Coverage Control for Mobile Anisotropic Sensor Networks

Bruno Hexsel, Nilanjan Chakraborty, *Member, IEEE*, and Katia Sycara, *Fellow, IEEE*

***Abstract*— Distributed algorithms for (re)configuring mobile 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 paper, we consider a system of holonomic mobile robots equipped with anisotropic sensors (e.g., limited field of view cameras) that are required to cover a polygonal region with polygonal obstacles to detect interesting events. We assume a given probability distribution of the events over a region. 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 orientation of observation. We present a distributed gradient-ascent algorithm for reconfiguring the system of mobile robots so that the joint probability of detection of events over the whole region is maximized (i.e., positioning the mobile robots and determining their sensor parameters). As an example case study, we use a system of mobile robots equipped with limited field of view cameras with pan and zoom capabilities. We present simulation results demonstrating the performance of our algorithm.**

***Index Terms*— Distributed Coverage, Anisotropic Sensing.**

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). More recently, distributed algorithms for (re)configuring mobile sensors to cover a given area for surveillance and environmental monitoring applications have received widespread attention [1]. In most of this literature (except [2], [3], [4]), 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 limited field of view (FoV) cameras or acoustic receivers cannot be modeled as isotropic sensors. In such scenarios, the probability of detection of an event depends 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 also dependent on the orientation at which a point in the region is being sensed. We consider a system of holonomic mobile robots equipped with anisotropic sensors (e.g., limited field of view 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 segments of a disc [2], [3] or an ellipse [4]. In both cases, the sensor position is always within the region of coverage and the sensor performance is assumed to be inversely proportional to the distance of the points being sensed. In our paper, the region of coverage of the sensors can be any bounded set (we assume a trapezium in our illustrative example). The sensor position need not be contained within the region of coverage (see Figure 1). Thus, in our sensor model, the performance of the sensor does not necessarily deteriorate monotonically with distance from the sensor. This sensor model is relevant in situations where we may not detect an event if it is too close to 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. In addition, we assume our sensing performance depends on the sensor model and the angle at which an object is being sensed.

Since we do not assume that the sensor positions are part of the region of coverage of the sensor, there is a region near the sensor that may lie outside the coverage region. With this model, regions near the sensor may not perform well, hence using Voronoi region based distributed coverage algorithms (as introduced in [5] and also used for anisotropic sensors in [2]) 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 robots and the sensor parameters to maximize the joint detection performance of events by all the robots. Like Voronoi region based coverage algorithms, our gradient-ascent algorithm converges to a local optimal solution of the objective function. Due to the presence of obstacles and bounded field of view of our sensors, our objective function is not continuous and not differentiable. Consequently, we use tools from non-smooth analysis to form the generalized gradients of the objective function at various discontinuities (as also done in [1], [2]). To illustrate our algorithm, we use an example case study of a group of mobile robots with pan and zoom cameras

The authors are with the Robotics Institute, School of Computer Science, Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213, {bhexsel, nilanjan, katia}@cs.cmu.edu

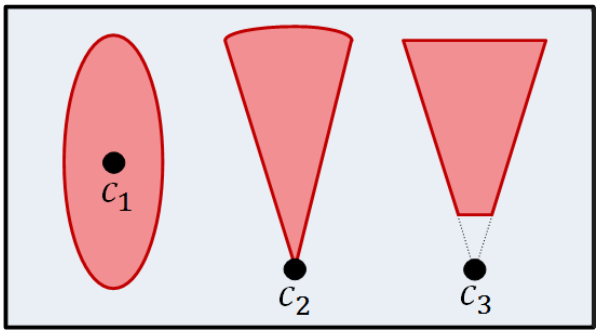


Fig. 1. Examples of anisotropic regions of coverage. The black dots are the robot positions and the shaded regions are their anisotropic sensing areas. The rightmost figure shows a coverage region not connected to the robot positioned at c_3. Here, the sensing performance is not a monotonically decreasing function of the distance from the robot. All these cases can be treated with the technique presented in this paper.

trying to maximize their performance while operating in an environment with obstacles.

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 in Section V and present simulation results in VI. Finally, in Section VII we present the conclusion and point out future research directions.

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 have 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 [6] for covering a given polygonal region with polygonal obstacles with omni-directional infinite range sensors 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 [7], 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 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 intensively in recent years ([5], [8] are two of the initial papers). In [8], the authors use a potential field approach for deploying a system of sensor nodes for maximizing area coverage. In [5], 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 [5] assumed a convex polygonal environment, this approach has been extended subsequently to consider nonconvex environments with obstacles and limited FoV sensors in [1], [9], [10]. In [11], 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 [10] and the work was extended to regions with polygonal obstacles in [12]. Game theoretic models for coverage control have been proposed in [13], [14].

In [2], [3], [4], 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 [3] and as an ellipse in [4]. For anisotropic sensors, the method in [5] 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 [2] 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 [4], 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 [5] 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 optimized by maximizing the coverage over the individual Voronoi regions of the robots (a key property of the metric proposed in [5]).

