Distributed Coverage Control for Mobile Anisotropic Sensor Networks

Bruno Hexsel
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
hexsel@gmail.com

Nilanjan Chakraborty
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
nilanjan@cs.cmu.edu

Katia Sycara
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
katia@cs.cmu.edu

Abstract—Distributed algorithms for (re)configuring sensors to cover a given area are important for autonomous multi-robot operations in application areas such as surveillance and environmental monitoring. Depending on the assumptions about the choice of the environment, the sensor models, the coverage metric, and the motion models of sensor nodes, there are different versions of the problem that have been formulated and studied. In this work, we consider the problem of (re)configuring systems equipped with anisotropic sensors (e.g., mobile robot with limited field of view cameras) that cover a polygonal region with polygonal obstacles for detecting interesting events. We assume that a given probability distribution of the events over this polygonal region is known. Our model has two key distinguishing features that are inherently present in covering problems with anisotropic sensors, but are not addressed adequately in the literature. First, we allow for the fact that the sensing performance may not be a monotonically decreasing function of distance. Second, motivated by scenarios where the sensing performance not only depends on the resolution of sensing, but also on the relative orientation between the sensing axis and the event, we assume that the probability of detection of an event depends on both sensing parameters and the angle of observation. We present a distributed gradient-ascent algorithm for (re)configuring the system of mobile sensors so that the joint probability of detection of events over the whole region is maximized. Simulation results illustrating the performance of our algorithms on different systems, namely, mobile camera networks, mobile acoustic sensor networks, and static pan-tilt-zoom camera networks are presented.

Index Terms—Distributed Multi-robot Coverage, Anisotropic Sensing, Sensor Placement.

I. INTRODUCTION

Multiple robot coverage problems have been studied in the context of a wide variety of applications such as surveillance, environmental monitoring, demining, floor cleaning, lawn mowing, harvesting, and industrial applications (e.g. drilling, milling, painting) [1]. Distributed algorithms for (re)configuring mobile sensors to cover a given area for surveillance and environmental monitoring applications have also received attention [2], [3]. In most of this literature (exceptions include [4], [5], [6], [7]), the sensors are either assumed to have infinite range or to be bounded range isotropic sensors (i.e., their performance does not depend on the direction in which they are sensing the object or event). Many popular sensors such as cameras or acoustic receivers with limited sensor footprint cannot be modeled as isotropic sensors (see Figure 1 for example anisotropic sensor footprints). In such scenarios, the probability of detection of an event may depend on the resolution of sensing as well as the angle at

which the event is being sensed. Thus, the performance of the sensors does not only depend on the relative distance between the sensor and the sensed point, but also on the relative orientation between them. Motivated by such application scenarios of mobile sensor networks, we study the problem of reconfiguring a system of mobile anisotropic sensors to optimize a given coverage metric that is dependent on the position of a point and the orientation at which the point is being sensed. We consider a system of active anisotropic sensors (e.g., mobile robots with limited sensor footprint cameras) that are required to cover a polygonal area with polygonal obstacles to maximize the joint probability of detection of interesting events.

For isotropic sensor models, the sensing region of coverage is usually assumed to be a disc of finite or infinite radius. For anisotropic sensors, they are assumed to be a segment of a disc [4], [5] or an ellipse [6] or an infinite cone [7]. In all these cases, the sensor position is always within the region of coverage and the sensor performance is assumed to be monotonically decreasing with the distance of the points being sensed. In our paper, the region of coverage of the sensors can be any bounded set. The sensor position need not be contained within the region of coverage. Thus, in our sensor model, the performance of the sensor does not necessarily deteriorate monotonically with distance from the sensor. Figure 1 shows example sensor footprints and probability of detection for a camera and an acoustic receiver (the sensor parameters used in generating the figures is given in Section VII). These sensor models are relevant in situations where we may not detect an event if it is too close to the sensor or if an event occurs at a point opposite to the heading of the sensor. For example, if an object is too close to a camera (or if we zoom in too much), such that only a fraction of the object is in the entire field of view, we may not be able to detect the object and hence our sensing performance will be poor. Furthermore, the acoustic receiver model (a cardioid [8]) indicates that the sensor performance for a point in the front of the sensor is much better than a point at the back.

With our anisotropic sensing model, the sensing performance in regions close to the sensor may not be good, hence using Voronoi region based distributed coverage algorithms (as used in [9], [10] and also used for anisotropic sensors in [4]) is not appropriate. Assuming that we have a probability distribution of occurrence of events over the entire region, we present a distributed algorithm that controls the position of the mobile agents and the sensor parameters to maximize
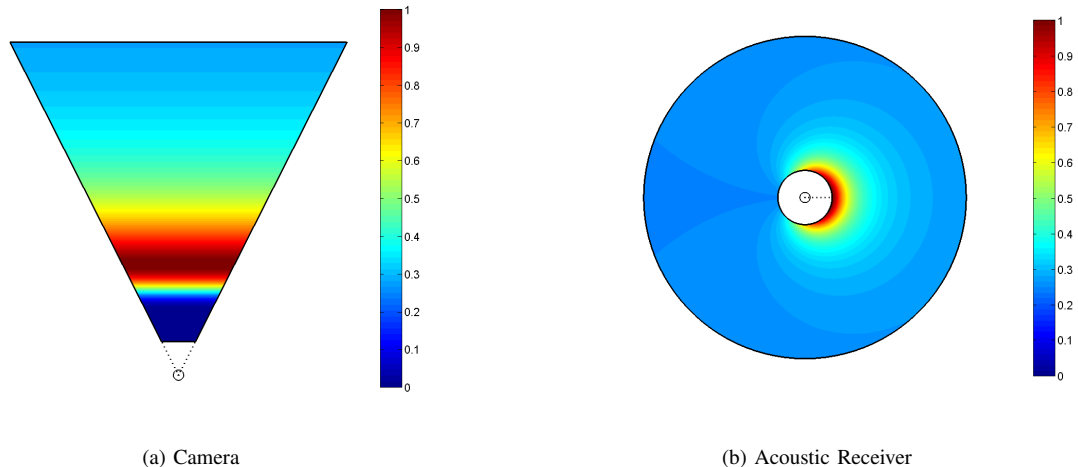
(a) Camera

(b) Acoustic Receiver

Fig. 1: Sensor footprint of a typical camera and acoustic receiver showing variation of probability of detection. Note that the detection probability is anisotropic and does not decrease monotonically with distance.

the joint detection performance of events by all the agents. Like Voronoi region based coverage algorithms, in general, our gradient-ascent algorithm converges to a local optimal solution of the objective function. The key challenge in developing our distributed control algorithm is that the presence of obstacles and bounded field of view of our sensors, makes our objective function discontinuous and non-differentiable. Consequently, by exploiting the geometry of the sensor footprints, we form the generalized gradients of the objective function at various discontinuities to compute the direction of motion for each robot. To illustrate our algorithm, we provide case studies of a group of mobile agents with cameras or directional acoustic receivers and static pan-tilt-zoom cameras. The objective is to maximize their performance while operating in an environment with obstacles.

Figure 2 shows a simple example of four agents with acoustic receivers moving to positions and orientations that maximize their joint probability of event detection over the rectangular environment. It is assumed that the density function of events occurring is uniform over the area. Note the directions of the mobile robots at the final configuration, which points away from the center of the rectangular environment. Since the robots have bad detection probability behind them, and each sensor has independent probability of detection, by overlapping the regions of coverage behind them the overall joint probability of detection is optimized.

This paper is organized as follows: In Section II we give a brief review of the recent literature on distributed coverage control. In Section III, we introduce our problem model and in Section IV we introduce our distributed controller. We illustrate our concepts using an example with mobile camera networks, mobile sound sensor networks and static pan-tilt-zoom cameras in Section V. In Section VI we present the overall distributed algorithm and in Section VII, we present simulation results demonstrating the performance of the algorithm on exemplar cases for each of the three examples above. Finally, in Section VIII, we present our conclusions and point

out future research directions. A preliminary version of this work appeared in [11].

II. RELATED WORK

Depending on the assumptions about the choice of the environment, the sensor models, the coverage metric, and the motion models of sensor nodes, there are different abstract versions of the coverage problem that has been formulated and studied for surveillance and monitoring applications. One of the most well known coverage problems with static sensors (cameras) is the *Art Gallery* problem for covering a given polygonal region with polygonal obstacles with omnidirectional infinite range sensors [12] of constant resolution. Here, the problem is to find the minimum number of sensors (and their positions) that are required to cover the region. The problem in its simplest version stated above is NP-hard and it is also shown to be APX-hard. Different variations of this basic problem have been studied with limited FoV sensors and with resolution metrics [13], [14], where the problem is still hard to solve. The literature on static sensor placement or camera placement to cover an area is quite substantive and we will restrict ourselves here to mobile sensor network coverage problems. Here, the usual assumption is that the number of mobile sensors is already known and we need to find their positions (or configurations) for maximizing a given coverage metric.

Coverage problems with a system of mobile sensors have been studied in recent years (e.g., [9], [15], [16]). In [15], the authors use a potential field approach for deploying a system of sensor nodes for maximizing area coverage. In [9], the authors use techniques from the facility location literature to give a distributed algorithm for mobile sensor placement. In their approach, the nodes at each step compute their (generalized) Voronoi regions and move towards the centroid of their Voronoi region until they converge. The underlying sensing model is isotropic and the sensing performance is assumed to decrease with distance. Although [9] assumed
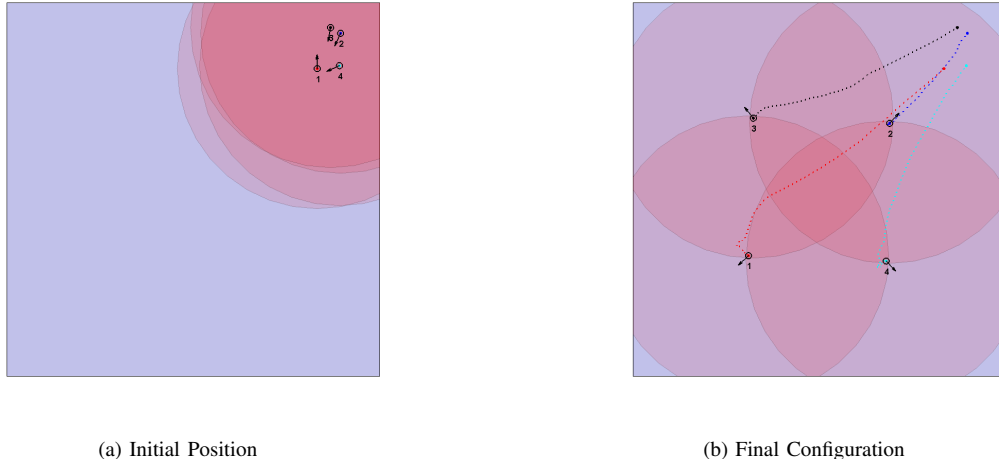
(a) Initial Position   (b) Final Configuration

Fig. 2: Result of applying our algorithm for configuring mobile robots equipped with directional acoustic receivers in a convex environment Q with uniform event density function. Figure 2a is the initial configuration and Figure 2b shows the final configuration. Note that the sensors each sensor face outward from the center of the convex region. They not only optimize the coverage over the region but also optimize their placement in the orientation space.

a convex polygonal environment, this approach has been extended subsequently to consider non-convex environments with obstacles and limited FoV sensors [2], [17], [18]. An approach that combines Voronoi coverage and the TangentBug path planning algorithm to address non-convex coverage is shown in [19], where the path planning algorithm is used to compute the motion around corners and obstacles. In [20], a distributed technique that aimed to minimize the information in each pixel was proposed for hovering robots with downward facing cameras. Optimizing the joint detection probability of events by a network of mobile robots was proposed in [18] and the work was extended to regions with polygonal obstacles in [21].

