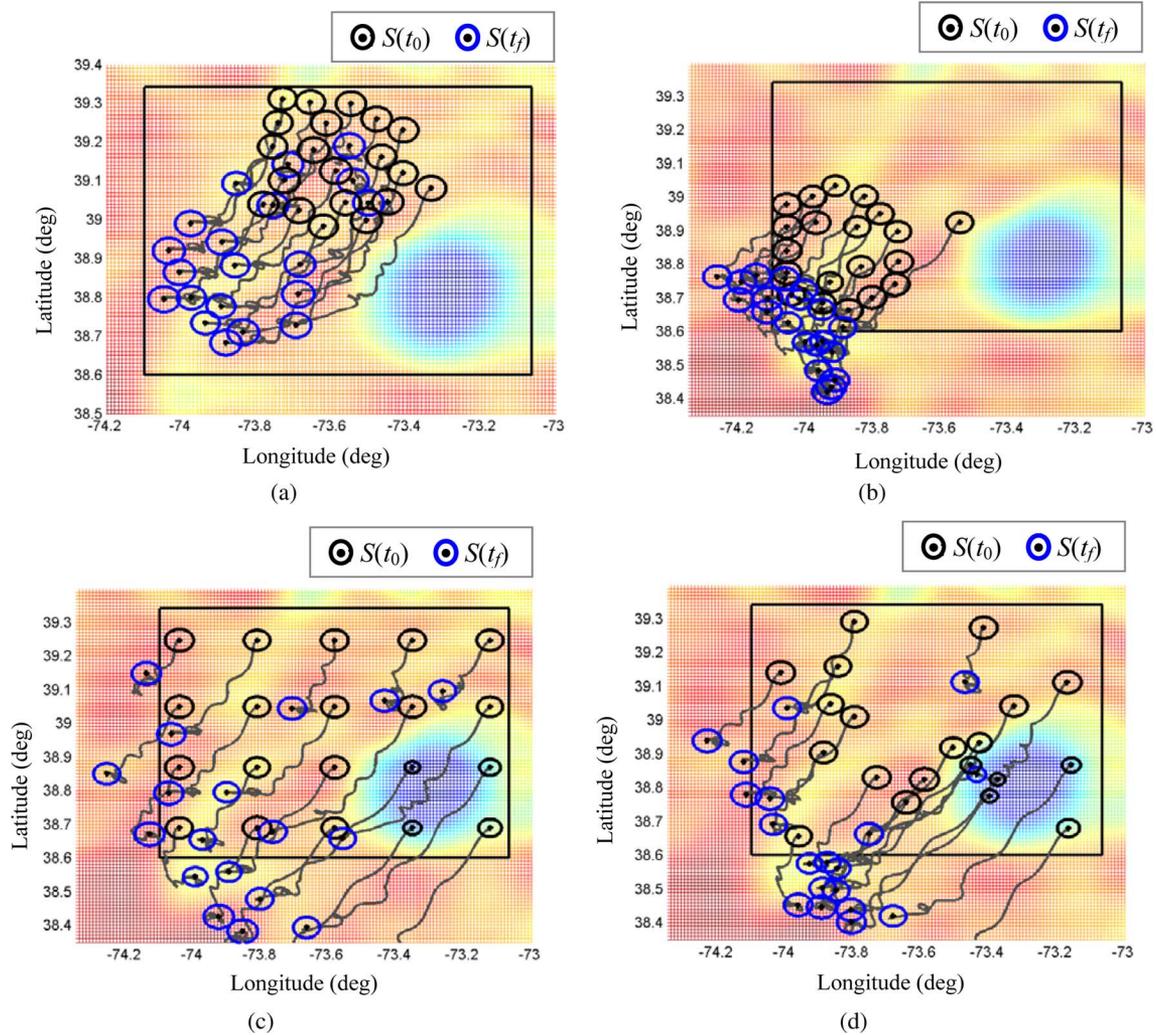


Fig. 14. State trajectories of the sensor networks in Fig. 15 initially placed by (a) OD, (b) static optimization, (c) grid, and (d) random techniques.

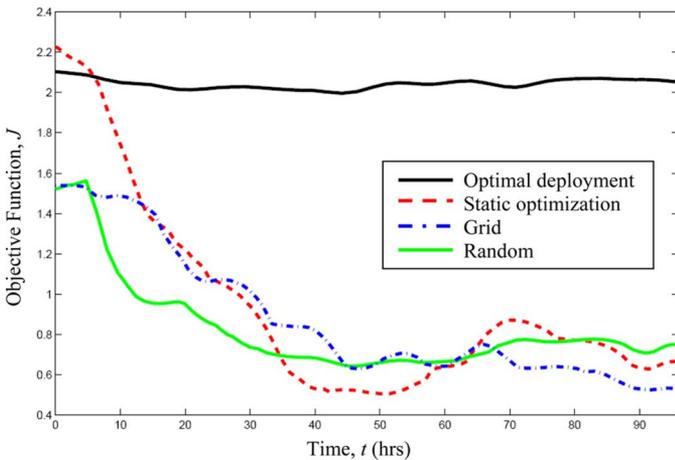


Fig. 15. Objective-function time history for different deployment strategies, $n = 20$, $k = 3$, and $\Delta T = 4$ days.

that of grid and random deployments. It can be seen that, by optimizing quality-of-service over time, the OD approach can effectively prolong the operability of a sensor network, without any changes to the sensor design or instrumentation.

B. Performance Robustness Analysis Results

In this section, the robustness of the OD approach is analyzed using the approach described in Section V. A MC simulation is used to propagate uncertainties over time, by sampling Gaussian models of errors in the sensors' initial positions, \mathbf{n}_ϵ , current velocities, \mathbf{v}_ϵ , and sensors' ranges, \mathbf{r}_ϵ . The means and standard deviations of the simulated errors are chosen based on expert knowledge [22], [25], [52], and are listed in Table II. The standard deviation of \mathbf{v}_ϵ is computed as a percentage of the local velocity magnitude [22]. Based on the approach in Section V, a total of $M = 10\,000$ evaluations are conducted in order to obtain a representative sample of the stochastic sensor state and performance history from the MC simulation. From this sample, mean performance metrics and confidence levels are computed and compared to their zero-error or *nominal* counterparts. As an example, in Fig. 16, the results of one MC evaluation are compared to the nominal performance and sensor trajectories, which are obtained by the OD approach for a network with $n = 10$, and $k = 2$. As can be expected, Fig. 16(b) shows that, in the presence of the Gaussian errors listed in Table II, all three performance measures are slightly decreased because the sensors have deviated from the nominal OD trajectories [Fig. 16(a)].