III. Problem Formulation

Here we describe some of the notation used in this paper. 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 Sphere space, respectively. Let $\text{int}(S)$ and ∂S denote the interior and the boundary of set S, respectively. Let $\mathbf{n}$ denote the outward unit normal vector at a boundary of a set.

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

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

where $\text{dist}_{\text{C}}(\alpha,\beta) = (\alpha-\beta) \mod 2\pi$ is the clockwise distance and $\text{dist}_{\text{CC}}(\alpha,\beta) = (\beta-\alpha) \mod 2\pi$ is the counterclockwise distance.

Let the closed segment between the points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ be denoted by $[\mathbf{x}, \mathbf{y}] = \{\lambda\mathbf{x} + (1-\lambda)\mathbf{y} \mid \lambda \in [0,1]\}$. For a set $S \subset \mathbb{R}^d$, a point $\mathbf{y} \in S$ is said to be *visible* from $\mathbf{x} \in S$ if the closed segment $[\mathbf{x}, \mathbf{y}]$ is contained in S. The *visibility set* $\mathrm{Vi}(\mathbf{x}, S)$ is the set of all points in S visible from $\mathbf{x}$.

A. ENVIRONMENT

The environment to be covered is assumed to be a planar bounded environment $Q \in \mathbb{R}^2$ defined by a nonconvex 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 and unfeasible for the robots to traverse, so the feasible subset of Q is

$$F = Q \setminus \left(\bigcup_{j=1}^{m} \mathrm{int}(H_j) \right) \tag{2}$$

Some points in the set F may not be accessible to the robots for reasons such as obstacle avoidance and forbidden zones. We define the traversable space $T \subseteq F$ as the set of all possible locations where a robot can be located.

B. ANISOTROPIC DENSITY FUNCTION

In this paper we present an *anisotropic density function* defined both in position and orientation in the special Euclidean group $SE(2)$. This function is defined by the map $\phi : \mathbb{R}^2 \times \mathbb{S}^1 \to \mathbb{R}_{\geq 0}$ and represents the probability or a measure of information about events that happened at a certain point $\mathbf{q} \in Q \subset \mathbb{R}^2$ when observed at an orientation $\alpha \in \mathbb{S}^1$. It is assumed that $\phi(\mathbf{q}, \alpha) = 0 \ \forall \ \mathbf{q} \notin F$ and $\int_\Theta \int_Q \phi(\mathbf{q}, \alpha) dq d\alpha < \infty$, where $\Theta = (-\pi, \pi]$. This orientation is not to be confused with the angle between a point $\mathbf{q}$ and the robot, as it is a property of the density function independent of the state of the robot.

The intuition behind the orientation parameter α in the function $\phi(\mathbf{q}, \alpha)$ 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.

C. ANISOTROPIC SENSING MODEL

The configuration of the robot along with the sensor parameters influence the observability of events. The configuration of the robot, i, consists of its center position, $\mathbf{c}_i = [c_{ix} \ c_{iy}]^T$ and its heading θ_i. Robots are only allowed to move along the traversable space (T) described in III-A. Let the parameter space for the sensor specific parameters be denoted by $\mathcal{P}$. For example, for a camera, pan, tilt, and zoom of the camera may constitute the parameter space. The combined parameter-space is $\mathcal{X} = T \times \mathbb{S}^1 \times \mathcal{P}$. For multiple robots with identical sensors, the combined parameter-space is denoted by the cartesian product $\mathcal{X}^N$ of the individual parameter space. Let $\mathbf{x} = (\mathbf{x}_1, ..., \mathbf{x}_N) \in \mathcal{X}^N$ be the concatenated vector denoting the configuration of the system with $\mathbf{x}_i$ denoting the configuration (or parameters) of robot i.

The region covered by the robot i is denoted $\mathcal{R}_i$, which is defined based on the sensing properties of the robot and its current parameters. We place no restrictions on the geometry of $\mathcal{R}_i$, which can be, for instance, a disk, polygon, non connected regions, wedge, etc. The *visibility set* $\mathrm{Vi}(\mathbf{c}_i, F)$ is the set of all points in F visible from the robot position $\mathbf{c}_i$. Since $\mathrm{Vi}(\mathbf{c}_i, F) \subset F$, the set of all visible points covered by robot i is $V_i = \mathcal{R}_i \cap \mathrm{Vi}(\mathbf{c}_i; F)$. The set of invisible points covered by the sensor is $\bar{V}_i = (\mathcal{R}_i \cap F) \backslash V_i$.