Game theoretic models for coverage control have been proposed in [22], [23], where each visual sensor individually attempts to optimize its own coverage. At the same time each sensor tries to optimize the energy and/or computational expenditures in processing the information. They assume a square 2D discretized lattice space where agents are deployed to detect events of interest. Each mobile agent caries visual sensors that are assumed to be pan-tilt-zoom cameras with limited field of view. The coverage optimization problem is posed as a constrained strategic game. The reward for observing a point in the lattice is unknown in the beginning of the game and whenever an agent observes the point the reward is shared among the other agents upon observing the point. The algorithm is provably convergent to the set of constrained Nash equilibrium and global optima for the given performance metric.

More recently, in [4], [5], [6], the Voronoi region based coverage algorithm has been extended to anisotropic sensor models. The sensor FoV is modeled as a segment of a disc in [5] and as an ellipse in [6]. For anisotropic sensors, the method in [9] does not give a distributed algorithm because the sensing regions of nodes that are not Voronoi neighbors

may overlap. To overcome this problem, the authors in [4], propose an alternative metric that approximates their coverage metric within a constant factor and it is shown that a Voronoi region based distributed algorithm can minimize the alternative metric. In [6], the authors propose to discretize the possible sensor orientations by assuming fixed, equally spaced sensor orientations and then show that they can modify the algorithm in [9] to obtain a distributed algorithm. However, a common assumption in all of this work is that the sensing performance decreases monotonically with distance. In this paper, we remove this assumption of monotonically decreasing performance functions. Motivated by applications with mobile camera networks, we assume that there may be a region close to the sensor that may not be within its FoV. We note that with this sensing model, it is no longer true that our coverage metric can be maximized by maximizing the coverage over the individual Voronoi regions of the robots (a key property of the metric proposed in [9]).

III. PROBLEM FORMULATION

In this work we study the problem of (re)configuring a system of agents (or sensor nodes) equipped with anisotropic sensors to cover a bounded environment with obstacles, so as to maximize a given coverage metric. The obstacles may affect both sensing performance and/or mobility of the sensor nodes. We will first define the basic notations that we use throughout this work and then present our modeling assumptions and optimization problem.

A. Notation

Let $\mathbb{R}$ and $\mathbb{R}_{\geq 0}$ be the set of real and non-negative real numbers, respectively. Let $\mathbb{R}^d$ and $\mathbb{S}^d$ denote the d-dimensional Euclidean and the d-dimensional space of orientations, respectively. Let $\mathrm{int}(S)$ and ∂S denote the interior and the boundary

of set S, respectively. Let n denote the outward unit normal vector at the boundary of a set.

Let $\|x - y\|_2$ denote the Euclidean distance between points $x, y \in \mathbb{R}^d$. Let the geodesic distance in $\mathbb{S}^1$ be

$$\text{dist}_g(x,y) = \min\{\text{dist}_C(x,y), \text{dist}_{CC}(x,y)\} \ x,y \in \mathbb{S}^1 \quad (1)$$

where $\text{dist}_C(x,y) = (x-y) \bmod 2\pi$ is the clockwise distance and $\text{dist}_{CC}(x,y) = (y-x) \bmod 2\pi$ is the counterclockwise distance. We also define the 1-dimensional orientation interval Θ as the set of possible orientations such that

$$\Theta = (-\pi, \pi]. \quad (2)$$

Let the closed line segment between the points $x, y \in \mathbb{R}^d$ be denoted by

$$[x,y] = \{\lambda x + (1-\lambda)y \mid \lambda \in [0,1]\}.$$

For a set $S \subset \mathbb{R}^d$, two points $x, y \in S$ are said to be visible if the closed segment $[x,y]$ is contained in S. The *visibility set* $\mathcal{V}(x,S)$ is the set of all points in S visible from x. Note that the visibility set is a geometric property, determined by the geometry of the environment and the geometry of the obstacles and their locations. It is independent of the sensor characteristics used to sense the environment. A set is said to be *convex* if the visibility set $\mathcal{V}(x,S)$ is S for all $x \in S$, otherwise the set is said to be *non-convex* [2].

B. Environment

The environment that we desire to cover is assumed to be a planar bounded environment $Q \in \mathbb{R}^2$ defined by a possibly non-convex polygon with non-self-intersecting edges. This environment may contain non-traversable *obstacles* that may also interfere with visibility. These obstacles are modeled as m non-self-intersecting polygons denoted by $H_j \subset Q, j = 1, \ldots, m$. The interior of these polygons are unobservable, so the feasible subspace of Q to be covered by sensors is

$$F = Q \backslash \left(\bigcup_{j=1}^{m} \text{int}(H_j) \right). \quad (3)$$

Moreover, the robot may not be able to traverse through the obstacles in Q and there may be other forbidden zones (or unsafe zones for the robot to traverse). We define the traversable space $T \subseteq F$ as the set of all possible locations that a mobile agent can occupy. In Figure 3, a non-convex environment with one obstacle and a connected traversable region is shown. Mobile agents may not be located outside the defined traversable region.

C. State Space of the Multi-agent System

The quality of information sensed by an agent, i, depends on its state x_i, which consists of the position and orientation of the agent as well as the sensor parameters. In a planar environment the agent's position is denoted by $c_i = [c_{ix} \ c_{iy}]^T$ and its orientation by θ_i. Agents are only allowed to move within the traversable space, T, described in III-B. The sensor specific parameter space is denoted by $\mathscr{P}$. For example, for a pan-tilt camera, the sensor parameter space is a subset of $\mathbb{S}^2$
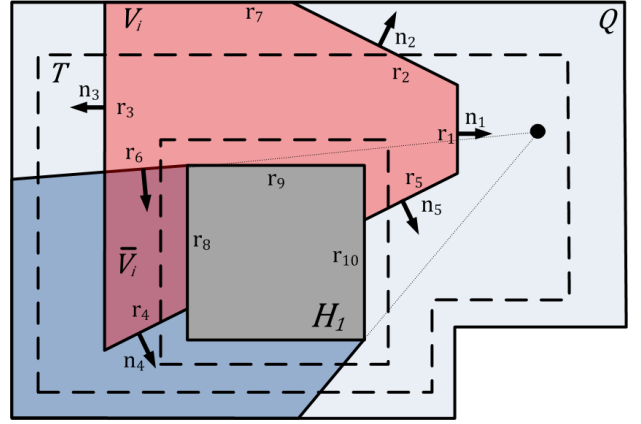

Fig. 3: Polygonal non-convex environment Q with obstacles H in gray. The area bounded by the two dashed lines indicate the traversable region T. Areas outside the traversable region in Q can be observed but not traversed. The different types of boundary line segments r_k are also shown. Boundaries r_1 through r_5 are coverage boundaries. Boundary r_6 is a visibility boundary. Boundaries r_7 through r_{10} are rigid boundaries. The vectors $n_1, \ldots, n_5$, are the outward normal unit vectors of the corresponding boundary segments.

consisting of all possible pan and tilt angles. Thus the state space of an agent is $\mathscr{X} = T \times \mathbb{S}^1 \times \mathscr{P}$. For the multiple agents scenario proposed in this paper, the joint state space, denoted by $\mathscr{X}^N$ is a Cartesian product of the individual state spaces. The state of the overall system is denoted by the concatenated vector $\mathbf{x} = (x_1, ..., x_N) \in \mathscr{X}^N$.

D. Anisotropic Density Function

In this paper we use an *anisotropic density function* defined by the map $\phi : Q \times \mathbb{S}^1 \to \mathbb{R}_{\geq 0}$. This density function is known to the agents and it represents the prior knowledge about the probability or a measure of information about events at a point $q \in Q$ when observed at an orientation $\alpha \in \mathbb{S}^1$. It is assumed that $\phi(q,\alpha) = 0 \forall q \notin F$ and

$$\int_\Theta \int_Q \phi(q,\alpha) dq d\alpha < \infty \quad (4)$$

This orientation is not to be confused with the angle between a point q and the agent, as it is a property of the density function independent of the state of the agent.

The intuition behind the orientation parameter α in the function $\phi(q,\alpha)$ is that it is a parameter that varies the sensing reward according to the absolute orientation in the map. A simple example of this is facial recognition, where the back of a person's head provides very little information about the individual. Conversely, the frontal view of the head maximizes the detection performance [24].

E. Anisotropic Sensing Model

The region covered by an agent i, denoted by $\mathscr{R}_i$ and called the *sensor footprint* of i, is dependent on the sensor characteristics of the agent and its current state. We place no restrictions on the geometry of $\mathscr{R}_i$, which can be, for

instance, a disk, polygon, non connected regions, wedge, etc. As stated earlier, the *visibility set* $\mathscr{V}(c_i, F)$ is the set of all points in F visible from the agent position c_i. Since $\mathscr{V}(c_i, F) \subset F$, the set of all visible points covered by the agent i is $V_i = \mathscr{R}_i \cap \mathscr{V}(c_i, F)$. The set of invisible points covered by the sensor is $\bar{V}_i = (\mathscr{R}_i \cap F) \backslash V_i$.

The probability of detecting an event is given by the function $p : \mathscr{X} \times Q \times \mathbb{S}^1 \to [0, 1]$. This probability not only depends on the specifics of the sensors on each agent, it may also depend on the nature of the task. It may or may not degrade upon the presence of obstacles or effects caused by the non-convexity of region Q. Therefore, the detection probability has to be considered for both the visible and invisible sets of points within the field of view as

$$p_i(x_i, q, \alpha) = \begin{cases} \hat{p}_i(x_i, q, \alpha) & \text{if } q \in V_i \\ \tilde{p}_i(x_i, q, \alpha) & \text{if } q \in \bar{V}_i \\ 0 & \text{otherwise} \end{cases}, \quad (5)$$

where $\hat{p}_i(x_i, q, \alpha)$ is the detection probability for a point q visible from agent i and $\tilde{p}_i(x_i, q, \alpha)$ is the detection probability for a point q invisible from agent i. For some sensors, example cameras, events invisible to the agent may not be sensed at all, leading to a detection probability $\tilde{p}_i(x_i, q, \alpha) = 0$. The set of points where the agent has a non-zero probability of event detection is the *field of view* of agent i.