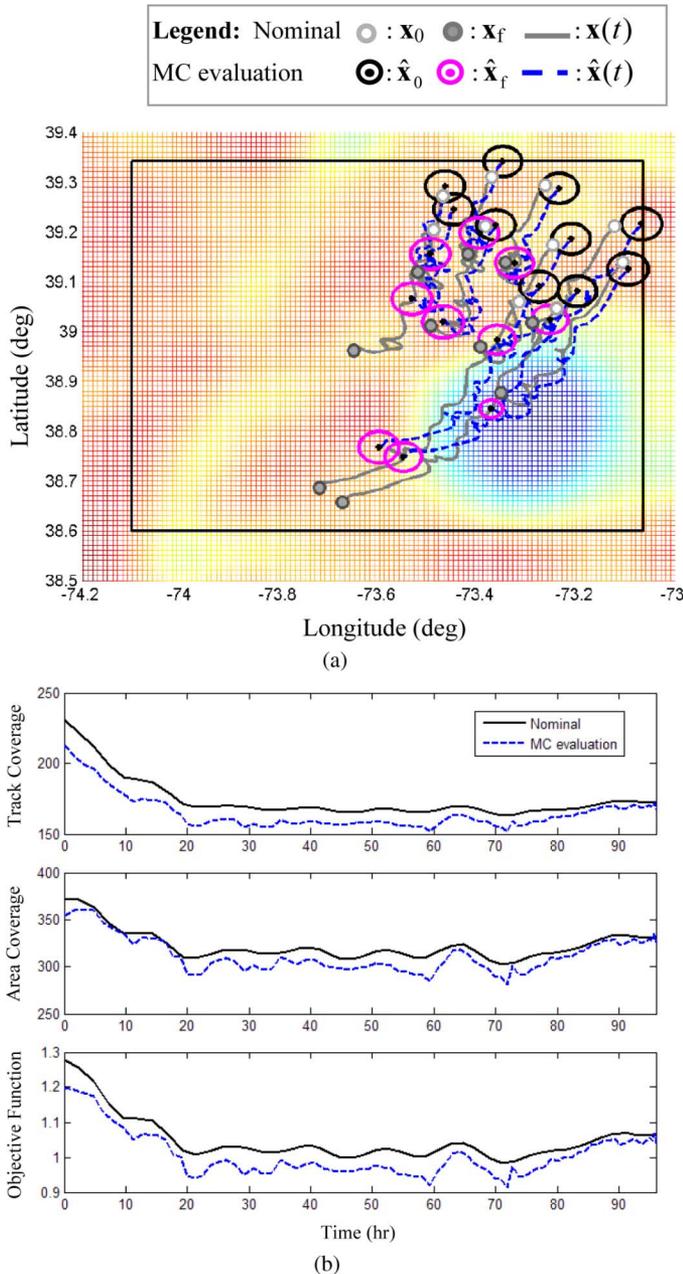


Fig. 16. (a) The state trajectories and (b) performance of one MC evaluation are compared to their nominal counterparts, for $n = 10$ and $k = 2$.

TABLE II
SIMULATED GAUSSIAN ERROR MODELS

Errors	Standard deviation	Mean
\mathbf{d}_ϵ	1.7 Km	± 5 Km
\mathbf{n}_ϵ	1.7 m	0
\mathbf{r}_ϵ	75 m	0
\mathbf{v}_ϵ	5% Km/hr	0

Trajectory and performance envelopes are obtained by plotting the results of all M evaluations with respect to time, to determine numerical bounds on these quantities in the presence of propagated uncertainties. The effects of uncertainties on the initial and final sensor positions are shown in Fig. 17 by plotting the frequency of occurrence of $\hat{\mathbf{x}}_0$ and $\hat{\mathbf{x}}_f$ for two networks that

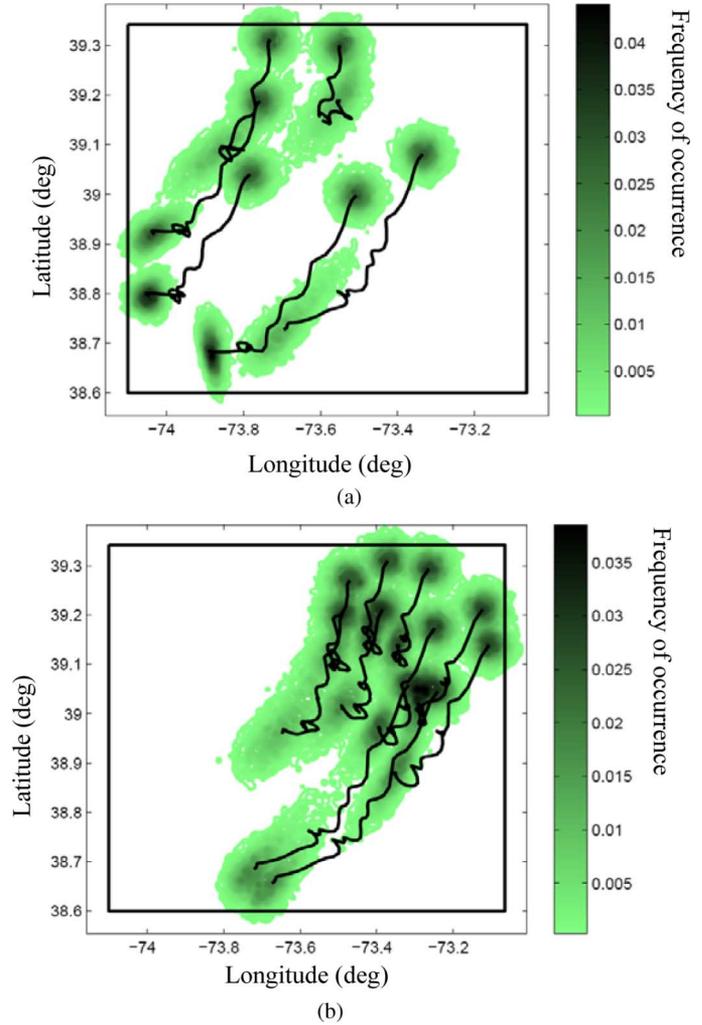


Fig. 17. Frequency of occurrence of $\hat{\mathbf{x}}_0$ and $\hat{\mathbf{x}}_f$ for (a) five and (b) ten sensors selected from two OD networks with $n = 20$ and $n = 10$, respectively, ($k = 3$).

are deployed using OD, and are subject to the errors \mathbf{n}_ϵ , \mathbf{v}_ϵ , and \mathbf{r}_ϵ . The envelopes of two sensor trajectories selected from a network with $n = 20$ and $k = 3$, shown in Fig. 18(a), illustrate that when the aforementioned errors are propagated through time the sensors' trajectories, like the initial and final positions, can deviate significantly from their nominal counterparts. Also, as a result, the quality-of-service of the sensor network is a random variable with a time-varying PDF. Therefore, evolution of the probability of track coverage is illustrated in Fig. 18(b)–(d) by evaluating and plotting \hat{T}_A^k in parameter space, at three sample moments in time (using the approach in Section VII-B). The performance envelopes plotted in Fig. 19 represent the numerical bounds on the sensor network coverage, and show that even in the presence of propagated uncertainties networks deployed by the OD approach significantly outperform networks deployed by static-optimization, grid, or random strategies.

In Table III, the nominal performance measures are compared to the mean performance measures that are computed by propagating the errors in Table II. It can be seen that the maximum average decrease in overall performance is 6.04%. Suppose the acceptable decrease in performance is 5% (i.e., $\rho = 0.95$). Then, the estimated probability of adequate coverage, $\hat{\mathbb{P}}$, varies from