The probability of detection is given by the function $p : \mathcal{X} \times Q \times \mathbb{S}^1 \to [0,1]$. This probability not only depends on the specifics of the *possible* multiple sensors on each robot, 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 nonconvexity of region Q. Therefore, the detection probability has to be considered for both the visible and invisible sets of points as

$$p_i(\mathbf{x}_i, \mathbf{q}, \alpha) = \begin{cases} \hat{p}_i(\mathbf{x}_i, \mathbf{q}, \alpha) & \text{if } \mathbf{q} \in V_i \\ \tilde{p}_i(\mathbf{x}_i, \mathbf{q}, \alpha) & \text{if } \mathbf{q} \in \bar{V}_i \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

where $\hat{p}_i(\mathbf{x}_i, \mathbf{q}, \alpha)$ is the detection probability for a point $\mathbf{q}$ visible from robot i and $\tilde{p}_i(\mathbf{x}_i, \mathbf{q}, \alpha)$ is the detection probability from a point $\mathbf{q}$ invisible from robot i. In some cases, events invisible to the robot may not be sensed at all, leading to a detection probability $\tilde{p}_i(\mathbf{x}_i, \mathbf{q}, \alpha) = 0$.

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

$$P(\mathbf{x}, \mathbf{q}, \alpha) = 1 - \prod_{j=1}^{N} [1 - p_j(\mathbf{x}_j, \mathbf{q}, \alpha)]. \tag{4}$$

Therefore, the coverage problem can be stated as follows: Find a feasible configuration $\mathbf{x}$ of the multirobot system, that maximizes the joint probability of detection $\mathcal{H}(\mathbf{x})$ over the entire environment Q, i.e.,

$$\max_{\mathbf{x}} \mathcal{H}(\mathbf{x}) = \max_{\mathbf{x}} \int_\Theta \int_Q P(\mathbf{x}, \mathbf{q}, \alpha) \phi(\mathbf{q}, \alpha) dq d\alpha$$
$$\text{subject to } \mathbf{x} \in \mathcal{X}^N \tag{5}$$

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 gradient based controller to coordinate the multiple robotic nodes according to the optimization problem described in Equation (5).

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 points at which p_i is discontinuous be $\mathrm{Dscn}(p_i)$. These include

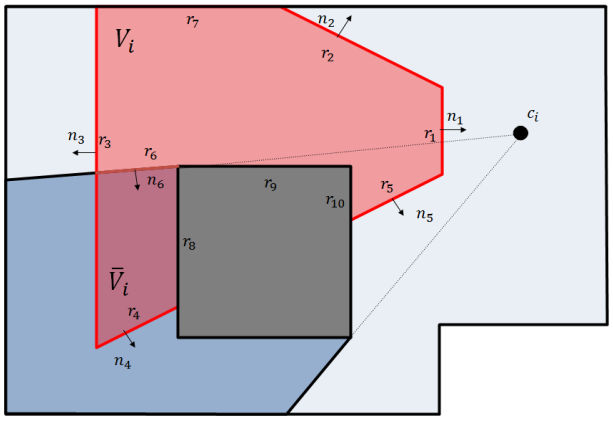


Fig. 2. This figure shows the different types of boundary intervals r_k. In gray is an obstacle. Boundaries r_1 through r_5 are coverage boundaries. Boundary r_6 is a visibility boundary. Boundaries r_7 through r_{10} are rigid boundaries and do not affect the gradient calculation. Also shown are the outward normal unit vectors n_k.

boundaries of the covered region, boundaries between the visible and invisible set, and possible discontinuities in the sensing model, as seen in Figure 2. Applying the rules of differentiation under the integral sign to the objective function $\mathcal{H}(\mathbf{x})$ in equation (5), we have

$$\frac{\partial \mathcal{H}(\mathbf{x})}{\partial \mathbf{x}_i} = \int_\Theta \int_{V_i \cup \bar{V}_i} \frac{\partial P(\mathbf{x},\mathbf{q},\alpha)}{\partial \mathbf{x}_i} \phi(\mathbf{q},\alpha) dq d\alpha + \sum_{r_k \in \text{Dscn}(p_i)} \int_\Theta \int_{r_k} \Phi_k(\mathbf{x},\mathbf{q},\alpha) \frac{\partial \mathbf{q}_{r_k}}{\partial \mathbf{x}_i}^T \mathbf{n}_k \phi(\mathbf{q},\alpha) dq d\alpha \quad (6)$$

where r_k is a discontinuity interval, $\mathbf{q}_{r_k}$ is, with abuse of notation, the point in r_k that lies on a discontinuity, and $\Phi_k(\mathbf{x},\mathbf{q},\alpha)$ is the difference of probabilities from a point immediately inside the discontinuity to a point immediately outside the discontinuity. Function $\Phi_k(\mathbf{x},\mathbf{q},\alpha)$ is defined as

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

where we define P_k^- and P_k^+ as

$$P_k^-(\mathbf{x},\mathbf{q},\alpha) = \lim_{\epsilon \to 0^+} P(\mathbf{x}, \mathbf{q} - \epsilon \mathbf{n}_k, \alpha) \quad (7)$$

$$P_k^+(\mathbf{x},\mathbf{q},\alpha) = \lim_{\epsilon \to 0^+} P(\mathbf{x}, \mathbf{q} + \epsilon \mathbf{n}_k, \alpha). \quad (8)$$

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

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + K \frac{\partial \mathcal{H}(\mathbf{x})}{\partial \mathbf{x}_i} \quad (9)$$

where K is a diagonal matrix with the individual gains for each of the parameters in $\mathbf{x}_i$ in its diagonal. The choice of gains for convergence are based on standard rules, for which we refer the reader to [15], [12], [3]. Note that this controller does not provide collision avoidance, for which we employ a simple heuristic. 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. In the future, we would like to incorporate collision avoidance within this framework using ideas from [16].