It is assumed that every agent can sense events independently. Therefore, the probability of detection of an event at point q and at orientation α is given by the joint probability of detection for each of the sensors given by

$$P(\mathbf{x}, q, \alpha) = 1 - \prod_{j=1}^{N} [1 - p_j(x_j, q, \alpha)]. \quad (6)$$

We want to find a joint state of the agents that maximize the joint probability of detection over the entire environment Q. The optimization problem is then

$$\max_{\mathbf{x}} \int_{\Theta} \int_Q P(\mathbf{x}, q, \alpha) \phi(q, \alpha) dq d\alpha$$
$$\text{subject to } \mathbf{x} \in \mathscr{X}^N \quad (7)$$

and the objective function function from (7) is

$$\mathscr{H}(\mathbf{x}) = \int_{\Theta} \int_Q P(\mathbf{x}, q, \alpha) \phi(q, \alpha) dq d\alpha. \quad (8)$$

Note that when p_i and ϕ are independent of α and ϕ is uniform, the problem is an area coverage problem.

IV. DISTRIBUTED CONTROLLER

The goal of this paper is to present a distributed controller that maximizes the joint detection probability over the environment. We use a (generalized) gradient based controller to coordinate the multiple robotic nodes according to the optimization problem described in (7).

We compute the gradient for both the visible set of points V_i and the invisible points $\bar{V}_i$. Also, all the discontinuities of the function p_i should be considered. Let the set of discontinuity intervals of points (or line segments) be denoted by $\mathrm{Dscn}(p_i)$. These include boundaries of the covered region,

boundaries between the visible and invisible set, and possible discontinuities in the sensing model, as seen in Figure 3. Applying the rules of differentiation under the integral sign to equation (8), we have

$$\frac{\partial \mathscr{H}(\mathbf{x})}{\partial x_i} = \int_{\Theta} \int_{V_i \cup \bar{V}_i} \frac{\partial P(\mathbf{x}, q, \alpha)}{\partial x_i} \phi(q, \alpha) dq d\alpha +$$
$$\sum_{r_k \in \mathrm{Dscn}(p_i)} \int_{\Theta} \int_{r_k} \Phi_k(\mathbf{x}, q, \alpha) \frac{\partial q_{r_k}}{\partial x_i}^T n_k \phi(q, \alpha) dq d\alpha \quad (9)$$

where r_k is a line segment where the performance function is discontinuous, n_k is the outward normal of r_k, q_{r_k} is, with abuse of notation, a point in r_k, and $\Phi_k(\mathbf{x}, q, \alpha)$ is defined as

$$\Phi_k(\mathbf{x}, q, \alpha) = P_k^-(\mathbf{x}, q, \alpha) - P_k^+(\mathbf{x}, q, \alpha)$$

where we define P_k^- and P_k^+ as

$$P_k^-(\mathbf{x}, q, \alpha) = \lim_{\varepsilon \to 0^+} P(\mathbf{x}, q - \varepsilon n_k, \alpha) \quad (10)$$
$$P_k^+(\mathbf{x}, q, \alpha) = \lim_{\varepsilon \to 0^+} P(\mathbf{x}, q + \varepsilon n_k, \alpha). \quad (11)$$

In Equation (9), the generalized gradient $\frac{\partial \mathscr{H}(\mathbf{x})}{\partial x_i}$ is a $m_i \times 1$ vector, $\frac{\partial P(\mathbf{x}, q, \alpha)}{\partial x_i}$ is a $m_i \times 1$ vector, $\frac{\partial q_{r_k}}{\partial x_i}$ is a $2 \times m_i$ matrix, and n_k is a 2×1 vector, where m_i is the dimension of the state space of robot i.

The control update from time k to $k+1$ is

$$x_i^{k+1} = x_i^k + K \frac{\partial \mathscr{H}(\mathbf{x})}{\partial x_i} \quad (12)$$

where K is a diagonal matrix with the individual gains for each of the parameters in x_i in its diagonal. The choice of gains for convergence are based on standard rules [21], [5], for which we refer the reader to [25]. Furthermore, if the position of the robot after the update is outside of the traversable subspace T, the robot will move instead to the closest point in T from the desired position.

A. Network Requirement

From (5) we see that the detection probability is always equal to zero outside the field of view $(V_i \cup \bar{V}_i)$. At any point in the environment, robots whose field of view intersect will contribute to the joint probability computation in (6) (and thereby the sensing performance). Thus, to compute (12), a robot needs to know the function $\phi(q, \alpha)$ and information about the state of its neighboring robots, $\mathscr{B}_i$, defined as

$$\mathscr{B}_i = \left\{ j \mid F \cap (V_i \cup \bar{V}_i) \cap (V_j \cup \bar{V}_j) \neq \emptyset, i \neq j \right\}.$$

This can be achieved if the communication range of the robots is at least twice that of their sensing range, which is reasonable in many applications.

Remark 1. To implement this controller, it is necessary to discretize the coverage region to compute the integrals over the possible orientations $\alpha \in \Theta$. Furthermore, the detection performance p_j of the neighbors $j \in \mathscr{B}_i$ over the node's coverage region is necessary to obtain (9).

B. Computation of Gradient Terms

Here we explain how to obtain the terms in (9). The first term described here is the internal gradient of the joint probability computed over regions V_i and $\bar{V}_i$. This gradient is assumed to be globally Lipschitz and continuously differentiable over the set $(V_i \cup \bar{V}_i) \setminus \text{Dscn}(p_i)$. The second term of (9) deals with the discontinuities caused by the environment and/or the sensing model. As stated earlier and depicted in Figure 3, there are three types of boundaries or line segments where $\mathscr{H}(\mathbf{x})$ is discontinuous. They are the rigid boundaries (e.g., line segment r_9 in Figure 3), the coverage boundaries (e.g., r_1, r_2, in Figure 3), and the visibility boundary (e.g., r_6 in Figure 3). In the interest of conciseness, we will omit the arguments x_i, q, and α of the function p.

1) Internal Gradient: The first term in equation (9) is the internal gradient and should be computed over the covered regions V_i and $\bar{V}_i$. Since every node is assumed to be independent of the other, from (6) we obtain

$$\frac{\partial P(\mathbf{x}, q, \alpha)}{\partial x_i} = \frac{\partial p_i}{\partial x_i} \prod_{j \in \mathscr{B}_i} [1 - p_j].$$

2) Rigid Boundary: The rigid boundary consists of points lying on the boundary of obstacles and the boundary of region Q. These points are in the set $\mathscr{R}_i \cap \text{bd}(F)$, where $\text{bd}(F)$ denotes the boundary of the set F. Since they do not vary as a function of the parameters x_i, they do not affect the gradient calculation.

$$\frac{\partial q_{r_k}}{\partial x_i} = 0 \ \forall \ q_{r_k} \in \text{bd}(F)$$

3) Coverage Boundary: For points on the boundary of the sensor coverage set $\partial \mathscr{R}_i \cap \text{int}(F)$ (where $\text{int}(F)$ denotes the interior of the set F), there is non-zero detection performance immediately inside the filed of view, $V_i \cup \bar{V}_i$, and zero detection performance immediately outside this region. Thus, from (10) we have

$$P_k^-(\mathbf{x}, q, \alpha) = 1 - (1 - p_i) \prod_{j \in \mathscr{B}_i} [1 - p_j]$$

$$P_k^+(\mathbf{x}, q, \alpha) = 1 - \prod_{j \in \mathscr{B}_i} [1 - p_j]. \tag{13}$$

The discontinuity probability function then becomes

$$\Phi_k(\mathbf{x}, q, \alpha) = p_i \prod_{j \in \mathscr{B}_i} [1 - p_j].$$

The values of $\partial q_{r_k} / \partial x_i$ on the coverage boundary vary according to x_i and are sensor parameter dependent (see Section V for illustration of this computation for different realistic sensor models).

4) Visibility Boundary: The visible and invisible set boundary is the interval where there is a transition between the visible coverage region V_i and the invisible coverage region $\bar{V}_i$. These points are defined by the set $\partial V_i \cap (\text{int}(\mathscr{R}_i) \setminus \partial F)$.

$$P_k^-(\mathbf{x}, q, \alpha) = 1 - (1 - p_i) \prod_{j \in \mathscr{B}_i} [1 - p_j]$$

$$P_k^+(\mathbf{x}, q, \alpha) = 1 - (1 - \tilde{p}_i) \prod_{j \in \mathscr{B}_i} [1 - p_j]. \tag{14}$$

The discontinuity probability function then becomes

$$\Phi_k(\mathbf{x}, q, \alpha) = (\hat{p}_i - \tilde{p}_i) \prod_{j \in \mathscr{B}_i} [1 - p_j].$$

Note that the visibility boundary is generated by a vertex either of the region Q or of an obstacle H_j. In this case also, the computation of $\partial q_{r_k} / \partial x_i$ are dependent on the sensor parameters and we discuss it further in Section V. For an excellent geometric description on how the visibility boundary is generated in a region with polygonal obstacles we refer the reader to [21].

V. CASE STUDIES

In this section we illustrate the computation of the gradient terms, i.e., the term $\partial q_{r_k} / \partial x_i$ in the distributed algorithm for different anisotropic sensors. We will present three different examples, a mobile robot with a limited footprint camera, a mobile robot with a directional acoustic receiver and static cameras with pan, tilt and zoom parameters. We will exploit the geometry of the sensor footprint, which is a function of the sensor parameters to compute the gradient terms. Since the algorithm is intended to be used in a planar 2D environment, some task specific assumptions must be made in the problem formulation for each example presented.

A. Mobile Camera Coverage

Here we formulate an example application of the distributed coverage algorithm in a scenario of a mobile robot with integrator dynamics equipped with a limited field of view camera. The probability of detection is a function of the resolution $\mathscr{N}_i$ at a given depth Z_i from the camera point of view. The boundaries are delimited by a minimum and maximum depth for a possible detection, given by Z_{min} and Z_{max}, respectively. Moreover, a frontal observation of targets is preferred in this scenario. The environment and the event density with the associated orientation $\phi(q, \alpha)$ will be provided beforehand.

The state space for sensor i is $x_i = [c_{xi} \ c_{y_i} \ \theta_i]^T$, where c_x and c_y are the center position of the agent in the x and y axis, respectively. The orientation of the agent and the region of coverage are given by the parameter θ. For a given image sensor, let l_H and l_V be the camera sensor height and length, respectively. The number of pixels along the horizontal and vertical axes are given by N_H and N_V, respectively.

It is useful to analyze the environment in the camera coordinate frame. Let the rotation matrix Rot_{θ_i} be defined as

$$\text{Rot}_{\theta_i} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{bmatrix}, \tag{15}$$

points can be converted into a camera centered coordinate frame by applying the transformation

$$\begin{bmatrix} Z_i \\ Y_i \end{bmatrix} = \text{Rot}_{\theta_i}^{-1}(q - c_i) = \text{Rot}_{\theta_i}^T \begin{bmatrix} q_x - c_{xi} \\ q_y - c_{y_i} \end{bmatrix} \tag{16}$$

where Z_i and Y_i are the depth and horizontal displacement with respect to the camera center in camera coordinates.