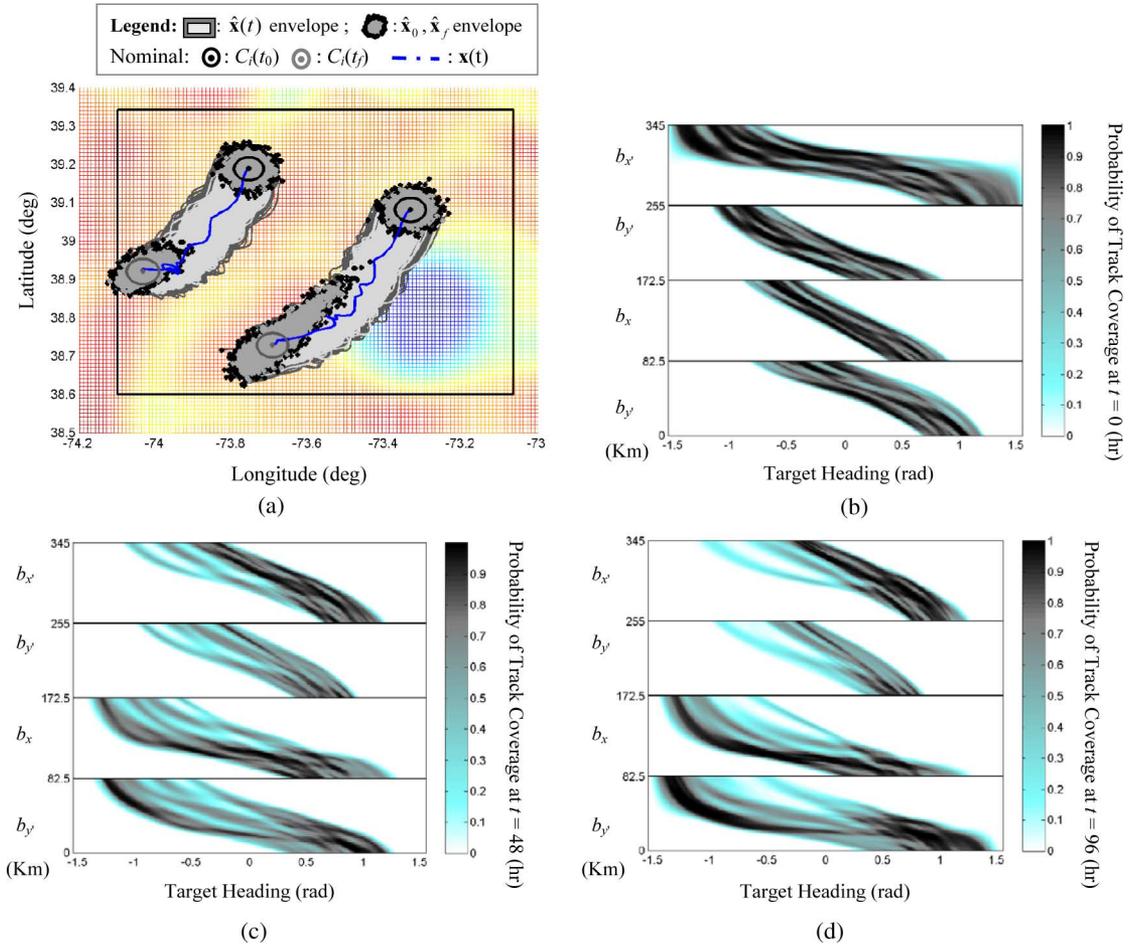


Fig. 18. Examples of state envelopes (a), and probability of track coverage in parameter space at $t = 0$ hr (b), $t = 48$ hr (c), and $t = 96$ hr (d), for an OD network with $n = 20$ and $k = 3$.

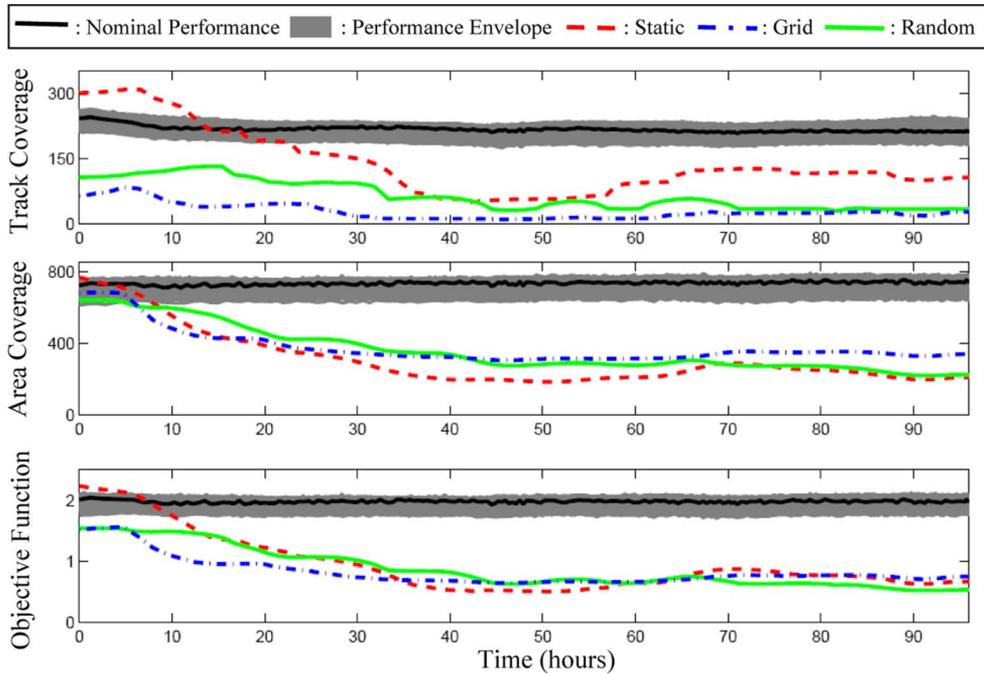


Fig. 19. Performance envelopes for an OD network with $n = 20$ and $k = 3$.

0.8524 to 0.9868 (Table III). The narrow 95% confidence intervals in Table III indicate that \hat{J} and $\hat{\mathbb{P}}$ are estimated with high

confidence from the MC evaluations (Section V). Based on these results, it can be concluded that the performance of the OD ap-

TABLE III
PERFORMANCE ROBUSTNESS RESULTS

(n, k) :	Performance Measures	Nominal Value	Mean Value (% Difference)	$\hat{\mathbb{P}}$	95% Confidence Interval
(10,2)	Track Coverage	$1.671 \cdot 10^4$	$1.5090 \cdot 10^4$	0.8524	$(1.507, 1.511) \cdot 10^4$
	Area Coverage	$3.091 \cdot 10^4$	$3.064 \cdot 10^4$	0.9366	$(3.061, 3.066) \cdot 10^4$
	Objective Function	100.5	96.29 (-4.19%)	0.9468	(96.21, 96.38)
(15,3)	Track Coverage	$1.630 \cdot 10^4$	$1.472 \cdot 10^4$	0.8730	$(1.471, 1.474) \cdot 10^4$
	Area Coverage	$5.368 \cdot 10^4$	$5.110 \cdot 10^4$	0.9484	$(5.106, 5.114) \cdot 10^4$
	Objective Function	145.6	136.8 (-6.04%)	0.9726	(136.7, 136.9)
(20,3)	Track Coverage	$2.245 \cdot 10^4$	$2.048 \cdot 10^4$	0.9426	$(2.046, 2.050) \cdot 10^4$
	Area Coverage	$7.174 \cdot 10^4$	$6.892 \cdot 10^4$	0.9768	$(6.888, 6.896) \cdot 10^4$
	Objective Function	196.1	185.9 (-5.20%)	0.9868	(185.8, 186.0)

proach is robust and, therefore, OD can potentially be used to deploy sensor networks even in the presence of modeling and measurement errors.

VIII. SUMMARY AND CONCLUSION

The problem of cooperative track detection by a dynamic sensor network arises in many applications, including security and surveillance, environmental and atmospheric monitoring, and tracking of endangered species. Several authors have recently shown that the quality-of-service of these networks can be optimized by placing the sensors in the ROI via static optimization. However, if the sensors are subject to external forcing, such as, winds or currents, they may be rapidly displaced and experience a significant loss in coverage and tracking performance. This paper presents an OD approach that treats the sensor network as a dynamic system, and optimizes its quality-of-service over time by a direct-shooting method that transcribes the dynamic optimization problem into an NLP. An integral objective function is derived by a discrete computational geometry approach that represents the area and track coverage of the sensor network by unions of circular segments and cones of line transversals (stabbers), respectively.