Communication Requirements: For robot i, let B_i be the set of robots with whom its sensing coverage area intersect, i.e.,

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

From Equation (3) we see that the detection probability is always equal to zero outside of $(V_i \cup \bar{V}_i)$, and does not contribute to the gradient computation. Only robots that intersect the visible and invisible coverage sets will contribute to the computation of the gradient (4). Therefore, each node requires local knowledge about its region of coverage and information about the detection performance of robots in the set $\mathcal{B}_i$ and the density function $\phi(\mathbf{q},\alpha)$ to compute (4). Therefore, for the controller in Equation 9 to be distributed, each robot, i, has to communicate with robots in the set B_i. This is easily achievable when the communication range of the robot is twice the sensing range.

Remark 1: To implement this controller, it is necessary to discretize the coverage region to compute the integrals over the possible orientations $\alpha \in \Theta$.

A. Gradient Terms

Here we explain how to obtain the terms in Equation (6). 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 Equation (6) deals with the discontinuities caused by the environment and/or the sensing model. In the interest of conciseness, we will omit the arguments $\mathbf{x}_i$, $\mathbf{q}$, and α of the function p.

1) Internal Gradient: The first term in Equation (6) 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 (4) we obtain

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

2) Rigid Boundary: These discontinuities are introduced by obstacles and the boundary of region Q. These points are in the set $\mathcal{R}_i \cap \partial F$. Since they do not vary as a function of the parameters $\mathbf{x}_i$, their gradient is zero.

$$\frac{\partial \mathbf{q}_{r_k}}{\partial \mathbf{x}_i} = 0 \; \forall \; \mathbf{q}_{r_k} \in \partial F$$

3) Coverage Boundary: Points on the boundary of the sensor coverage set $\partial \mathcal{R}_i \cap \text{int}(F)$ have non-zero detection performance immediately inside $V_i \cup \bar{V}_i$ and zero detection performance immediately outside this region. Thus, from Equation (7) we have

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

$$P_k^+(\mathbf{x},\mathbf{q},\alpha) = 1 - \prod_{j \in \mathcal{B}_i} [1 - p_j]. \quad (10)$$

The discontinuity probability function then becomes

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

The values of $\frac{\partial \mathbf{q}_{r_k}}{\partial \mathbf{x}_i}$ on the coverage boundary vary according to $\mathbf{x}_i$ and are sensor parameter dependent.

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 $\partial V_i \cap (\text{int}(\mathcal{R}_i) \backslash \partial F)$. Similarly to (10), we can apply (3) to obtain the discontinuity probability function for a visibility boundary

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

We notice that the visibility boundary is generated by a vertex either of the region Q or of an obstacle H_j. We refer the reader to [12] for an excellent geometric description on how the visibility boundary is generated in a region with polygonal obstacles.

V. CASE STUDY

In this section we are going to discuss how the distributed gradient algorithm previously described could be applied to an example camera coverage scenario. Since our algorithm is designed to cover a planar 2D environment, some task specific assumptions will be made in the problem formulation so that the complex volumetric coverage of a camera under perspective can be treated as a planar coverage problem.

A. Pan and Zoom 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 camera capable of panning and zooming. This robot has to cover an area in order to detect targets with a vertical height H and a horizontal length L which is assumed to be symmetric along its center axis. The robot is assumed to be smaller than the target and have the center of the camera sensor at height h aligned orthogonally to the ground plane. Obstacles interfere with visibility but other robots do not. The measure of performance is given by the pixel area $\mathcal{N}_t$ that a target would occupy in the image. The robot should be able to observe the target in its full height and, after the pixel area $\mathcal{N}_t$ drop below a value $\mathcal{N}_{\min}$, no further detection can be made. Moreover, a frontal observation of the target is preferred. The environment and the event density with the associated orientation $\phi(\mathbf{q}, \alpha)$ is known.

Let $\beta \in \mathbb{S}^1$, and $f \in (0, \infty)$ denote the pan and focal distance parameters, respectively. The sensor specific parameter-space is $\mathcal{P} = \mathbb{S}^1 \times (0, \infty)$. Thus, for robot i, $\mathbf{x}_i = [c_{xi}, c_{yi}, \theta_i, \beta_i, f_i]^T$. For a given image sensor, let l_x and l_y be the camera sensor height and length, respectively, and N_x and N_y be the number of pixels along the horizontal and vertical axes, respectively.