1) Sensor Footprint: The boundaries of coverage are determined by the minimum and maximum depth Z_{min} and Z_{max} in camera coordinates and the limitations given by the angle of the field of view of the camera. For a given focal distance f, the horizontal angle γ_H and the vertical angle γ_V are given by

$$\gamma_H = 2\arctan\left(\frac{l_H}{2f}\right) \qquad \gamma_V = 2\arctan\left(\frac{l_V}{2f}\right). \qquad (17)$$

The absolute value of the angle between a point q in the field of view and the camera center c_i must be less or equal to γ_H. Therefore, the coordinate Y_i is restricted by the field of view as in

$$\left|\arctan\left(\frac{Y_i}{Z_i}\right)\right| \leq \gamma_H \Rightarrow |Y_i| \leq \frac{l_H}{2f}Z_i. \qquad (18)$$

The field of view of the camera, $\mathscr{R}_i$, is

$$\mathscr{R}_i = \left\{q \mid Z_{min} \leq Z_i \leq Z_{max} \bigwedge |Y_i| < \frac{l_H}{2f}Z_i\right\}. \qquad (19)$$

Figure 4 gives the camera parameters and variables that define the field of view.

2) Sensor Model: The resolution of the camera at a depth Z_i is given by the the total number of pixels divided by the observed area, $A(Z_i)$. This observed area is given by

$$A(Z_i) = \left(2Z_i\tan\left(\frac{\gamma_H}{2}\right)\right)\left(2Z_i\tan\left(\frac{\gamma_V}{2}\right)\right) = Z_i^2\frac{l_H l_V}{f^2} \qquad (20)$$

by using γ_H and γ_V from (17). The resolution is given by

$$\mathscr{N}_i = \frac{N_H N_V}{A(Z_i)} = \frac{N_H N_V}{l_H l_V}\frac{f^2}{Z_i^2} = \frac{\mathscr{K}_{cam}}{Z_i^2}, \qquad (21)$$

where $\mathscr{K}_{cam}$ is a constant dependent on the physical characteristics of the camera sensor.

We model the detection probability as a function of the resolution at a given depth Z_i from the camera point of view as an exponential function. This function was chosen because of its smoothness properties and relevance in a probabilistic framework (however other choices relevant to the application at hand may be made). The probability of detection according to the pixel area is given by

$$p_{\mathscr{N}_i} = \exp\left(-\frac{(\mathscr{N}_i - \mathscr{N}_\mu)^2}{2\mathscr{N}_\sigma^2}\right), \qquad (22)$$

where $\mathscr{N}_\mu$ is the optimal pixel area for a detection and $\mathscr{N}_\sigma$ is a constant to determine how spread out the detection distribution $p_{\mathscr{N}_i}$ is with respect to the depth Z_i. Since a frontal direct observation is desired, the probability of detection varies as a function of geodesic distance between the current direction of observation θ and the orientation α. It is also modeled as an exponential in

$$p_\alpha = \exp\left(-\frac{\text{dist}_g(\alpha, \theta_i)^2}{2\sigma_\alpha^2}\right), \qquad (23)$$

where σ_α is a constant to determine how spread out the orientation detection distribution p_α is with respect to the geodesic distance between the observation direction and the orientation α.

Since no detection is possible behind obstacles in a camera scenario, the overall detection probability function is

$$p_i = \begin{cases} p_0 p_{\mathscr{N}_i} p_\alpha & \text{if } q \in V_i \\ 0 & \text{otherwise} \end{cases} \qquad (24)$$

where p_0 is the maximum detection probability of the sensor.

3) Controller: In order to obtain the gradient ascent controller, the partial derivative of the detection probability should be obtained for region V_i. The boundary gradients for the visibility and coverage boundary should also be obtained. We describe here derivation of these expressions.

a) Gradient at interior points: The partial derivative of p_i with respect to x_i in the region V_i can be obtained by applying the chain rule to (24) as in

$$\frac{\partial p_i}{\partial x_i} = \left[-\frac{(\mathscr{N}_i - \mathscr{N}_\mu)}{\sigma_{\mathscr{N}}^2}\frac{\partial \mathscr{N}_i}{\partial x_i} - \frac{\text{dist}_g(\alpha, \theta_i)}{\sigma_\alpha^2}\frac{\partial \theta_i}{\partial x_i}\right]p_i. \qquad (25)$$

where from (21) and (16), we have

$$\frac{\partial \mathscr{N}_i}{\partial x_i} = -\frac{2\mathscr{K}_{cam}}{Z_i^3}\frac{\partial Z_i}{\partial x_i} = -\frac{\mathscr{K}_{cam}}{Z_i^3}\begin{bmatrix} -\cos(\theta_i) \\ -\sin(\theta_i) \\ Y_i \end{bmatrix}. \qquad (26)$$

and $\frac{\partial \theta_i}{\partial x_i} = [0\ 0\ 1]^T$.

Thus, for the first term of (9), the gradient for the region V_i is

$$\mathscr{G}_{V_i} = \int_\Theta \int_{V_i} \frac{\partial p_i}{\partial x_i} \prod_{j\in\mathscr{B}_i}[1 - p_j]\phi(q, \alpha)dqd\alpha$$

b) Gradient at coverage boundary: To obtain the variation of a point q (or q_{r_k}) on the coverage boundary with respect to c_{x_i}, c_{y_i}, θ_i, we first notice that the geometry of the boundary is independent of any of the parameters in x_i. Thus, from (16) we have for the position parameter c_i

$$\frac{\partial}{\partial c_i}\begin{bmatrix} Z_i \\ Y_i \end{bmatrix} = \frac{\partial}{\partial c_i}\left(\text{Rot}_{\theta_i}^T(q - c_i)\right) = 0$$

Since Rot_{θ_i} is an orthonormal matrix, we have

$$\frac{\partial q}{\partial c_i} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad (27)$$

For the orientation parameter θ_i, we have

$$\frac{\partial}{\partial \theta_i}\begin{bmatrix} Z_i \\ Y_i \end{bmatrix} = \frac{\partial}{\partial \theta_i}\left(\text{Rot}_{\theta_i}^T(q - c_i)\right) = 0. \qquad (28)$$

Isolating $\partial q/\partial \theta$ in (28) we obtain

$$\frac{\partial q}{\partial \theta_i} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}(q - c_i) \qquad (29)$$

Combining (27) and (29), the boundary varies with respect to the parameters in x_i as

$$\frac{\partial q_{r_k}}{\partial x_i} = \begin{bmatrix} 1 & 0 & -q_y + c_{y_i} \\ 0 & 1 & q_x - c_{x_i} \end{bmatrix}.$$

Thus, for points on the coverage boundary we have

$$\mathscr{G}_{\text{bd}(V_i)} = \sum_{r_k\in\text{Dscn}_{\text{bd}(V_i)}(p_i)} \int_\Theta \int_{r_k} p_i \prod_{j\in\mathscr{B}_i}[1 - p_j]\frac{\partial q_{r_k}}{\partial x_i}^T n_k\phi dqd\alpha$$

where $\text{Dscn}_{\text{bd}(V_i)}(p_i)$ is the union of all the line segments forming the coverage boundary.
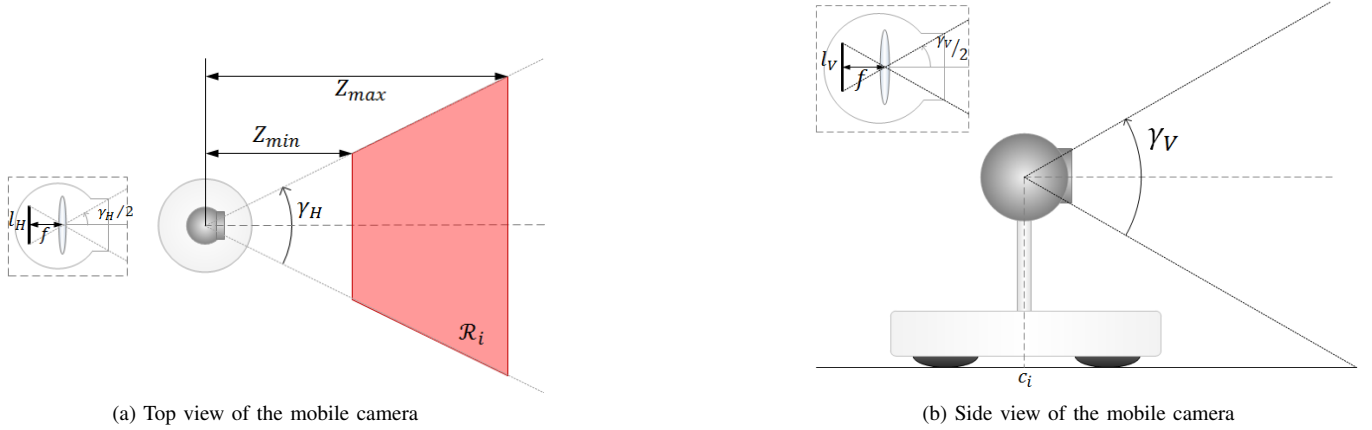
(a) Top view of the mobile camera
(b) Side view of the mobile camera

Fig. 4: Variables for the determination of the coverage boundary $\mathscr{R}_i$ and the pixel resolution $\mathscr{N}_i$.

c) Gradient at visibility boundary: As discussed before, obstacles and the environment introduce visibility boundaries. In this example, the visibility boundary varies only with respect to the position of the robot c_i. This boundary is induced by a vertex v_j that can either belong to an obstacle H_j or to the boundary Q. The proof of this gradient can be found in [21].

$$\frac{\partial q_{r_k}}{\partial x_i} = \begin{bmatrix} -\frac{\|q-v_j\|_2}{\|c_i-v_j\|_2} & 0 & 0 \\ 0 & -\frac{\|q-v_j\|_2}{\|c_i-v_j\|_2} & 0 \end{bmatrix}$$

For points on a visibility boundary, we have

$$\mathscr{G}_{\text{vis}} = \sum_{r_k \in \text{Dscn}_{\text{vis}}(p_i)} \int_\Theta \int_{r_k} p_i \prod_{j \in \mathscr{B}_i} [1-p_j] \frac{\partial q_{r_k}}{\partial x_i}^T n_k \phi dq d\alpha$$

where $\text{Dscn}_{\text{vis}}(p_i)$ is the union of all the line segments forming the visibility boundary.

d) Control law: After calculating all the terms as in (9), we obtain the gradient

$$\frac{\partial \mathscr{H}(\mathbf{x})}{\partial x_i} = \mathscr{G}_{V_i} + \mathscr{G}_{\text{bd}(V_i)} + \mathscr{G}_{\text{vis}}, \tag{30}$$

and we can update the state using the control law in (9).

B. Directional Acoustic Receiver

Here we consider an example of distributed coverage using mobile robots with integrator dynamics equipped with acoustic receivers. The probability of detection is a function of the angle and the distance of a point q from the sound sensor (see Figure 1b for a typical detection distribution). We assume that the response is in the form of a cardioid (as in many of the commercially available microphone sensors [8]). This implies maximum response directly in front of the sensor and minimum in the back of the sensor. The coverage boundaries are given by a minimum and maximum distance for a possible detection, namely, D_{min} and D_{max}, respectively. Moreover, a frontal observation of targets is preferred in this scenario.