The OD approach is demonstrated through an ocean sensor network application in which simulated networks of sonobuoys, with no on-board controls, are deployed to track and detect moving targets in the ROI near the New Jersey coast. The sensor network's equation of motion is obtained by combining a sonobuoy field drift model with a neural model of the current velocity forecast. A model of effective range as a function of position is obtained by training a Bayesian network with a range-dependent parabolic-equation model to perform geoacoustic inversion. The numerical results show that the OD approach improves quality-of-service by up to a factor of five compared to static optimization, grid, or random deployments. The possibility of inadequate performance in the face of uncertainty is investigated by SRA. In this approach, uncertainties are propagated through time by a MC simulation, and the probability of adequate performance is obtained along with its confidence level. These performance robustness criteria show that the OD approach is robust to modeling and deployment errors, and the probability that the performance decreases by less than 5% is higher than 0.8524. The results also show that by maximizing the quality-of-service over time, the approach effectively prolongs the operability of the network, for example

delaying the need for replenishing the network, thus leading to sensor networks that are considerably more cost effective without changes to the sensor design or instrumentation.

APPENDIX I COVERAGE CONE

Let $\mathcal{R}_\theta(b_y)$ denote a ray that intersects a circle $C_i(t) = C_i[\mathbf{x}_i(t), r_i(t)]$ with radius $r_i(t)$, and centered at $\mathbf{x}_i(t)$ at time $t \in \Delta T$, in \mathbb{R}_+^2 . Consider any two points that lie on $\mathcal{R}_\theta(b_y)$ and inside $C_i(t)$, and let $\mathbf{u}_1, \mathbf{u}_2 \in \mathbb{R}_+^2$ denote their positions relative to the origin b_y of the coverage cone $K[C_i(t), b_y]$. By construction, $\mathbf{u}_1, \mathbf{u}_2 \in C_i(t)$, and a vector \mathbf{z} joining the two points will lie on the ray $\mathcal{R}_\theta(b_y)$. Let c_1 and c_2 denote any two positive constants. By definition of vector sum and subtraction [33], if $\mathbf{z} = c_1\mathbf{u}_1 + c_2\mathbf{u}_2$ then \mathbf{z} has the same origin as \mathbf{u}_1 and \mathbf{u}_2 . Thus, since \mathbf{z} lies on $\mathcal{R}_\theta(b_y)$, $\mathcal{R}_\theta(b_y)$ intercepts the y axis at the cone's origin b_y . If $\mathbf{z} = \pm c_1\mathbf{u}_1 \mp c_2\mathbf{u}_2$, \mathbf{z} does not have the same origin as \mathbf{u}_1 and \mathbf{u}_2 and, thus, $\mathcal{R}_\theta(b_y)$ does not intercept the y axis at b_y . By definition (7), $K[C_i(t), b_y]$ is the set of all nonnegative combinations of the elements in $C_i(t)$. Since \mathbf{u}_1 and \mathbf{u}_2 are two elements in $C_i(t)$, and any nonnegative combination of these two elements can be written as $c_1\mathbf{u}_1 + c_2\mathbf{u}_2$, with $c_1, c_2 > 0$, it follows that $\mathbf{z} = c_1\mathbf{u}_1 + c_2\mathbf{u}_2 \in K[C_i(t), b_y]$. In the trivial case of $\mathbf{u}_1 = \mathbf{u}_2$, when only one point lies on $\mathcal{R}_\theta(b_y)$ and inside $C_i(t)$, it follows that $\mathcal{R}_\theta(b_y)$ is tangential to $C_i(t)$. In this case, it also follows that $\mathbf{z} = c\mathbf{u}_1 \in K[C_i(t), b_y]$ for any $c > 0$, because \mathbf{z} belongs to an extremal of $K[C_i(t), b_y]$. Finally, since $\mathcal{R}_\theta(b_y)$ denotes any ray with intercept b_y that intersects $C_i(t)$ in \mathbb{R}_+^2 , and $\mathbf{z} = c_1\mathbf{u}_1 + c_2\mathbf{u}_2$ provided $\mathcal{R}_\theta(b_y)$ intersects the y axis at b_y , it also follows that any $\mathcal{R}_\theta(b_y)$ that intersects both $C_i(t)$ and the y axis at b_y is contained by $K[C_i(t), b_y]$ in (7).

APPENDIX II PROOF OF PROPOSITION 4.1

This proof considers a family of $k = 3$ nontranslate circles $S_k(t) = \{C_i(t), C_j(t), C_l(t)\}$ with index set $I_{S_k} = \{i, j, l\}$. The results can be extended to higher k by induction. The coverage cone $K[C_l(t), b_y]$ contains the set of all rays that intersect

$C_\ell(t)$ in \mathbb{R}_+^2 , at time t , where $\ell \in I_{S_k}$. Then, from set theory, the set of tracks intersecting all circles in the family $S_k(t)$ is given by the following intersection:

$$\begin{aligned} K_k[S_k(t), b_y] &= \bigcap_{\ell \in I_{S_k}} K[C_\ell(t), b_y] \\ &= K[C_i(t), b_y] \cap K[C_j(t), b_y] \cap K[C_l(t), b_y]. \end{aligned} \quad (46)$$

From the properties of cones ([32, p. 70]), the intersection of a collection of cones is also a cone and, thus, $K_k[S_k(t), b_y]$ is a cone. A vector \mathbf{z} representing a ray \mathcal{R}_θ with the same slope and origin lies in a cone K if and only if \mathcal{R}_θ lies in K , since any point on \mathcal{R}_θ can be written as $c\mathbf{z}$, with $c > 0$.

Consider a ray $\mathcal{R}_\theta \in K[C_\ell(t), b_y]$, where $K[C_\ell(t), b_y] = \text{cone}(\hat{\mathbf{l}}_\ell, \hat{\mathbf{h}}_\ell)$, and thus can be represented by a vector $\mathbf{z}_\ell = c_1\hat{\mathbf{l}}_\ell + c_2\hat{\mathbf{h}}_\ell$ with constants $c_1, c_2 > 0$. Then, $\mathbf{z}_\ell \in K[C_\ell(t), b_y]$ and, by the properties of vector sum, $\hat{\mathbf{l}}_\ell \prec \mathbf{z}_\ell \prec \hat{\mathbf{h}}_\ell$. Next, consider a cone $K^* = \text{cone}(\hat{\mathbf{l}}^*, \hat{\mathbf{h}}^*)$ that is finitely generated by two unit vectors $\hat{\mathbf{h}}^* = \hat{\mathbf{h}}_j$ and $\hat{\mathbf{l}}^* = \hat{\mathbf{l}}_i$ with $j, i \in I_{S_k}$, and assume $\hat{\mathbf{l}}_i \prec \hat{\mathbf{h}}_j$. By the properties of finitely generated cones [32], defined as in (7), any vector $\mathbf{z}^* = b_1\hat{\mathbf{l}}^* + b_2\hat{\mathbf{h}}^*$ with constants $b_1, b_2 > 0$ must lie in K^* . It follows that a ray \mathcal{R}_θ^* with the same slope and origin as \mathbf{z}^* must also lie in K^* , since any point on \mathcal{R}_θ^* can be written as $c\mathbf{z}^*$, with $c > 0$. Since \mathbf{z}^* is a linear combination of $\hat{\mathbf{l}}^*$ and $\hat{\mathbf{h}}^*$, it also follows that $\hat{\mathbf{l}}^* \prec \mathbf{z}^* \prec \hat{\mathbf{h}}^*$.