It is useful to analyze the environment in the camera coordinate frame. Let the rotation matrix $\text{Rot}_{\theta,\beta}$ be defined as

$$\text{Rot}_{\theta,\beta} = \begin{bmatrix} \cos(\theta+\beta) & -\sin(\theta+\beta) \\ \sin(\theta+\beta) & \cos(\theta+\beta) \end{bmatrix}, \tag{11}$$

points can be converted into a camera centered coordinate frame by

$$\begin{bmatrix} X(\mathbf{x}_i, \mathbf{q}) \\ Y(\mathbf{x}_i, \mathbf{q}) \end{bmatrix} = \text{Rot}_{\theta,\beta}^{-1}(\mathbf{q} - \mathbf{c}_i) = \text{Rot}_{\theta,\beta}^{T} \begin{bmatrix} q_x - c_x \\ q_y - c_y \end{bmatrix} \tag{12}$$

where $X(\mathbf{x}_i, \mathbf{q})$ and $Y(\mathbf{x}_i, \mathbf{q})$ are points in camera coordinate. For simplicity we will drop the arguments $\mathbf{x}_i$, $\mathbf{q}$, and α from functions X, Y, p from now on.

The number of vertical pixels is the fraction of the image occupied by the target height times the number of horizontal pixels

$$\mathcal{N}_y = \frac{\mathrm{H}}{2X \tan\left(\frac{\gamma_y}{2}\right)} \mathrm{N_y}, \tag{13}$$

where γ_y is the vertical field of view of the camera. Similarly, the horizontal number of pixels is obtained by the fraction of the image occupied by the target length times the number of horizontal pixels

$$\mathcal{N}_x = \frac{\mathrm{L}}{2X \tan\left(\frac{\gamma_x}{2}\right)} \mathrm{N_x}. \tag{14}$$

where γ_x is the horizontal field of view of the camera. The horizontal and vertical fields of view are obtained by

$$\tan\left(\frac{\gamma_x}{2}\right) = \frac{l_x}{2f} \qquad \tan\left(\frac{\gamma_y}{2}\right) = \frac{l_y}{2f}. \tag{15}$$

The pixel area occupied by a target at depth X is given by the product of the functions $\mathcal{N}_x$ and $\mathcal{N}_y$ as in

$$\mathcal{N}_t = \frac{\mathrm{N_x N_y} LH}{l_x l_y} \frac{f^2}{X^2} = \mathcal{K} \frac{f^2}{X^2}, \tag{16}$$

where $\mathcal{K}$ is a constant that depends on the camera parameters and the target parameters L and H.

We can now define the boundaries of the sensor. The problem definition stated that targets can only be detected when their full height is in the field of view. The minimum depth X that a target can be in camera coordinates is

$$X \tan\left(\frac{\gamma_y}{2}\right) > (H - h) \therefore X_{min} = 2f \frac{(H-h)}{l_y}. \tag{17}$$

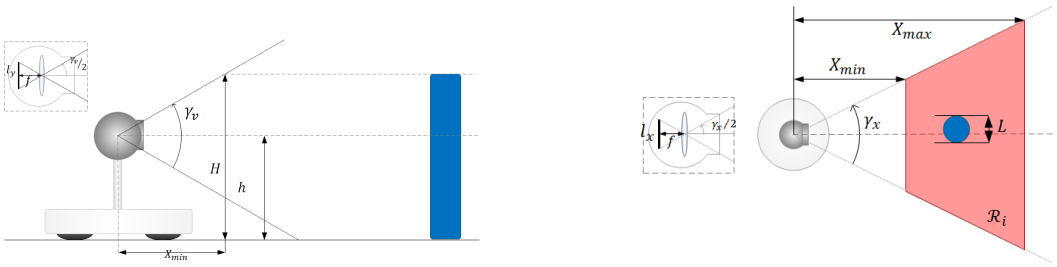
Also, no detection is possible for targets with less then a certain pixel area. Thus the maximum depth in camera coordinates is

$$\mathcal{K} \frac{f^2}{X^2} > \mathcal{N}_{min} \therefore X_{max} = f \sqrt{\frac{\mathcal{K}}{\mathcal{N}_{min}}}. \tag{18}$$

The angle of the field of view restricts points Y as in

$$\arctan\left(\frac{Y}{X}\right) \leq \left|\frac{\gamma_x}{2}\right| \therefore |Y| \leq \frac{l_x}{2f} |X|. \tag{19}$$

The region of coverage $\mathcal{R}_i$ is bounded by the field of view of the camera, the minimum point where the target can be



(a) Side view (b) Top view

Fig. 3. This figure shows the coverage scenario proposed in the case study. In Fig. 3a, a side view of the robot observing a target with height H is shown. The gray dashed line box shows a side cut of the camera sensor and how the focal distance f affects the angle of the vertical field of view γ_y. Also depicted the height of the robot h and the minimum depth for detection X_{min}. In Fig. 3b the top view of the robot observing a symmetric target with length L is shown. The gray dashed line box shows a transversal cut of the camera sensor to show how the focal distance f affects the angle of the horizontal field of view γ_x. Also shown here the minimum and maximum depth for detection denoted by X_{min} and X_{max}, respectively. The coverage region $\mathcal{R}_i$ is displayed as the red region.