The state space for sensor i is $x_i = [c_{xi}\ c_{yi}\ \theta_i]^T$, where c_x and c_y are the center position of the agent in the x and y axis, respectively. The orientation of the agent and its sound sensor is given by the parameter θ.

1) Sensor Footprint: The boundaries of coverage for this sensor are delimited by the minimum and maximum distance from the sensor, given by D_{min} and D_{max}, respectively. The region of coverage is therefore a hollow disc defined by

$$\mathscr{R}_i = \{q \mid D_{min} \leq \|q-c_i\|_2 \leq D_{max}\}. \tag{31}$$

2) Sensor Model: The detection probability is modeled as a function of the perceived intensity and the orientation. For a given sound sensor, let b_{mic} be a constant whose value indicates the best possible reception by the sensor. We also define the angle between a point q and the sensor as

$$\varphi(x_i, q) = \theta_i - \text{atan2}(q_y - c_y, q_x - c_x). \tag{32}$$

The intensity, $I(x_i, q)$, of the received sound is a function of the angle and the distance from the sensor to a point. The intensity of sound waves decay quadratically with distance. We simplify the cardioid model of sound perception as

$$I(x_i, q) = \frac{b_{mic}}{2} \frac{[1 + \cos(\varphi(x_i, q))]}{\|q-c_i\|_2^2}, \tag{33}$$

where b_{mic} is a constant related to the physical characteristics of the microphone. Once more we model the detection probability as function of the sound intensity as

$$p_I = \exp\left(-\frac{(I(x_i, q) - I_\mu)^2}{2I_\sigma^2}\right), \tag{34}$$

where I_μ is the optimal sound intensity for a detection and I_σ is a constant that determines the spread of the distribution p_I.

A frontal direct observation is desired. We have a probability of detection that varies as a function of geodesic distance between the current direction of observation θ_i and the orientation α. It is also modeled as an exponential in

$$p_\alpha = \exp\left(-\frac{\text{dist}_g(\alpha, \theta_i)^2}{2\sigma_\alpha^2}\right), \tag{35}$$

where σ_α is a constant to determine how spread out the orientation detection distribution p_α is in respect to the geodesic distance between the observation direction and the orientation α.

In contrast to the camera case, sound can be perceived even if the source is not directly in the line of sight of the sensor. To simplify inherently complex problem of estimating sound intensity decay around obstacles, we assume that the probability of detection is different for visible and invisible points. The overall detection performance is given by

$$p_i = \begin{cases} p_0 p_I p_\alpha & \text{if } q \in V_i \\ \tilde{p}_0 p_I p_\alpha & \text{if } q \in \bar{V}_i \\ 0 & \text{otherwise} \end{cases} \tag{36}$$

where p_0 is the maximum detection probability by the sensor if a point is visible and $\tilde{p}_0$ is the maximum detection probability if a point cannot be seen directly.

3) Controller: The derivation of the gradients for the mobile anisotropic sound sensor is similar to the mobile camera case. What differs for this sensor is the capability of detection of events even in the invisible region. The derivation of the gradient terms will be shown for this case.

a) Gradient at interior points: The partial derivative of the detection probability p_i in the region V_i is

$$\frac{\partial p_i}{\partial x_i} = \left[-\frac{(I(x_i, q) - I_\mu)}{I_\sigma^2} \frac{\partial I(x_i, q)}{\partial x_i} - \frac{\text{dist}_g(\alpha, \theta)}{\sigma_\alpha^2} \frac{\partial \theta_i}{\partial x_i} \right] p_i. \tag{37}$$

where $\frac{\partial \theta_i}{\partial x_i} = [0\ 0\ 1]^T$ and

$$\frac{\partial I(x_i, q)}{\partial x_i} = -\frac{b_{mic}}{2} \left[\frac{\sin(\varphi(x_i, q))}{\|q - c_i\|_2^2} \frac{\partial \varphi(x_i, q)}{\partial x_i} \right. \\ \left. + \frac{2 + 2\cos(\varphi(x_i, q))}{\|q - c_i\|_2^3} \frac{\partial \|q - c_i\|_2}{\partial x_i} \right], \tag{38}$$

with

$$\frac{\partial \varphi(x_i, q)}{\partial x_i} = \left[-\frac{q_y - c_{y_i}}{\|q - c_i\|_2^2} \quad \frac{q_x - c_{x_i}}{\|q - c_i\|_2^2} \quad 1 \right], \tag{39}$$

$$\frac{\partial \|q - c_i\|_2}{\partial x_i} = \left[\frac{q_x - c_{x_i}}{\|q - c_i\|_2} \quad \frac{q_y - c_{y_i}}{\|q - c_i\|_2} \quad 0 \right]. \tag{40}$$

The gradient term for points in the region V_i is

$$\mathcal{G}_{V_i} = \int_\Theta \int_{V_i} \frac{\partial p_i}{\partial x_i} \prod_{j \in \mathcal{B}_i} [1 - p_j] \phi(q, \alpha) dq d\alpha.$$

For the region invisible region $\bar{V}_i$, the gradient term

$$\mathcal{G}_{\bar{V}_i} = \int_\Theta \int_{\bar{V}_i} \frac{\partial \tilde{p}_i}{\partial x_i} \prod_{j \in \mathcal{B}_i} [1 - p_j] \phi(q, \alpha) dq d\alpha,$$

where

$$\frac{\partial \tilde{p}_i}{\partial x_i} = \left[-\frac{(I(x_i, q) - I_\mu)}{I_\sigma^2} \frac{\partial I(x_i, q)}{\partial x_i} - \frac{\text{dist}_g(\alpha, \theta)}{\sigma_\alpha^2} \frac{\partial \theta_i}{\partial x_i} \right] \tilde{p}_i.$$

b) Gradient at coverage boundary: As in the camera case, the geometry of the coverage boundary does not vary with respect to the parameter x_i. Therefore, the same equations from V-A3b can be used. Points in the boundary vary with respect to x_i as

$$\frac{\partial q_{r_k}}{\partial x_i} = \begin{bmatrix} 1 & 0 & -q_y + c_{y_i} \\ 0 & 1 & q_x - c_{x_i} \end{bmatrix}.$$

The gradient term for points on the coverage boundary is

$$\mathcal{G}_{\text{bd}(V_i \cup \bar{V}_i)} = \sum_{r_k \in \text{Dscn}_{\text{bd}(V_i \cup \bar{V}_i)}(p_i)} \int_\Theta \int_{r_k} p_i \prod_{j \in \mathcal{B}_i} [1 - p_j] \frac{\partial q_{r_k}}{\partial x_i}^T n_k \phi dq d\alpha.$$

where $\text{Dscn}_{\text{bd}(V_i \cup \bar{V}_i)}(p_i)$ is the union of all line segments forming the coverage boundary.

c) Gradient at visibility boundary: : The visibility varies for this sensor similarly to the camera case as discussed in V-A3c. The visibility boundary is given by

$$\frac{\partial q_{r_k}}{\partial x_i} = \begin{bmatrix} -\frac{\|q - v_j\|_2}{\|c_i - v_j\|_2} & 0 & 0 \\ 0 & -\frac{\|q - v_j\|_2}{\|c_i - v_j\|_2} & 0 \end{bmatrix}$$

As discussed previously, it is a characteristic of this sensor to have detection performance in the invisible coverage region. Therefore, the gradient term for the visibility becomes

$$\mathcal{G}_{\text{vis}} = \sum_{r_k \in \text{Dscn}_{\text{vis}}(p_i)} \int_\Theta \int_{r_k} (\hat{p}_i - \tilde{p}_i) \prod_{j \in \mathcal{B}_i} [1 - p_j] \frac{\partial q_{r_k}}{\partial x_i}^T n_k \phi dq d\alpha.$$

where $\hat{p}_i$ is the detection probability for a visible set and $\tilde{p}_i$ is the probability of detection for a point in the invisible set.

d) Control law: After calculating all the terms as in (9), we obtain the gradient

$$\frac{\partial \mathcal{H}(\mathbf{x})}{\partial x_i} = \mathcal{G}_{V_i} + \mathcal{G}_{\bar{V}_i} + \mathcal{G}_{\text{bd}(V_i \cup \bar{V}_i)} + \mathcal{G}_{\text{vis}}, \tag{41}$$

and we can update the state by using the control law in (9).

C. Pan-Tilt-Zoom Coverage

In this section we consider an example application of distributed planar coverage with static pan-tilt-zoom cameras. The cameras are assumed to be placed at a fixed position $c = [c_{xi}\ c_{yi}]^T$ and height h_i. We will assume that the coverage area is the projection of the image plane onto the ground (see Figure 5). The probability of detection is a function of the resolution $\mathcal{N}_i$ at a given depth Z_i from the camera point of view and the camera focal distance and tilt parameter. A frontal observation of targets is preferred in this scenario. The environment and the event density with the associated orientation $\phi(q, \alpha)$ will be provided beforehand.

The state space for sensor i is $x_i = [\theta_i\ \beta_i\ f_i]^T$, where θ_i is the pan angle, β_i is the tilt angle and f is the focal length. Let l_H and l_V be the camera image sensor height and length, respectively. The number of pixels along the horizontal and vertical axes are given by N_H and N_V, respectively. Let $q = [q_x\ q_y\ 0]^T$ be the position of a point in the world frame and $[X_i\ Y_i\ Z_i]^T$ be its position in the camera coordinate frame. Then,

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = R_{tilt}^T(\beta_i) R_{pan}^T(\theta_i) \begin{bmatrix} q_x - c_{xi} \\ q_y - c_{yi} \\ 0 - h_i \end{bmatrix}, \tag{42}$$

where $[c_{xi}\ c_{yi}\ h_i]^T$ is the position of the camera in the world frame. The pan rotation is about the z axis and tilt rotation is

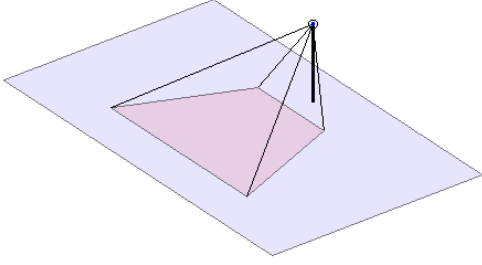Fig. 5: Pan-Tilt-Zoom camera projecting its image plane onto the region Q. Camera is at a height h_i and the coverage region $\mathscr{R}_i$ is the red area projected onto the planar surface.