According to Proposition 4.1, choose $\hat{\mathbf{h}}^* = \hat{\mathbf{h}}_j \succeq \hat{\mathbf{h}}_\ell$ and $\hat{\mathbf{l}}^* = \hat{\mathbf{l}}_i \succeq \hat{\mathbf{l}}_\ell \forall \ell \in I_{S_k}$. Suppose the unit vectors of $S_k(t)$ can be ordered as $\hat{\mathbf{h}}_i \prec \hat{\mathbf{h}}_j \prec \hat{\mathbf{h}}_i$ and $\hat{\mathbf{l}}_i \prec \hat{\mathbf{l}}_l \prec \hat{\mathbf{l}}_j$. Then, the unit vectors and \mathbf{z}^* can be ordered as follows:

$$\begin{aligned} \hat{\mathbf{l}}_\ell \preceq \hat{\mathbf{l}}_j = \hat{\mathbf{l}}^* \prec \mathbf{z}^* \prec \hat{\mathbf{h}}^* = \hat{\mathbf{h}}_i \preceq \hat{\mathbf{h}}_\ell \\ \forall \ell \in \{i, j, l\} = I_{S_k} \end{aligned} \quad (47)$$

or, more explicitly

$$\hat{\mathbf{l}}_i \prec \hat{\mathbf{l}}_l \prec \hat{\mathbf{l}}_j = \hat{\mathbf{l}}^* \prec \mathbf{z}^* \prec \hat{\mathbf{h}}^* = \hat{\mathbf{h}}_i \prec \hat{\mathbf{h}}_j \prec \hat{\mathbf{h}}_i. \quad (48)$$

Since the above order also implies $\hat{\mathbf{l}}_\ell \prec \mathbf{z}^* \prec \hat{\mathbf{h}}_\ell, \forall \ell \in I_{S_k}$, then $\mathbf{z}^*, \mathcal{R}_\theta^* \in K[C_\ell(t), b_y] \forall \ell \in I_{S_k}$. Thus, from (46), $\mathbf{z}^*, \mathcal{R}_\theta^* \in K_k[S_k(t), b_y] = K^* = \text{cone}(\hat{\mathbf{l}}^*, \hat{\mathbf{h}}^*)$, provided $\hat{\mathbf{h}}^*$ and $\hat{\mathbf{l}}^*$ are chosen subject to (47).

So far, it was assumed that $\hat{\mathbf{l}}_i \prec \hat{\mathbf{h}}_j$. If the unit vectors of $S_k(t)$ are such that $\hat{\mathbf{l}}_i \succ \hat{\mathbf{h}}_j$, then there are no vectors that can satisfy the order $\hat{\mathbf{l}}_i = \hat{\mathbf{l}}^* \prec \mathbf{z}^* \prec \hat{\mathbf{h}}^* = \hat{\mathbf{h}}_j$, and $K_k[S_k(t), b_y] = K^* = \emptyset$.

APPENDIX III

DERIVATION OF TRACK COVERAGE FUNCTION

First, we seek a Lebesgue measure μ on the set of tracks that are detected by at least k sensors in $S(t)$ at time t given by

$$\mathcal{K}_k[S(t), b_y] = \bigcup_{j=1}^q K_k[S_k^j(t), b_y], \quad q = \binom{n}{k} \quad (49)$$

where $S_k^j(t)$ denotes the j th k -subset of $S(t)$ at time t , and the number q of possible k -subsets is given by the binomial coef-

ficient n choose k , as shown in (49). Since \mathcal{K}_k is a union of possibly disjoint cones, defined in (8), it may not be a cone [32], and is computed using the principle of inclusion-exclusion [80], such that

$$\begin{aligned} \mu\{\mathcal{K}_k[S(t), b_y]\} &= \mu\left\{\bigcup_{j=1}^q K_k[S_k^j(t), b_y]\right\} \\ &= \sum_{j=1}^q (-1)^{j+1} \sum_{1 \leq i_1 < \dots < i_j \leq q} \mu\left\{K_k[S_k^{i_1}(t), b_y] \cap \dots \cap K_k[S_k^{i_j}(t), b_y]\right\} \end{aligned} \quad (50)$$

where

$$q = \binom{n}{k} = \frac{n!}{(n-k)!k!}, \quad \text{and} \quad \sum_{1 \leq i_1 < \dots < i_j \leq q}$$

is a sum over all the $[q!/(q-j)!j!]$ distinct integer j -tuples (i_1, \dots, i_j) satisfying $1 \leq i_1 < \dots < i_j \leq q$. Also, $\mu\{\cdot\}$ denotes a measure on the set. Since the right-hand side of (50) is an intersection of cones, it also is a cone on which we can impose the Lebesgue measure μ .

Now, consider the intersection of cones $K_k[S_k^{i_1}(t), b_y] \cap \dots \cap K_k[S_k^{i_j}(t), b_y]$ inside the inner summation in (50). $S_k^{i_l}(t)$ denotes the i_l th k -subset of $S(t)$ at time t , where i_l is a positive integer between 1 and $i_j \leq q$, and q is the total number of k -subsets in $S(t)$. By the properties of cones, this intersection is also a cone, and represents the set of tracks through b_y that intersect all sensors in the family $\{S_k^{i_1}(t) \cup \dots \cup S_k^{i_j}(t)\} \equiv S_p^{i_1, \dots, i_j}(t)$. Based on the properties of k -subsets, this set must contain $k \leq p \leq n$ elements of $S(t)$ and, thus, is a p -subset of $S(t)$. From Proposition 4.1, the set of line transversals of $S_p^{i_1, \dots, i_j}(t)$ through b_y can be represented by the p -coverage cone $K_p[S_k^{i_1}(t) \cup \dots \cup S_k^{i_j}(t), b_y]$. Using the Lebesgue measure (10) on k -coverage cones, (50) can be written as

$$\begin{aligned} \mu\{\mathcal{K}_k[S(t), b_y]\} &= \sum_{j=1}^q (-1)^{j+1} \sum_{1 \leq i_1 < \dots < i_j \leq q} \mu\left\{K_p[S_k^{i_1}(t) \cup \dots \cup S_k^{i_j}(t), b_y]\right\} \end{aligned} \quad (51)$$

$$\begin{aligned} &= \sum_{j=1}^q (-1)^{j+1} \sum_{1 \leq i_1 < \dots < i_j \leq q} \psi\left[S_k^{i_1}(t) \cup \dots \cup S_k^{i_j}(t), b_y\right] \end{aligned} \quad (52)$$

$$\begin{aligned} &= \sum_{j=1}^q (-1)^{j+1} \sum_{1 \leq i_1 < \dots < i_j \leq q} \psi\left[S_p^{i_1, \dots, i_j}(t), b_y\right] \end{aligned} \quad (53)$$

where p is the number of elements in the union of j k -subsets of $S(t)$, and $\psi[\cdot]$ is given by (10).

The set of tracks that traverse \mathcal{A} and are detected by at least k sensors is given by the union of all k -coverage cones with origins $b_y, b_x, b_{y'}, b_{x'} \in \partial\mathcal{A}$, and with opening angles denoted by ψ, ζ, φ , and ρ , respectively (Fig. 3). In order to obtain representations that are computationally tractable, $\partial\mathcal{A}$

is discretized into increments of size δb , and $2(L_1 + L_2)/\delta b$ intercept values, indexed by ℓ , are considered. Using an appropriate transformation [13], (5)–(10) can be used to compute all opening angles, $\psi[S_p^{i,j}(t), b_y]$, $\rho[S_p^{i,j}(t), b_x]$, $\xi[S_p^{i,j}(t), b_{y'}]$, and $\zeta[S_p^{i,j}(t), b_{x'}]$, as a function of their origins and of the sensors' positions, $\mathbf{x}(t)$, and ranges, $\mathbf{r}(t)$, in the xy -frame. By considering that every track in this union intersects two sides of \mathcal{A} and belongs to two k -coverage cones, and by shifting the indices in order to consider intercepts at the corners only once, the final track coverage function in (12) is obtained.