seen in its full height and the minimum pixel area where a target can still be detected.

$$\mathcal{R}_i = \left\{ \mathbf{q} \,|\, X_{min} \leq X \leq X_{max} \bigwedge |Y| < \frac{l_x}{2f} |X| \right\} \quad (20)$$

B. Sensor Model

We model the detection probability as a function of pixel area that a target would occupy at a given depth X 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. The probability of detection according to the pixel area is given by

$$p_{\mathcal{N}_t} = \exp\left(-\frac{(\mathcal{N}_t - \mathcal{N}_\mu)^2}{2\sigma_{\mathcal{N}_t}^2} \right), \quad (21)$$

where $\mathcal{N}_\mu$ is the optimal pixel area for a detection and $\sigma_{\mathcal{N}_t}$ is a constant to determine how spread out the detection distribution $p_{\mathcal{N}_t}$ is with respect to the depth X.

Since 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 $\theta + \beta$ and the orientation α. It is also modeled as an exponential in

$$p_\alpha = \exp\left(-\frac{\text{dist}_g(\alpha, \theta + \beta)^2}{2\sigma_\alpha^2} \right), \quad (22)$$

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

$$p_i = \begin{cases} p_0 p_{\mathcal{N}_t} p_\alpha & \text{if } \mathbf{q} \in V_i \\ 0 & \text{otherwise} \end{cases} \quad (23)$$

where p_0 is the maximum detection probability by the sensor.

1) Inner Gradient: The gradient in the region V_i is given by the partial derivative of p_i with respect to x_i. Here we use the notation g' as the partial derivative of g with respect to x_i. The partial derivative of p_i with respect to x_i is

$$p_i' = \left[-\frac{(\mathcal{N}_t - \mathcal{N}_\mu)}{\sigma_{\mathcal{N}_t}^2} \frac{2f\mathcal{K}}{X^2} \left(f' - \frac{fX'}{X} \right) + - \frac{\text{dist}_g(\alpha, \theta + \beta)}{\sigma_\alpha^2} (\theta' + \beta') \right] p_i. \quad (24)$$

Applying this equation to the first term of (6) we have

$$\mathcal{G}_{\text{inner}} = \int_\Theta \int_{V_i} p_i' \prod_{j \in \mathcal{B}_i} [1 - p_j] \phi dq d\alpha$$

2) Coverage Boundary: The variation of the coverage boundary for parameters c_x, c_y, θ and β can be found by implicitly differentiating Equation (12) w.r.t. these parameters (we omit this for space constraints). To obtain the partial derivative of X and Y w.r.t f we note that points on the boundary of $\mathcal{R}_i$ are projected on the image sensor as

$$\frac{l_y}{2} = f\frac{h}{X} \qquad \frac{l_x}{2} = f\frac{Y}{X}. \quad (25)$$

Implicitly differentiating both equations (note that l_x, l_y and h do not vary w.r.t f), we get

$$\frac{\partial}{\partial f} \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} \frac{X}{f} \\ 0 \end{bmatrix}.$$

Differentiating Equation (12) we have

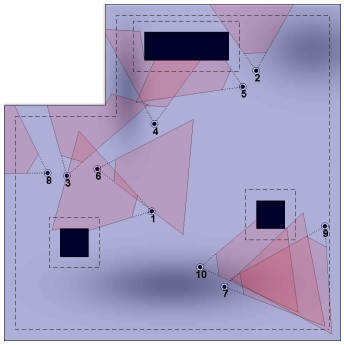
$$\frac{\partial}{\partial f} \begin{bmatrix} X \\ Y \end{bmatrix} = \frac{\partial}{\partial f} \left(\text{Rot}_{\theta_i, \beta_i}^T (\mathbf{q} - \mathbf{c}_i) \right) \therefore \frac{\partial \mathbf{q}}{\partial f} = \text{Rot}_{\theta_i, \beta_i} \begin{bmatrix} \frac{X}{f} \\ 0 \end{bmatrix}$$

Therefore, points in the boundary vary with respect to x_i as

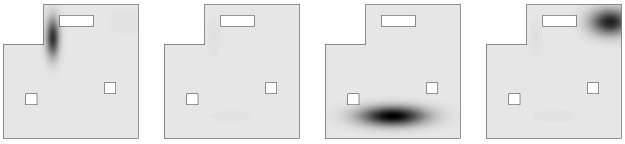
$$\frac{\partial \mathbf{q}_{rk}}{\partial \mathbf{x}_i} = \begin{bmatrix} 1 & 0 & -q_y + c_{iy} & -q_y + c_{iy} & \cos(\theta_i + \beta_i)\frac{X}{f} \\ 0 & 1 & q_x - c_{ix} & q_x - c_{ix} & \sin(\theta_i + \beta_i)\frac{X}{f} \end{bmatrix}.$$