Fig. 6: Example of probability of detection of events with static pan-tilt-zoom camera.

about the y axis in the world coordinate frame. The rotation matrices are

$$R_{tilt}(\beta_i) = \begin{bmatrix} \cos(\beta_i) & 0 & \sin(\beta_i) \\ 0 & 1 & 0 \\ -\sin(\beta_i) & 0 & \cos(\beta_i) \end{bmatrix} \quad (43)$$

$$R_{pan}(\theta_i) = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

By substituting 43 in 42 we can obtain $P_i = [X_i \, Y_i]^T$ and Z_i such that

$$P_i = T(\beta_i, \theta_i)(q - c_i) + h_i V(\beta_i). \quad (44)$$

$$Z_i = t(\beta_i, \theta_i)(q - c_i) + h_i v(\beta_i). \quad (45)$$

where $v(\beta_i) = -\cos(\beta_i)$, $V(\beta_i) = [\sin(\beta_i) \, 0]^T$, and

$$T(\beta_i, \theta_i) = \begin{bmatrix} \cos(\beta_i)\cos(\theta_i) & \cos(\beta_i)\sin(\theta_i) \\ -\sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \quad (46)$$

$$t(\beta_i, \theta_i) = [\sin(\beta_i)\cos(\theta_i) \quad \sin(\beta_i)\sin(\theta_i),]. \quad (47)$$

The coordinates of a point on the plane is given in the image coordinates according to the perspective transformation, namely,

$$\begin{bmatrix} x_{image_i} \\ y_{image_i} \end{bmatrix} = f_i \frac{P_i}{Z_i}. \quad (48)$$

1) Sensor Footprint: The planar coverage set is a projection of the ground plane on the image plane. All the points q in the plane are projected onto the image plane through a perspective transform. Thus, the sensing set is given by

$$\mathscr{R}_i = \left\{ q \mid \left| f_i \frac{P_i}{Z_i} \right| \leq \begin{bmatrix} l_H \\ l_V \end{bmatrix} \right\}. \quad (49)$$

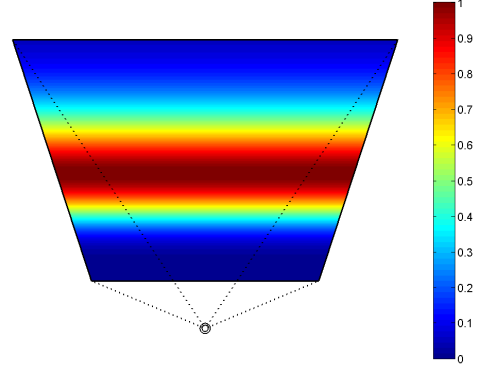where P_i is given by (44) and Z_i is given by (45).

2) Sensor Model: As in Section V-A2, the performance of this sensor is a function of the resolution $\mathscr{N}_i$. The derivation is analogous to section V-A2, where

$$\mathscr{N}_i = \frac{N_H N_V}{l_H l_V} \frac{f_i^2}{Z_i^2}. \quad (50)$$

This can be further simplified as

$$\mathscr{N}_i = \mathscr{K}_{PTZ} \frac{f_i^2}{Z_i^2} \quad (51)$$

where $\mathscr{K}_{PTZ}$ is a constant that depends on the physical characteristics of the camera. The probability of detection as a function of resolution and orientation are modeled exactly as described in Section V-A2. No detection is possible behind obstacles, therefore the probability of detection for the pan-tilt-zoom camera is

$$p_i = \begin{cases} p_0 p_{\mathscr{N}_i} p_\alpha & \text{if } q \in V_i \\ 0 & \text{otherwise} \end{cases} \quad (52)$$

where p_0 is the maximum detection probability by the camera.

3) Controller: In order to obtain the gradient ascent controller, the partial derivative of the detection probability should be obtained for region V_i. Since the parameters for the static pan-tilt-zoom camera case are purely rotational, the visible boundary does not vary with respect to x_i.

a) Gradient at interior points: The partial derivative of p_i with respect to x_i for the pan-tilt-zoom camera case is

$$\frac{\partial p_i}{\partial x_i} = \left[-\frac{(\mathscr{N}_i - \mathscr{N}_\mu)}{\sigma_{\mathscr{N}}^2} \frac{\partial \mathscr{N}_i}{\partial x_i} - \frac{\text{dist}_g(\alpha, \theta_i)}{\sigma_\alpha^2} \frac{\partial \theta_i}{\partial x_i} \right] p_i. \quad (53)$$

where $\frac{\partial \theta_i}{\partial x_i} = [1 \, 0 \, 0]^T$ and from (51) we have

$$\frac{\partial \mathscr{N}_i}{\partial x_i} = \frac{2\mathscr{K}_{PTZ}}{Z_i^3} \left[Z_i \frac{\partial f_i}{\partial x_i} - f^2 \frac{\partial Z_i}{\partial x_i} \right] \quad (54)$$

where $\frac{\partial f_i}{\partial x_i} = [0 \, 0 \, 1]^T$ and from (45), we have

$$\frac{\partial Z_i}{\partial x_i} = \begin{bmatrix} -\sin(\beta_i)\sin(\theta_i) & \cos(\beta_i)\cos(\theta_i) \\ \cos(\beta_i)\cos(\theta_i) & \cos(\beta_i)\sin(\theta_i) \\ 0 & 0 \end{bmatrix} (q - c_i) + h_i \begin{bmatrix} 0 \\ \sin(\beta_i) \\ 0 \end{bmatrix}. \quad (55)$$

Applying the above to the first term of (9) we have the gradient for the region V_i given as

$$\mathscr{G}_{V_i} = \int_\Theta \int_{V_i} \frac{\partial p_i}{\partial x_i} \prod_{j \in \mathscr{B}_i} [1 - p_j] \phi(q, \alpha) dq d\alpha$$

b) Gradient at coverage boundary: In contrast to the previous two cases, in this case the coverage boundary changes with the sensor parameters. We are interested in obtaining the variation of the boundary with respect to the camera parameters x_i. From the perspective equation, we have from (48) the partial

$$\frac{\partial}{\partial x_i}\left(\begin{bmatrix} x_{image_i} \\ y_{image_i} \end{bmatrix}\right) = \frac{\partial}{\partial x_i}\left(f_i \frac{P_i}{Z_i}\right). \quad (56)$$

The image coordinates does not change as a function of the camera parameters. Thus, from (56) we can deduce

$$\frac{1}{f_i} Z_i P_i \frac{\partial f_i}{\partial x_i} + Z_i \frac{\partial P_i}{\partial x_i} - P_i \frac{\partial Z_i}{\partial x_i} = 0. \quad (57)$$

Substituting for the gradients of P_i and Z_i, we can deduce

$$\frac{\partial q}{\partial x_i} = (Z_i T - P_i t)^{-1} \left[-\frac{1}{f_i} Z_i P_i \frac{\partial f_i}{\partial x_i} + \left(P_i \frac{\partial t}{\partial x_i} - Z_i \frac{\partial T}{\partial x_i}\right)(q - c_i) + h_i \left(P_i \frac{\partial v}{\partial x_i} - Z_i \frac{\partial V}{\partial x_i}\right) \right].$$

Thus, for the pan parameter θ_i we have (after some algebraic simplification)

$$\frac{\partial q}{\partial \theta_i} = \begin{bmatrix} c_{y_i} - q_y \\ q_x - c_{x_i} \end{bmatrix} \quad (58)$$

For the tilt parameter, β_i, we have

$$\frac{\partial q}{\partial \beta_i} = -\frac{1}{h_i} \begin{bmatrix} \cos(\beta_i) h_i^2 + (\Delta q_{x_i})^2 \cos(\beta_i) + (\Delta q_{x_i})(\Delta q_{y_i}) \sin(\beta_i) \\ \sin(\beta_i) h_i^2 + (\Delta q_{y_i})^2 \sin(\beta_i) + (\Delta q_{x_i})(\Delta q_{y_i}) \cos(\beta_i) \end{bmatrix} \quad (59)$$

where $\Delta q_{x_i} = q_x - c_{x_i}$ and $\Delta q_{y_i} = q_y - c_{y_i}$.

For the focal length parameter f_i we have

$$\frac{\partial q}{\partial f_i} = -(Z_i T - P_i t)^{-1} \frac{Z_i P_i}{f_i} \quad (60)$$

Expanding and simplifying (60) (steps not shown here), we obtain

$$\frac{\partial q}{\partial f_i} = -\frac{1}{f_i}\left[\frac{X_i}{h_i}\left(\sin(\theta_i)(q - c_i) - h_i \cos(\theta_i)\begin{bmatrix}\cos(\beta_i) \\ \sin(\beta_i)\end{bmatrix}\right) + \begin{bmatrix} \sin(\beta_i)\sin(\theta_i) & \sin(\beta_i)\cos(\theta_i) \\ -\cos(\beta_i)\sin(\theta_i) & -\cos(\beta_i)\cos(\theta_i) \end{bmatrix}(q - c_i) \right]$$

Note that points in the boundary of the pan-tilt-zoom coverage vary as the following

$$\frac{\partial q_{r_k}}{\partial x_i} = \begin{bmatrix} \frac{\partial q}{\partial \theta_i} & \frac{\partial q}{\partial \beta_i} & \frac{\partial q}{\partial f_i} \end{bmatrix}.$$

Thus, for points on the coverage boundary we have

$$\mathscr{G}_{\mathrm{bd}(V_i)} = \sum_{r_k \in \mathrm{Dscn}_{\mathrm{bd}(V_i)}(p_i)} \int_\Theta \int_{r_k} p_i \prod_{j \in \mathscr{B}_i} [1 - p_j] \frac{\partial q_{r_k}}{\partial x_i}^T n_k \phi dq d\alpha$$

c) Control law: After calculating all the terms as in (9), we obtain the gradient

$$\frac{\partial \mathscr{H}(\mathbf{x})}{\partial x_i} = \mathscr{G}_{V_i} + \mathscr{G}_{\mathrm{bd}(V_i)},$$

and we can update the control rule by following the control rule in (9). Note that $\mathscr{G}_{\mathrm{vis}} = 0$ in this case, since the camera positions are fixed and visibility boundaries depend only on camera positions and not on the states, i.e., the pan, tilt, and zoom angles.

VI. Overall Algorithm

In this section, we discuss the implementation of the overall distributed gradient-ascent algorithm for the collection of agents. First, note that to compute the integrals of (9), the region of coverage must be discretized. Let $\widehat{V_i \cup \bar{V}_i}$ be the discretized set of points in the visible and invisible regions, $\widehat{\Theta}$ be the discretized orientation space and $\widehat{r_k}$ be the discretized set of discontinuity points. The discussion in the previous sections was centered around finding a direction of motion by using local sensing information. However, due to the presence of obstacles or forbidden regions in the environment, the robots may not be able to take a full step along the computed direction. To ensure that the action of the robot is feasible with respect to all the other constraints we need to modify the computed control input.