APPENDIX IV

DERIVATION OF AREA COVERAGE FUNCTION

Let $A_0(t)$ denote the instantaneous area coverage all sensors in $S(t) = \{C_1(t), \dots, C_n(t)\}$ with $\mathbf{x}_i(t) \in \mathcal{A}$ at time t , assuming $C_i(t) \cap C_j(t) = \emptyset$ for $\forall i \neq j$, and $C_i(t) \cap \partial\mathcal{A} = \emptyset$ for $\forall i$. Then, when these assumptions are violated (Fig. 4), the effective area coverage $A_C(t)$ is given by (15), where $A_s(t)$ and $A_p(t)$ are the instantaneous reductions due to $C_i(t) \cap C_j(t) \neq \emptyset$ and $C_i(t) \cap \partial\mathcal{A} \neq \emptyset$ for some i, j , respectively. When $C_i(t) \cap C_j(t) \neq \emptyset$, the inequality

$$\begin{aligned} d_{ij}(t) &\equiv \|\mathbf{x}_i(t) - \mathbf{x}_j(t)\| \\ &= \sqrt{[x_i(t) - x_j(t)]^2 + [y_i(t) - y_j(t)]^2} \\ &< r_i(t) + r_j(t) \end{aligned} \quad (54)$$

holds, where $\|\cdot\|$ denotes the Euclidian norm. The area of the circular segment A_{ij}^c representing the area-coverage reduction experienced by the i th sensor due to $C_i(t) \cap C_j(t) \neq \emptyset$ (Fig. 6), can be written as (16) in terms of the effective range $r_i(t)$ and the central angle [36]

$$\theta_{ij}(t) = 2 \cos^{-1} \left[\frac{d_{ij}(t)}{r_i(t)} \right] \quad (55)$$

defined in Fig. 6, where d_i is defined in Fig. 5, and can be written in terms of the sensors' effective ranges and positions by applying the law of cosines to the triangles in Fig. 7

$$\begin{aligned} d_i(t) &= r_i(t) \sin \left[\frac{\omega_{ij}(t)}{2} \right] \\ &= r_i(t) \sin \left\{ \frac{1}{2} \cos^{-1} \left[\frac{r_i^2(t) + r_j^2(t) - d_{ij}^2(t)}{2r_i(t)r_j(t)} \right] \right\}. \end{aligned} \quad (56)$$

Then, by substituting (56) into (55), the central angle (17) is obtained, and can be substituted in (16) to obtain A_{ij}^c as a function of the sensors' effective ranges and positions. By multiplying A_{ij}^c by the Heaviside function $H[r_i(t) + r_j(t) - d_{ij}(t)]$, the area-coverage reduction is zero anytime the inequality (54) is violated. The total area-coverage reduction of the sensor network is obtained by considering all n sensors in $S(t)$ pairwise, i.e.,

$$A_s(t) = \sum_{\substack{i=1 \\ j \neq i}}^n H[r_i(t) + r_j(t) - d_{ij}(t)] A_{ij}^c(t). \quad (57)$$

The above computation can be reduced by recognizing that when the indices i and j are exchanged in (16), it can be easily shown that

$$A_{ji}^c(t) = \frac{r_j^2(t)}{r_i^2(t)} A_{ij}^c(t). \quad (58)$$

Thus, the reduction A_{ji}^c experienced by the j th sensor due to $C_i(t) \cap C_j(t) \neq \emptyset$ (Fig. 6) can be computed from A_{ij}^c , and the total area-coverage reduction can be obtained from (18) solely as a function of $r_1(t), \dots, r_n(t)$, and $\mathbf{x}_1(t), \dots, \mathbf{x}_n(t)$.

When $C_i(t) \cap \partial\mathcal{A} \neq \emptyset$, the area-coverage reduction experienced by the i th sensor is also given by the area of a circular segment with the central angle (55), and the geometry shown in Fig. 5. However, in this case, d_i , now represents the distance between \mathbf{x}_i and the side of \mathcal{A} intersected by C_i . Adopting the reference frames in Fig. 3, let the distance d_{ik} denote the distance between \mathbf{x}_i and the axes indexed by $k = x, y, x', y'$, such that $d_{ix} = y_i, d_{iy} = x_i, d_{ix'} = (L_2 - y_i)$, and $d_{iy'} = (L_1 - x_i)$. Then, the area-coverage reduction experienced by the i th sensor due to an intersection with the k th axes is

$$\begin{aligned} A_{ik}^c(t) &= \frac{1}{2} r_i^2(t) [\theta_{ik}(t) - \sin \theta_{ik}(t)] \\ &= r_i^2(t) \left\{ \cos^{-1} \left[\frac{d_{ik}(t)}{r_i(t)} \right] - \frac{1}{2} \sin \left[2 \cos^{-1} \left(\frac{d_{ik}(t)}{r_i(t)} \right) \right] \right\}. \end{aligned} \quad (59)$$

A_{ik}^c is multiplied by the Heaviside function $H(r_i - d_{ik})$, such that the area-coverage reduction is zero when C_i does not intersect the k th axes. Thus, the total area-coverage reduction can be written solely as a function of $r_1(t), \dots, r_n(t)$, and $\mathbf{x}_1(t), \dots, \mathbf{x}_n(t)$, as shown in (19), by summing (59) over all values of n and k .

ACKNOWLEDGMENT

We gratefully acknowledge Dr. J. T. Kohut at the Coastal Ocean Observation Laboratory, Rutgers University, for providing the CODAR data used in this research, Dr. W. J. Fox at the Applied Physics Laboratory, University of Washington, for providing the range-dependent PE model, and Dr. D. S. Hammond at the Naval Air Warfare Center Aircraft Division for his helpful guidance and suggestions on the SFDM.

REFERENCES

- [1] D. A. Paley, F. Zhang, and N. E. Leonard, "Cooperative control for ocean sampling: The glider coordinated control system," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 4, pp. 735–744, 2008.
- [2] M. Parlange, W. Eichinger, and J. Albertson, "Regional scale evaporation and the atmospheric boundary layer," *Rev. Geophys.*, vol. 33, no. 11, pp. 99–124, 1995.
- [3] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein, "Energy efficient computing for wildlife tracking: Design trade-offs and early experiences with zebnet," in *Proc. 10th Int. Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS-X)*, 2002, pp. 96–107.
- [4] J. R. Siebert and J. R. Nielsen, *Electronic Tagging and Tracking in Marine Fisheries*. Dordrecht, The Netherlands: Kluwer, 2001.
- [5] V. Isler, S. Khanna, J. Spletzer, and C. Taylor, "Target tracking with distributed sensors: The focus of attention problem," *Comput. Vision and Image Understanding*, vol. 100, pp. 225–247, 2005.