For points on the coverage boundary we have

$$\mathcal{G}_{\text{cov}} = \sum_{r_k \in \text{Dscn}_{\text{cov}}(p_i)} \int_\Theta \int_{r_k} p_i \prod_{j \in \mathcal{B}_i} [1 - p_j] \frac{\partial \mathbf{q}_{r_k}}{\partial \mathbf{x}_i}^T \mathbf{n}_k \phi dq d\alpha$$



(a) Initial Configuration



(b) $\alpha = -\pi$ (c) $\alpha = -\pi/2$ (d) $\alpha = 0$ (e) $\alpha = \pi/2$

Fig. 4. Initial configuration of the robots over the region for the simulation is shown in Fig. 4a. Darker areas in the environment are the interest regions, here shown independent to the orientation α. In Fig. 4a, the orientation $\alpha = 0$ corresponds to a direct observation from the left to right. In Fig. 4b through 4e the anisotropic function ϕ used in the simulation is shown for different values of α. Darker areas indicate higher values of ϕ when a point is observed at orientation α. White indicates $\phi(q, \alpha) = 0$ and black $\phi(q, \alpha) = 1$.As an example, Fig. 4b shows higher values of ϕ on the upper left corner when observed with an absolute orientation of $-\pi$. That same area does not present very high values when observed at orientation $\alpha = -\pi/2$, as seen on Fig. 4c.

3) Visibility Boundary: As discussed before, obstacles and the environment introduce visibility boundaries. In this example this boundary varies only with respect to the position of the robot $\mathbf{c}_i$. This boundary is induced by a vertex v_j that can either belong to an obstacle H_j or to the boundary Q. The formula of this gradient can be found in [12].

$$\frac{\partial \mathbf{q}_{r_k}}{\partial \mathbf{x}_i} = \begin{bmatrix} -\frac{\|\mathbf{q}-\mathbf{v}_j\|_2}{\|\mathbf{c}_i-\mathbf{v}_j\|_2} & 0 & 0 & 0 & 0 \\ 0 & -\frac{\|\mathbf{q}-\mathbf{v}_j\|_2}{\|\mathbf{c}_i-\mathbf{v}_j\|_2} & 0 & 0 & 0 \end{bmatrix}$$

For points on a visibility boundary, we have

$$\mathcal{G}_{\text{vis}} = \sum_{r_k \in \text{Dscn}_{\text{vis}}(p_i)} \int_\Theta \int_{r_k} p_i \prod_{j \in \mathcal{B}_i} [1 - p_j] \frac{\partial \mathbf{q}_{r_k}}{\partial \mathbf{x}_i}^T \mathbf{n}_k \phi dq d\alpha$$

4) Controller: After calculating all the terms as in Equation (6), we obtain the gradient

$$\frac{\partial \mathcal{H}(\mathbf{x})}{\partial \mathbf{x}_i} = \mathcal{G}_{\text{inner}} + \mathcal{G}_{\text{cov}} + \mathcal{G}_{\text{vis}}, \tag{26}$$

and we can update the control rule by following the control rule in Equation (6).

Remark 2: In this example, the pan parameter β and the robot heading parameter θ have identical terms in the calculation, as their variation has the same effect over the coverage region. However, one might assign higher gains to the pan parameter, as it is likely to be less power intensive.

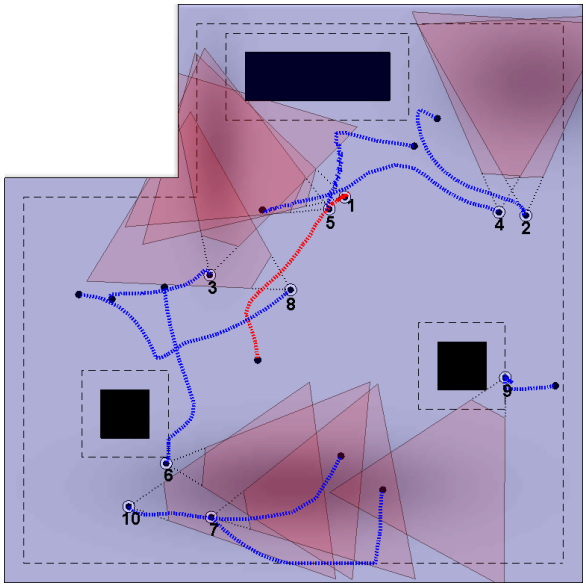


Fig. 5. Final configuration of a simulation of ten robots covering a nonconvex region with obstacles. The traversable region T is denoted by the dashed lines and the obstacles are black polygons. Darker areas in the environment are the interest regions, here shown independent to the orientation α. Robots start at an arbitrary configuration in Fig. 4a and change their position, heading, pan, and focal length parameter until reaching the configuration shown here. Trajectories are shown by dotted lines. As an example, the trajectory of robot 1 is shown in red, the remaining are blue. Red regions represent the visible area of coverage of each robot V_i. Robot 9 cannot advance through the obstacle and is stuck in that position. Also notice that the coverage area of robots 4 and 2 are closely aligned to $\alpha = \pi/2$, which is the orientation that provides the best detection performance for the given anisotropic density function ϕ (see Fig. 4e).