Algorithm 1 summarizes the computation done by each agent for computing its control action. Line 7 gives the computation of the control input to improve the sensing performance. It is a discrete form of Equation (9) with Δq and $\Delta \alpha$ being the size of the discrete interval. The function *Reproject* in Line 8 modifies this control input. Re-projection consists of the following steps:

Algorithm 1 Distributed Discrete Controller

1: *Require: Sensor i knows its parameters x_i, the extent of the region Q, obstacles in Q, the traversable region T and the anisotropic density function $\phi(q, \alpha)$.*
2: *Require: Sensor i can communicate with the set of neighbors that intersect the region of coverage $\mathscr{B}_i$.*
3: *Require: Sensor i can compute p_j for all neighbor sensors $j \in \mathscr{B}_i$.*
4: **if** Convergence criterion is not met **then**
5:　　Communicate with neighbors and acquire $x_j \in \mathscr{B}_i$
6:　　Compute
7:　　$u_i^-(t) = K \sum_{\alpha \in \widehat{\Theta}} \sum_{q \in \widehat{V_i \cup \bar{V}_i}} \frac{\partial p_i}{\partial x_i} \prod_{j \in \mathscr{B}_i} [1 - p_j] \phi(q, \alpha) \Delta q \Delta \alpha$
　　　　$+ K \sum_{r_k} \sum_{\alpha \in \widehat{\Theta}} \sum_{q_{r_k}} \frac{\partial q_{r_k}}{\partial x_i}^T n_k \Phi_k(\mathbf{x}, q, \alpha) \phi(q_{r_k}, \alpha) \Delta q_{r_k} \Delta \alpha$
8:　　$u_i(t) = \mathrm{Reproject}(x_i(t), u_i^-(t))$
9:　　Move to $x(t + \Delta T) = x(t) + u_i(t)$
10: **end if**

---

d) Collision avoidance: When the sensors are on mobile robots, we use a potential field approach to avoid collisions. The repulsion force from the obstacles follows the inverse square law.

$$\rho_i = \sum_j \frac{1}{\|c_i - p_j\|^2}(c_i - p_j),$$

where p_j is either the closest point to the polygonal edge of Q or an obstacle H_j, or the position of another agent $c_j \in \mathcal{B}_i$. The repulsive action u_{rep} is:

$$u_{rep} = k_{rep} \max\{0, \|\rho_i\| - \rho_0\} \rho_i,$$

where k_{rep} is a proportional gain to the repulsive action and ρ_0 is the minimum value of $\|\rho_i\|$ for a repulsive action. The value of u_{rep} is added to u_i^- to obtain the next action. Note that the use of repulsive potential functions is one approach. We could also have used other approaches for obstacle avoidance [26].

e) Physical limitations: Robots have physical limitations such as maximum speed, maximum angle for pan and tilt, etc. The output of the command is clamped to the maximum and minimum value for actions u_i and the state x_i.

f) Traversable region: For mobile agents, the position c_i is constrained to the traversable region $T \subseteq Q$. If the mobile agent is projected to move outside of the traversable region T, the next position c_i is reprojected to the closest position inside the traversable region T.

VII. SIMULATION RESULTS

Here we show some example simulations of the proposed algorithm developed in a Matlab environment. The VisiLibity Library [27] was used to obtain the visible region. We performed simulations with both deterministic communication models and probabilistic communication models. In deterministic communication models, it is assumed that robots can always communicate with their neighbors (i.e., robots with whom their sensor field-of-view overlap). In the probabilistic model, the probability of communication link failure between two agents at any time was modeled as linearly decreasing from 0 at zero distance to 1 at a distance of 60m (similar to [20]). Thus, this model allows for intermittent communication dropouts between neighbors. The results shown here are for the probabilistic communication model. In all examples, the repulsion term for obstacle and collision avoidance k_{rep} and ρ_i were set to 1 for mobile sensors. The density function used in all the examples (except Example 2) is shown in Figure 7.

We consider three different scenarios. In the first scenario, we consider a group of holonomic mobile robots with cameras, the second is a group of holonomic mobile robots with acoustic receivers, and the third is a group of static pan-tilt-zoom cameras. Although in the examples, there is only one type of sensor for each example, they are for illustration only. Our framework can easily be used for teams of robots with different sensors and with robots containing multiple sensors with different modality of sensing and sensor footprint (e.g., acoustic receiver and PTZ camera).

A. Mobile Robot with Camera

We consider a collection of mobile robots equipped with cameras in a non-convex environment with obstacles and anisotropic density function. In this example, ten robots are to be assigned to maximize the probability of detection over a non-convex region with maximum length and width of 60m. We also assume here poor sensing capabilities for these sensors, so overlap between the regions of coverage improve the overall detection performance. The parameters of the image sensor sensor are $N_H = 640$ pixels, $N_V = 480$ pixels, $l_H = 3.04$mm, $l_V = 1.98$mm, and focal distance $f = 3$mm. The optimal parameters for a target detection was set as $\mathcal{N}_\mu = 3840$ pixel^2/m^2, $\mathcal{N}_\sigma = 2800$ pixel^2/m^2. The minimum distance and maximum distance for an observation are $Z_{min} = 1.5$m and $Z_{max} = 22.5$m, respectively. For the anisotropic term, the parameter $\sigma_\alpha = \pi/6$ rad was assigned to the sensor. The maximum set detection probability for each robot is $p_0 = 0.2$. The control gains used for the position c_i were 1 and for the rotation θ_i was 0.05.

The result of the simulation is shown in Figure 8. Figure 8a shows the initial position and Figure 8c shows the final position of the robots. As can be observed, from Figure 8c, the sensor footprint of the robots overlap to increase the overall detection performance, since the individual cameras have low detection performance. Furthermore, the orientation of the robots are in the direction that enhances detection performance. The paths taken by the individual robots are shown by the dotted lines. Figure 8b shows the variation of the objective function with the iterations, which shows that the objective function increases monotonically.

B. Mobile Robot with Acoustic Receiver

Here we present simulation results for mobile robots equipped with acoustic receivers. The first example is the example shown in Figure 2 that illustrates that even when no prior information is known about the environment the algorithm optimizes the placement and orientation of the sensors. In this example, four mobile directional sound sensors are assigned to maximize the probability of detection over a square convex region of 40m on each side. The second example is a more challenging non-convex environment with obstacles and multiple regions of interest. In this example, eight mobile directional sensors are assigned to maximize the probability of detection over a non-convex region with maximum length and width of 60m.

The parameters of the acoustic receiver are minimum radius of sensing $D_{min} = 0.5$m and maximum radius of sensing $D_{max} = 12$m and a microphone constant $b_{mic} = 2$m^2. The optimal parameters for a target detection was set as $I_\mu = 1$, $I_\sigma = 0.9$. For the anisotropic term, the parameter $\sigma_\alpha = 3\pi/4$rad was assigned to the sensor. The maximum detection probability for each robot in the visible region is $p_0 = 1$ and in the invisible region $\tilde{p}_0 = 0.5$. The control gains used for the position c_i were 0.02 and for the rotation θ_i was 0.003.

The result of the simulation is shown in Figure 9. Figures 9a and 9c are the initial and final positions of the robots. The paths the robots take are shown by the dotted lines. Note the path taken by robot 3, which shows the combined effect of the repulsive forces along with the force along the direction of maximum gradient. Figure 9b shows the monotonic increase in objective function as the iterations progress.

C. Static Pan-Tilt-Zoom Cameras

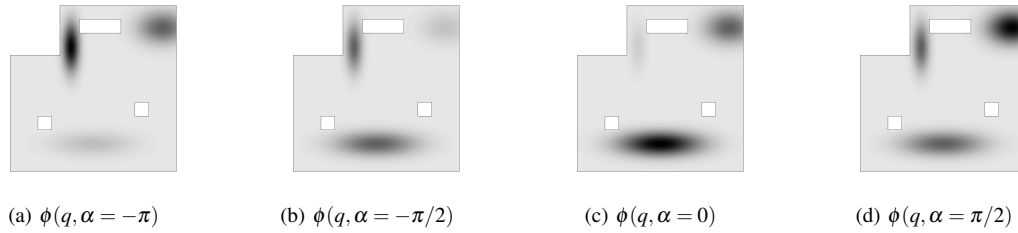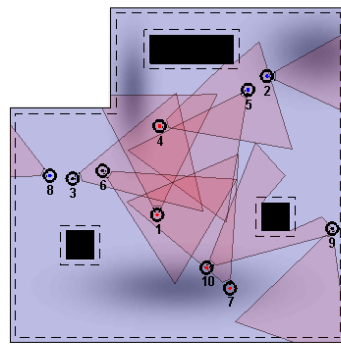We now present simulation results for a collection of static pan-tilt-zoom (PTZ) cameras in a non-convex environment
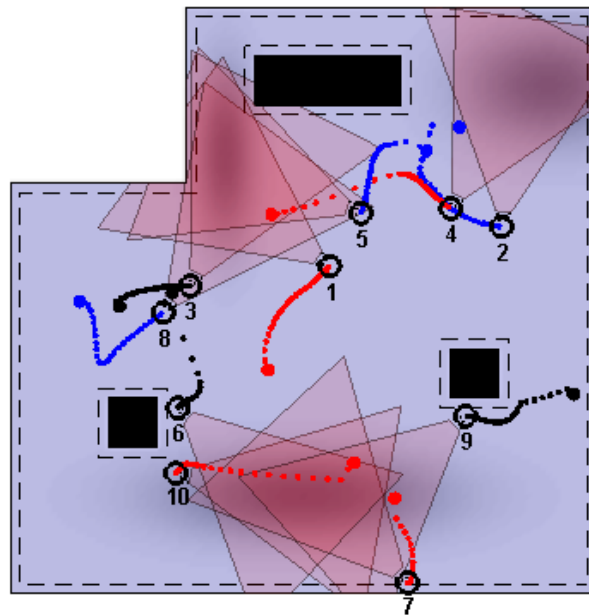
(a) $\phi(q, \alpha = -\pi)$ (b) $\phi(q, \alpha = -\pi/2)$ (c) $\phi(q, \alpha = 0)$ (d) $\phi(q, \alpha = \pi/2)$

Fig. 7: The density function is shown at different orientations α in Figure 7a through Figure 7d, where darker areas indicate higher values of ϕ when a point is observed at orientation α. White areas indicate $\phi(q, \alpha) = 0$ and black areas $\phi(q, \alpha) = 1$.
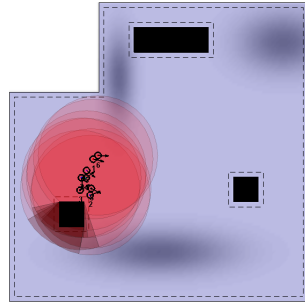

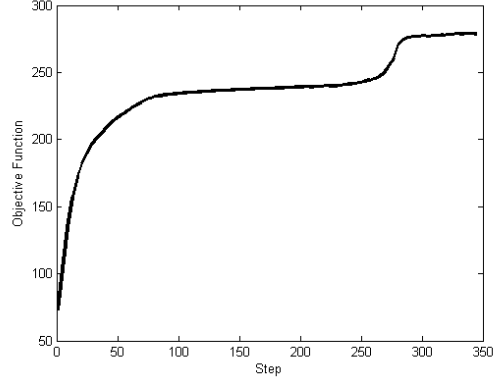(a) Initial Position   (b) Objective Function
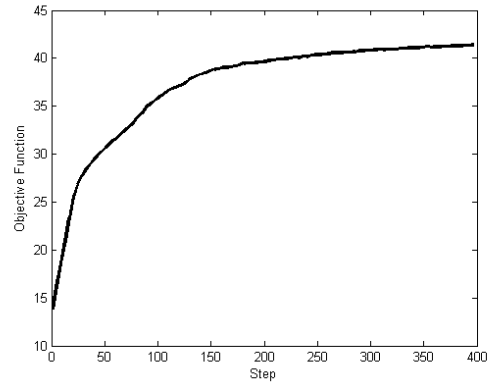
(c) Final Configuration

Fig. 8: Example 1 - Simulation of mobile camera sensors in a non-convex environment Q with obstacles. Dotted lines indicate the traversable region T. Note that the regions of coverage align to the orientation where $\phi(q, \alpha)$ is maximum.