- [6] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "Challenges: Building scalable mobile underwater wireless sensor networks for aquatic applications," *IEEE Network*, pp. 12–18, 2006.
- [7] E. Cayirci, H. Tezcan, Y. Dogan, and V. Coskun, "Wireless sensor networks for underwater surveillance systems," *Ad Hoc Networks*, vol. 4, pp. 431–446, 2006.
- [8] I. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: Research challenges," *Ad Hoc Networks*, vol. 3, pp. 257–279, 2005.
- [9] D. Hammond, Sonobuoy field drift prediction Naval Air Warfare Center Aircraft Division, Patuxent River, MD, Tech. Rep. NAW-CADPAX/TR-2004/223, Jan. 2005.
- [10] P. Atkins, "Tutorial introduction and historical overview of the need for heading sensors in sonar applications," *IEE Colloquium on Heading Sensors for Sonar and Marine Applications*, 1994.
- [11] J.-P. LeCadre and G. Souris, "Searching tracks," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 36, no. 4, pp. 1149–1166, 2000.
- [12] S. Ferrari, "Track coverage in sensor networks," in *Proc. Amer. Control Conf.*, 2006, pp. 1–10.
- [13] K. C. Baumgartner and S. Ferrari, "A geometric transversal approach to analyzing track coverage in sensor networks," *IEEE Trans. Comput.*, vol. 57, no. 8, pp. 1113–1128, 2008.
- [14] T. A. Wettergren, "Performance of search via track-before-detect for distributed sensor networks," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 1, pp. 314–325, 2008.
- [15] V. Isler, S. Khanna, and K. Daniilidis, "Sampling based sensor-network deployment," in *Pro. IEEE/R SJ Int. Conf. Intell. Robots Syst.*, 2004, vol. 100, pp. 1780–1785.
- [16] L. Benyuan and D. Towsley, "A study of the coverage of large-scale sensor networks," in *Proc. IEEE Int. Conf. Mobile Ad-hoc and Sensor Syst.*, 2004, pp. 475–483.
- [17] N. Heo and P. K. Varshney, "Energy-efficient deployment of intelligent mobile sensor networks," *IEEE Trans. Syst., Man, Cybern.-Part A: Systems and Humans*, vol. 35, no. 1, pp. 78–92, Jan. 2005.
- [18] T. A. Wettergren, R. L. Streit, and J. R. Short, "Tracking with distributed sets of proximity sensors using geometric invariants," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 40, no. 4, pp. 1366–1374, Oct. 2004.
- [19] M. Marengoni, B. A. Draper, A. Hanson, and R. A. Sitaraman, "System to place observers on a polyhedral terrain in polynomial time," *Image and Vision Comput.*, vol. 18, pp. 773–780, 1996.
- [20] D. V. Hansen and A. Herman, "Temporal sampling requirements for surface drifting buoys in the tropical pacific," *J. Atmospheric Ocean. Technol.*, vol. 6, pp. 599–607, 1989.
- [21] M. D. Collins, R. J. Cederberg, D. B. King, and S. A. Chin-Bing, "Comparison of algorithms for solving parabolic wave equations," *J. Acoust. Soc. Amer.*, vol. 100, no. 1, pp. 178–182, 1996.
- [22] *COOL*, Rutgers University, 2002. [Online]. Available: <http://marine.rutgers.edu/>
- [23] K. Chakrabarty, S. 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, 2002.
- [24] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, and K. Saluja, "Sensor deployment for detection of targets traversing a region," *Mobile Networks and Applications*, vol. 8, pp. 453–461, Aug. 2003.
- [25] Z. Kone, E. G. Rowe, and T. A. Wettergren, "Sensor repositioning to improve undersea sensor field coverage," in *Proc. OCEANS 2007*, Sep. 2007, pp. 1–6.
- [26] K. C. Baumgartner, S. Ferrari, and A. Rao, "Optimal control of an underwater sensor network for cooperative target tracking," *IEEE J. Oceanic Eng.*, 2009, accepted, in print (2009).
- [27] B. Koopman, *Search and Screening: General Principles With Historical Applications*. New York: Pergamon, 1980.
- [28] H. Cox, "Cumulative detection probabilities for a randomly moving source in a sparse field of sensors," in *Proc. Asilomar Conf.*, 1989, pp. 384–389.
- [29] J. Goodman, R. Pollack, and R. Wenger, "Geometric transversal theory," in *New Trends in Discrete and Computational Geometry*, J. Pach, Ed. Berlin, Germany: Springer-Verlag, 1991, pp. 163–198.
- [30] D. Avis, J. Robert, and R. Wenger, "Lower bounds for line stabbing," *Inf. Process. Lett.*, vol. 33, pp. 59–62, 1989.
- [31] J. Urritia, "Art gallery and illumination problems," in *Handbook on Computational Geometry*, J. Sack and J. Urritia, Eds. New York: Elsevier, 1992, pp. 387–434.
- [32] D. P. Bertsekas, *Convex Analysis and Optimization*. Belmont, MA: Athena Scientific, 2003.
- [33] H. F. Davis and A. D. Snider, *Vector Analysis*. Dubuque, IA: William C. Brown, 1987.
- [34] S. Skiena, "Generating k -subsets," in *Implementing Discrete Mathematics: Combinatorics and Graph Theory With Mathematica*. Reading, MA: Addison-Wesley, 1990, pp. 44–46.
- [35] S. Ferrari, R. Fierro, B. Pertet, C. Cai, and K. Baumgartner, "A geometric optimization approach to detecting and intercepting dynamic targets using a mobile sensor network," *SIAM J. Control Optim.*, vol. 48, no. 1, pp. 292–320, 2009.
- [36] J. W. Harris and H. Stocker, "Segment of a circle," in *Handbook of Mathematics and Computational Science*. Berlin, Germany: Springer-Verlag, 1998, pp. 92–93.
- [37] K. A. C. Baumgartner and S. Ferrari, "Optimal placement of a moving sensor network for track coverage," in *Proc. 2007 Amer. Controls Conf.*, New York, pp. 4040–4046.
- [38] J. Betts, "Survey of numerical methods for trajectory optimization," *J. Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998.
- [39] R. F. Stengel, *Optimal Control and Estimation*. New York: Dover, 1986.
- [40] D. G. Hull, "Conversion of optimal control problems into parameter optimization problems," *J. Guidance, Control, and Dynamics*, vol. 20, no. 1, pp. 57–60, 1997.
- [41] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 2007.
- [42] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*. Hoboken, NJ: Wiley, 2006.
- [43] R. F. Stengel and L. R. Ray, "Stochastic robustness of linear time-invariant control systems," *IEEE Trans. Autom. Control*, vol. 36, no. 1, pp. 82–87, 1991.
- [44] Q. Wang and R. F. Stengel, "Robust control of nonlinear systems with parametric uncertainty," *Automatica*, vol. 38, pp. 1591–1599.
- [45] N. A. Lehtomaki, N. R. Sandell, and M. Athans, "Robustness results in linear quadratic Gaussian-based multivariable control design," *IEEE Trans. Autom. Control*, vol. 26, pp. 75–93, 1981.
- [46] D. D. Siljak, "Parameter space methods for robust control design," *IEEE Trans. Autom. Control*, vol. 34, pp. 674–688, 1989.
- [47] R. Y. Rubinstein, *Simulation and the Monte Carlo Method*. New York: Wiley, 1981.
- [48] T. G. Beckwith, R. D. Marangoni, and J. H. Lienhard, *Mechanical Measurements*. Reading, MA: Addison-Wesley, 1993.
- [49] S. E. Pazan, "Intercomparison of drogued and undrogued drift buoys," in *Proc. MTS/IEEE OCEANS'96. Prospects for the 21st Century*, 1996, vol. 2, no. 2, pp. 864–872.
- [50] D. S. Hammond and J. F. McEachern, "A model for horizontal line array deployment," in *Proc. OCEANS'93. Eng. Harmony With Ocean*, Victoria, BC, Canada, Oct. 1993, pp. 1317–1322.
- [51] A. R. Robinson, "Physical processes, field estimation and an approach to interdisciplinary ocean modeling," *Earth-Science Rev.*, vol. 40, pp. 3–54, 1996.
- [52] P. F. J. Lermusiaux, "Evolving the subspace of the three-dimensional multiscale ocean variability: Massachusetts bay," *J. Marine Syst.*, vol. 29, pp. 385–422, 2001.
- [53] A. Liu, Y. Zhao, and M.-K. Hsu, "Ocean surface drift revealed by synthetic aperture radar images," *EOS*, vol. 87, no. 24, pp. 233–239, Jun. 13, 2006.
- [54] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vols. I and II*. Belmont, MA: Athena Scientific, 1995.
- [55] F. Lekien, C. Coulliette, R. Bank, and J. Marsden, "Open-boundary modal analysis: Interpolation, extrapolation and filtering," *J. Geophys. Res.-Oceans*, vol. 109, no. C12, pp. 1–13, 2004.
- [56] T. Inanc, S. C. Shadden, and J. E. Marsden, "Optimal trajectory generation in ocean flows," in *Proc. 2005 Amer. Control Con.*, Portland, OR, pp. 674–679.
- [57] T. Lyche, K. Mørken, and E. Quak, "Theory and algorithms for nonuniform spline wavelets," in *Multivariate Approximation and Applications*, N. Dyn, D. Levitan, D. Levin, and A. Pinkus, Eds. Cambridge, U.K.: Cambridge Univ. Press, 2001.
- [58] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*. Belmont, MA: Athena Scientific, 1996.
- [59] J. H. Friedman, "Multivariate adaptive regression splines," *The Annals of Statistics*, vol. 19, pp. 1–141, 1991.
- [60] F. Girosi, M. Jones, and T. Poggio, "Regularization theory and neural networks architectures," *Neural Computation*, vol. 7, no. 2, pp. 219–269, 1995.
- [61] D. MacKay, "Bayesian interpolation," *Neural Computation*, vol. 4, no. 3, pp. 415–447, 1992.
- [62] Mathworks, *MATLAB Neural Network Toolbox*, function: trainbr, 2006. [Online]. Available: <http://www.mathworks.com>