VI. Simulation Results

We conducted numerical simulations with our algorithm in nonconvex environments with obstacles. We show here one run of the algorithm. The anisotropic density function used is shown in Fig. 4 where there are three distinct interest regions to be covered by the robots. These interest regions have a preferred orientation of observation, intuitively meaning that they have a higher reward when observed at a given orientation. In this simulation we also included a traversable region T that prevents collision with the boundaries.

The environment length and width are 60m. The traversable region T is provided to the robots so that they are always at least two meters away from the obstacles and the boundary of the environment. The parameters used in this simulation were target height $H = 1.8$m and length $L = 0.5$m. The camera center was assumed to be at a height $h = 0.5$m with an image sensor with parameters $\text{N}_\text{x} = 640$ pixels, $\text{N}_\text{y} = 480$ pixels, $l_x = 3.04$mm, and $l_y = 1.98$mm. The optimal parameters for a target detection was set as $\mathcal{N}_\mu = 3840$ pixel^2, $\sigma_{\mathcal{N}_t} = 2800$ pixel^2 and a minimum pixel area of $\mathcal{N}_{min} = 1000$ pixel^2. For the anisotropic term, the parameter $\sigma_\alpha = \pi/3$ rad was assigned to the sensor. The maximum detection probability for each robot at any given moment was limited to $p_0 = 0.2$, in order to favor an area being observed by multiple robots. The control gains for the position was 1, for the rotation

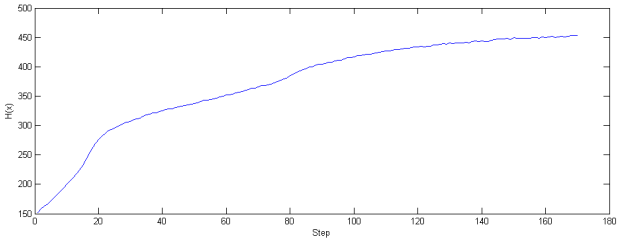


Fig. 6. The evolution of the objective function (Equation (5)) over time for this simulation. Network converged to a stable configuration after 170 steps.

and pan were 0.01 and 0.03, respectively, and the focal length gain was 10^{-9}. The gain for the focal length was set to a small value because small variations of f cause great variations in the field of view. Robots, whose field of view intersect, exchange information about their parameters and the geometry of their regions of coverage so that the detection performance over the overlapping regions so that the joint probability of detection can be computed for every point where there is an intersection. The simulation results are shown in Figures 5 and 6.

VII. CONCLUSION AND FUTURE WORK

In this paper, we introduced a gradient-ascent coverage control algorithm for maximizing the joint probability of detection of events over a planar nonconvex environment with polygonal obstacles. Our approach captures the property of preferred sensing orientations of certain types of events. For this, we defined an anisotropic density function that rewards observations according to the orientation in which an observation is being made. We characterized the discontinuities that should be taken into consideration when obtaining the gradient such as sensing and visibility boundaries. Furthermore, we relaxed a common assumption made in coverage algorithms, namely, the detection performance of a sensor decays inversely with the distance from the robot. To illustrate our technique, we presented an example application scenario, where a network of holonomic robots with pan and zoom capabilities maximized the joint probability of detection of a target with known geometry and dimensions.

As most gradient-ascent approaches, the final result depends strongly on the initial configuration of the nodes, as nodes can prematurely converge to local minima of the objective function. This problem can be partially mitigated, since the robots know the global event density function. The knowledge of the distribution can be used as a high level guide for the robots when they are in regions where their performance metric is poor. Presence of obstacles can also make motion along the gradient infeasible. Here, one can use path planning techniques in conjunction with gradient descent to avoid obstacles (like [17]). We plan to investigate these extensions in the future. Finding the configuration that gives the global maximum is an open problem in non-trivial environments.

We are currently studying improved distributed optimization techniques for the objective function presented, as the gradient-based approach shown in this paper has slow convergence in some cases. Moreover, there are application where there is no previous knowledge of the anisotropic density function and the environment map. Thus, acquiring and sharing this information online would be a major improvement to the technique presented in this paper.

ACKNOWLEDGMENTS

This work was partially supported by AFOSR MURI grant FA95500810356 and by ONR grant N000140910680. Thanks to Howie Choset for useful feedback.

REFERENCES

[1] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*, ser. Applied Mathematics Series. Princeton University Press, 2009.

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

[3] 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*, December 2008, pp. 4263–4268.

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

[5] 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, April 2004.

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

[7] 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, p. 12.

[8] 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.

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

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

[11] 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*, May 2009, pp. 3515 –3522.

[12] 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

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

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

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

[16] N. Chakraborty, S. Akella, and J. C. Trinkle, "Complementarity-based dynamic simulation for kinodynamic motion planning," in *IEEE/RSJ International Conference on Robots and Systems*, St. Louis, MO, Oct. 2009, pp. 787–794.

[17] 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*, May 2010, pp. 4982–4989.