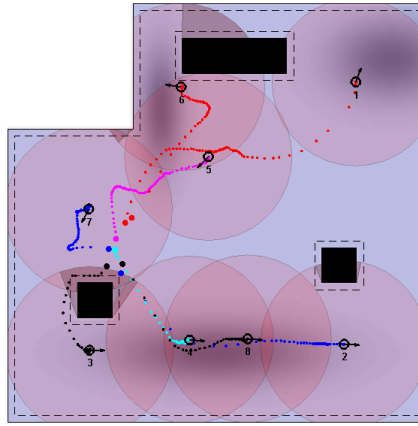with obstacles and anisotropic density function. In this example, six static PTZ cameras are assigned to maximize the probability of detection over a non-convex region with maximum length and width of 60m. The parameters of the camera

(a) Initial Position

(b) Objective Function

(c) Final Configuration

Fig. 9: Example 2 - Simulation of mobile directional sound sensors in a non-convex environment Q with obstacles. Dotted lines indicate the traversable region T. Darker areas in the region of coverage indicate the invisible region $\bar{V}_i$. Notice how the regions of coverage align to the orientation where $\phi(q, \alpha)$ is maximum.

are $N_H = 640$ pixels, $N_V = 480$ pixels, $l_H = 3.04$mm, $l_V = 1.98$mm, and focal distance $f = 3$mm. The optimal parameters for a target detection was set as $\mathcal{N}_\mu = 4000$ pixel^2/m^2, $\mathcal{N}_\sigma = 4000$ pixel^2/m^2. The cameras are placed at a height $h = 5$m. The maximum detection probability for each robot at any given moment was limited to $p_0 = 1$, in order to maximize the covered area. The control gains used for the pan and tilt parameters where $1e-3$ and $1e-4$ for the focal distance term. The result of the simulation is shown in Figure 10.

Figures 10a and 10c shows the initial and the final regions covered by the cameras. The cameras are fixed near the walls or the columns (obstacles) in the environment. The pan and tilt angles, as well as the zoom f the cameras are changed (e.g., see camera 1 and camera 3) to maximize performance detection. Figure 10b shows the monotonic increase in objective function as the iterations progress.

VIII. CONCLUSION

In this paper, a (generalized) gradient-ascent coverage control algorithm for maximizing the joint probability of detec-

tion of events over a planar non-convex environment with polygonal obstacles was presented. Realistic sensor models that capture the anisotropic nature of sensing performance was explicitly modeled. This approach also captures the property of preferred sensing orientation of certain types of events. For this, an anisotropic density function that rewards observations according to the orientation in which an observation is being made was defined. The discontinuities such as sensing and visibility boundaries that should be taken into consideration when obtaining the gradient were considered. Furthermore, some common assumptions in anisotropic sensing coverage were relaxed, such as detection performance decay is inversely proportional to distance. Case studies for common anisotropic sensors (like cameras and acoustic receivers) were shown and simulation results of this technique being applied to different scenarios where a network of agents jointly maximized the probability of detection over a region with known density function was shown.
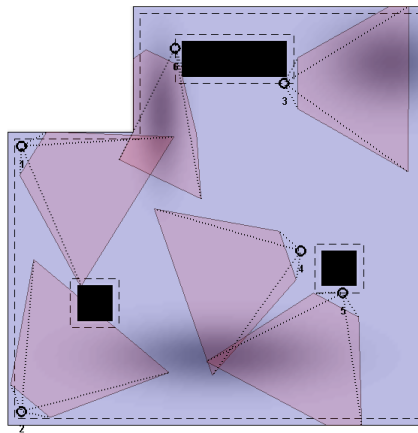
There are a few directions along which this work can be extended. The gradient-based approach along with the poten-
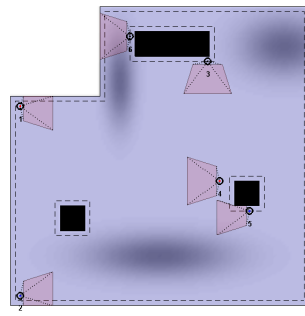
(a) Initial Position
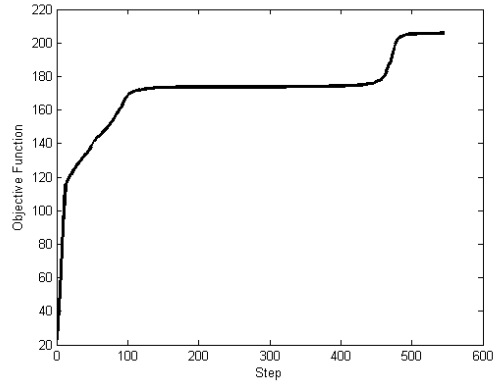
(b) Objective Function

(c) Final Configuration

Fig. 10: Example 3 - Simulation of static placed Pan-Tilt-Zoom cameras in a non-convex environment Q with obstacles. Dotted lines indicate the traversable region T.

tial field for obstacle avoidance may have slow convergence under certain configurations of the robot, especially in the presence of obstacles. Thus, devising improved techniques that enhances the convergence properties of the system would be useful. Furthermore, the proposed method converges to a local maxima and it is not clear apriori how close the local maxima is to the global maxima.. Thus, an open problem is to theoretically characterize the worst case performance of this algorithm. Finally, the performance function over the environment may not be known for some application scenarios. Devising algorithms for such systems under the sensing models described here is another avenue of future research.

REFERENCES

[1] H. Choset, "Coverage for robotics: A survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, pp. 113–126, May 2001.

[2] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*. Princeton University Press, 2009.

[3] A. Renzaglia, L. Doitsidis, A. Martinelli, and E. B. Kosmatopoulos, "Multi-robot three-dimensional coverage of unknown areas," *The International Journal of Robotics Research*, vol. 31, no. 6, pp. 738–752, 2012.

[4] K. Laventall and J. Cortes, "Coverage control by robotic networks with limited-range anisotropic sensory," in *2008 American Control Conference*, Jun 2008, pp. 2666 –2671.

[5] A. Gusrialdi, T. Hatanaka, and M. Fujita, "Coverage control for mobile networks with limited-range anisotropic sensors," in *47th IEEE Conference on Decision and Control*, Dec 2008, pp. 4263 –4268.

[6] A. Gusrialdi, S. Hirche, T. Hatanaka, and M. Fujita, "Voronoi based coverage control with anisotropic sensors," in *2008 American Control Conference*, Jun 2008, pp. 736 –741.

[7] M. Zhong and C. G. Cassandras, "Distributed coverage control and data collection with mobile sensor networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2445 –2455, Oct. 2011.

[8] J. G. Desloge, W. M. Rabinowitz, and P. M. Zurek, "Microphone-array hearing aids with binaural output. I. fixed-processing systems," *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 6, pp. 529 –542, Nov 1997.

[9] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243 – 255, Apr 2004.

[10] S. Bhattacharya, N. Michael, and V. Kumar, "Distributed coverage and exploration in unknown non-convex environments," in *Distributed Autonomous Robotic Systems*, ser. Springer Tracts in Advanced Robotics, A. Martinoli, F. Mondada, N. Correll, G. Mermoud, M. Egerstedt, M. A. Hsieh, L. E. Parker, and K. Sty, Eds. Springer Berlin Heidelberg, 2013, vol. 83, pp. 61–75.

[11] B. Hexsel, N. Chakraborty, and K. Sycara, "Coverage control for mobile anisotropic sensor networks," in *IEEE International Conference on*

Robotics and Automation, Shanghai, China, May 2011, pp. 2878–2885.

[12] J. O'Rourke, *Art gallery theorems and algorithms*. New York, NY, USA: Oxford University Press, Inc., 1987.

[13] S. Fleishman, D. Cohen-Or, and D. Lischinski, "Automatic camera placement for image-based modeling," in *PG '99: Proceedings of the 7th Pacific Conference on Computer Graphics and Applications*. Washington, DC, USA: IEEE Computer Society, 1999.

[14] R. Bodor, A. Drenner, P. Schrater, and N. Papanikolopoulos, "Optimal camera placement for automated surveillance tasks," *Journal of Intelligent and Robotic Systems*, vol. 50, no. 3, pp. 257–295, 2007.

[15] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: a distributed scalable solution to the area coverage problem," in *Proceedings of the 6th International Symposium of Distributed Autonomous Robotic Systems*, 2002, pp. 299–308.

[16] A. Ganguli, J. Corte's, and F. Bullo, "Maximizing visibility in nonconvex polygons: Nonsmooth analysis and gradient algorithm design," *SIAM J. Control Optim.*, vol. 45, no. 5, pp. 1657–1679, Nov. 2006.

[17] C. H. Caicedo-Nunez and M. Zefran, "Performing coverage on nonconvex domains," in *IEEE International Conference on Control Applications*, Sep 2008, pp. 1019 –1024.

[18] C. G. Cassandras and W. Li, "Sensor networks and cooperative control," in *44th IEEE Conference on Decision and Control*, Dec 2005, pp. 4237 – 4238.

[19] A. Breitenmoser, M. Schwager, J. C. Metzger, R. Siegwart, and D. Rus, "Voronoi coverage of non-convex environments with a group of networked robots," in *IEEE International Conference on Robotics and Automation (ICRA 10)*, May 2010, pp. 4982–4989.

[20] M. Schwager, B. J. Julian, and D. Rus, "Optimal coverage for multiple hovering robots with downward facing cameras," in *IEEE International Conference on Robotics and Automation (ICRA '09)*, May 2009, pp. 3515 –3522.

[21] M. Zhong and C. Cassandras, "Distributed coverage control in sensor network environments with polygonal obstacles," Dept. of Manufacturing Engineering and Center for Information and Systems Engineering, Boston University, Brookline, MA 02446, Tech. Rep., 2008. [Online]. Available: http://codescolor.bu.edu/docs/ifac08final.pdf

[22] M. Zhu and S. Martinez, "Distributed coverage games for mobile visual sensors (I): Reaching the set of nash equilibria," in *48th IEEE Conference on Decision and Control Decision and Control*, Dec 2009, pp. 169 –174.

[23] ——, "Distributed coverage games for mobile visual sensors (II): Reaching the set of global optima," in *48th IEEE Conference on Decision and Control Decision and Control*, Dec 2009, pp. 175 –180.

[24] W. Zhao, R. Chellappa, A. Rosenfeld, and P. Phillips, "Face recognition: A literature survey," 2000.

[25] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1995.

[26] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations (Intelligent Robotics and Autonomous Agents series)*. The MIT Press, 2005.

[27] K. J. Obermeyer and Contributors, "The VisiLibity library," http://www.VisiLibity.org, 2008, r-1.