- [63] F. Foresee and M. Hagan, "Gauss-Newton approximation to Bayesian regularization," in *Proc. 1997 Int. Joint Conf. Neural Networks*, 1997, pp. 1930–1935.
- [64] *Matlab* 2004. [Online]. Available: <http://www.mathworks.com>
- [65] B. B. Thompson, R. J. Marks, M. A. El-Sharkawi, W. J. Fox, and R. T. Miyamoto, "Inversion of neural network underwater acoustic model for estimation of bottom parameters using modified particle swarm optimizer," in *Proc. Joint Conf. Neural Networks*, Portland, OR, 2003, pp. 1301–1307.
- [66] C. Cai and S. Ferrari, "Bayesian network modeling of acoustic sensor measurements," in *Proc. IEEE Sensors Conf.*, Atlanta, GA, 2007, pp. 345–349.
- [67] S. Ferrari and A. Vaghi, "Demining sensor modeling and feature-level fusion by Bayesian networks," *IEEE Sensors*, vol. 6, pp. 471–483, 2006.
- [68] F. Jensen, *Bayesian Networks and Decision Graphs*. Berlin, Germany: Springer-Verlag, 2001.
- [69] Y. Yang and G. I. Webb, "A comparative study of discretization methods for naive-bayes classifiers," in *Proc. 2002 Pacific Rim Knowledge Acquisition Workshop (PKAW)*, Tokyo, Japan, 2002, pp. 159–173.
- [70] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning Bayesian networks: The combination of knowledge and statistical data," *Mach. Learn.*, vol. 20, pp. 197–243, 1995.
- [71] D. Heckerman, A Bayesian approach to learning causal networks, Tech. Rep. MSR-TR-95-04, May 1995, pp. 1–23.
- [72] D. Heckerman, "A tutorial on learning with Bayesian networks," in *Learning in Graphical Models*, M. Jordan, Ed., 1998, pp. 301–354.
- [73] R. Cowell, "Advanced inference in Bayesian networks," in *Learning in Graphical Models*, M. Jordan, Ed., 1998, pp. 27–50.
- [74] R. Urick, *Principles of Underwater Sound*, 3rd ed. New York: McGraw-Hill, 1983.
- [75] Harvard ocean prediction system (HOPS). Harvard University. [Online]. Available: <http://modelseas.mit.edu/HOPS/>
- [76] S. Meguerdichian, F. Koushanfar, G. Qu, and M. Potkonjak, "Exposure in wireless ad-hoc sensor networks," in *Proc. Mobile Comput. Networking*, 2001, pp. 139–150.
- [77] H. Qi, S. Iyengar, and K. Chakrabarty, "Distributed sensor networks—A review of recent research," *J. Franklin Institute*, vol. 338, no. 6, pp. 655–668, 2001.
- [78] Y. Zou and K. Chakrabarty, "A distributed coverage- and connectivity-centric technique for selecting active nodes in wireless sensor networks," *IEEE Trans. Comput.*, vol. 54, no. 8, pp. 978–991, Aug. 2005.
- [79] Q. Huang, "Solving an open sensor exposure problem using variational calculus," Washington Univ., Dept. Comput. Sci., St. Louis, MO, Tech. Rep. WUCS-03-1, Jan. 2003. [Online]. Available: <http://www.cs.wustl.edu/qingfeng/papers/ExposureTRShort.pdf>
- [80] S. Ross, *Introduction to Stochastic Dynamic Programming*. New York: Academic Press, 1983.



Kelli A. C. Baumgartner (M'07) received the B.S. degree from Embry-Riddle Aeronautical University, Prescott, AZ, and the M.S. and Ph.D. degrees from Duke University, Durham, NC.

She is a Flight Design and Controls Engineer in the System Development and Integration Division at Analex Corporation, Brook Park, OH, a subsidiary of QinetiQ North America. While at Duke University, her principal research interests included distributed sensor networks and intelligent systems using Bayesian networks.

Dr. Baumgartner is a member of the AIAA, the ASME, and a Ronald E. McNair Scholar.



Silvia Ferrari (SM'08) received the B.S. degree from Embry-Riddle Aeronautical University, Daytona Beach, FL, and the M.A. and Ph.D. degrees from Princeton University, Princeton, NJ.

She is an Assistant Professor of Mechanical Engineering and Materials Science at Duke University, Durham, NC, where she directs the Laboratory for Intelligent Systems and Controls (LISC). Her principal research interests include robust adaptive control of aircraft, learning and approximate dynamic programming, and optimal control of mobile sensor networks.

Prof. Ferrari is a member of ASME, SPIE, and AIAA. She is the recipient of the ONR Young Investigator Award (2004), the NSF CAREER Award (2005), and the Presidential Early Career Award for Scientists and Engineers (PECASE) Award (2006).



Thomas A. Wettergren (SM'06) received the B.S. degree in electrical engineering and the Ph.D. degree in applied mathematics, both from Rensselaer Polytechnic Institute, Troy, NY.

He joined the Naval Undersea Warfare Center, Newport, RI, in 1995, where he has served as a Research Scientist in the torpedo systems, sonar systems, and undersea combat systems departments. He currently is a Senior Scientist in the Center for Advanced System Technology, USW Combat Systems Department, where he leads a small group of researchers in the development of new mathematical techniques for undersea command and control systems. His personal research interests are in the mathematical modeling, analysis, optimization, and control of undersea sensing systems.

Dr. Wettergren is a member of the Institute for Operations Research and the Management Sciences (INFORMS).