Proceedings of the
47th IEEE Conference on Decision and Control
Cancun, Mexico, Dec. 9-11, 2008

Sensing and Coverage for a Network of Heterogeneous Robots

Luciano C. A. Pimenta, Vijay Kumar, Renato C. Mesquita, Guilherme A. S. Pereira

***Abstract*— We address the problem of covering an environment with robots equipped with sensors. The robots are heterogeneous in that the sensor footprints are different. Our work uses the location optimization framework in [1], [2], with three significant extensions. First, we consider robots with different sensor footprints, allowing, for example, aerial and ground vehicles to collaborate. We allow for finite size robots which enables implementation on real robotic systems. Lastly, we extend the previous work allowing for deployment in non convex environments.**

I. Introduction

A distributed and asynchronous approach for optimal coverage of a domain with identical mobile sensing agents is proposed in [2] based on a framework for optimized quantization derived in [1]. Each agent (robot) follows a control law, which is a gradient descent algorithm that minimizes a functional encoding the quality of the sensing coverage. Further, this control law depends only on the information of position of the robot and of its immediate neighbors. Neighbors are defined to be those robots that are located in neighboring Voronoi cells. Besides, these control laws are computed without the requirement of global synchronization. The functional also uses a *distribution density function* which weights points or areas in the environment that are more important than others. Thus it is possible to specify areas where a higher density of agents is required. Furthermore, this technique is adaptive due to its ability to address changing environments, tasks, and network topology.

Different extensions of the framework devised in [2] have been proposed in the literature. In [3] the problem of limited-range interaction between agents was addressed. The problem of learning the distribution density function online while moving toward the optimal locations was addressed in [4]. In [5] the basic approach was extended to deal with agents with limited energy. In this case, generalized Voronoi diagrams such as *power diagrams* [6] are employed. In the present work we propose three important extensions. First, we address the problem of incorporating heterogeneity in the robot team by allowing the robots to have different types of sensors. This first extension is actually a minor contribution since we use power diagrams, similarly to [5], with a different motivation. Second, we overcome the practical limitations of the point robot assumption in the original algorithm. Finally, we generalize the basic method to nonconvex environments. To the best of our knowledge, the last two extensions are not similar to any other extension found in the literature.

In the next section we present the main aspects of the basic method in [2] derived from the *Locational Optimization Framework* [7], using a distance function that is independent of the Euclidean metric.

L. C. A. Pimenta and V. Kumar are with the GRASP Laboratory, University of Pennsylvania, Philadelphia, PA 19104, USA. pimenta@seas.upenn.edu, kumar@grasp.upenn.edu

L. C. A. Pimenta, R. C. Mesquita, and G. A. S. Pereira are with the Department of Electrical Engineering, Universidade Federal de Minas Gerais, Belo Horizonte, MG, 31270-901, Brazil. {lucpim, renato, gpereira}@cpdee.ufmg.br

II. Locational Optimization Framework

Let $\Omega \subset \mathbb{R}^N$ be a given representation of the environment, $P = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$ be the configuration of n mobile sensors, where $\mathbf{p}_i \subset \Omega$, and $W = \{W_1, \ldots, W_n\}$ be a tessellation of Ω such that $I(W_i) \cap I(W_j) = \emptyset$, $\forall i \neq j$, where $I(\cdot)$ represents the interior of a given region, and $\cup_{i=1}^{n} W_i = \Omega$. The key idea is that each agent i is responsible for the coverage of the region W_i. As a measure of the system performance we define the *coverage functional*:

$$\mathcal{H}(P, W) = \sum_{i=1}^{n} \mathcal{H}(\mathbf{p}_i, W_i) = \sum_{i=1}^{n} \int_{W_i} f(d(\mathbf{q}, \mathbf{p}_i))\phi(\mathbf{q}) d\mathbf{q}\,, \tag{1}$$

where d corresponds to a function that measures distances between locations in Ω and sensors. Note that we do not require that this function defines a metric in Ω. The function $\phi : \Omega \to \mathbb{R}_+$ is a distribution density function which defines a weight for each point in Ω. The density function may reflect a knowledge of the probability of occurrence of events in different regions, or simply a measure of relative importance of different regions in Ω. Therefore, points with greater weight values should be better covered than points with smaller values. The function $f : \mathbb{R} \to \mathbb{R}$ is a smooth strictly increasing function over the range of d that measures the degradation of sensing performance with distance. We assume that $\mathcal{H}(P, W)$ is differentiable. The problem of covering the environment, Ω, is then translated to the problem of minimizing the functional in (1).

A. Centroidal Voronoi Tessellation

An important tool in the Locational Optimization theory is the *Voronoi tessellation*. Given the set of points $P = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$, often called *sites*, distributed over the bounded domain Ω, with boundary $\partial\Omega$, we define the Voronoi region, or Voronoi cell, V_i, associated to the point $\mathbf{p}_i$ according to a given distance function d as:

$$V_i = \{\mathbf{q} \in \Omega | d(\mathbf{q}, \mathbf{p}_i) \leq d(\mathbf{q}, \mathbf{p}_j), \forall j \neq i\}. \tag{2}$$

The definition in (2) is in fact a generalization of the ordinary definition of Voronoi regions based on the Euclidean

distance [7]. The generalized Voronoi tessellation of the set P, $V(P)$, is the collection of such regions. The Voronoi boundary ∂V_i is defined as:

$$\partial V_i = \cup_{j=1}^{n} l_{ij} \cup \{\partial\Omega \cap V_i\}. \tag{3}$$

where l_{ij} is the bisector:

$$l_{ij} = \{\mathbf{q} \in \Omega | d(\mathbf{q}, \mathbf{p}_i) = d(\mathbf{q}, \mathbf{p}_j), j \neq i\}. \tag{4}$$

Given a robot i we define the neighborhood of i, $\mathcal{N}_i$, as the set of robots that share Voronoi boundaries with V_i:

$$\mathcal{N}_i = \{j \in P | \partial V_i \cap \partial V_j \neq \emptyset\}. \tag{5}$$

Assuming that Ω is a convex polytope and d is the Euclidean distance, the boundaries are hyperplanes and the Voronoi cells are convex. In this case, two neighbor agents i and j are associated with cells that share a hyperplane, and this hyperplane intersects the segment $\overline{\mathbf{p}_i\mathbf{p}_j}$ at its midpoint, and perpendicular to the segment. For an extensive treatment of Voronoi tessellations we refer to [7]. The next two propositions relate Voronoi tessellations with the minimization of the objective functional in (1) for a general distance function, d. The proofs follow the same arguments, used by [8] and [4] in the case of the Euclidean distance.

Proposition 1 *A necessary condition for a minimizer of the objective functional in (1) is that the W tessellation corresponds to the Voronoi tessellation, $V(P)$, according to the distance function d.*

Proof: Let $\hat{V}$ be another tessellation than the Voronoi V. For a given point $\mathbf{q} \in V_i$, and $\mathbf{q} \in \hat{V}_j$, where $\mathbf{p}_k \in \hat{V}_j \Leftrightarrow k = j$, we can write:

$$f(d(\mathbf{q}, \mathbf{p}_i))\phi(\mathbf{q}) \leq f(d(\mathbf{q}, \mathbf{p}_j))\phi(\mathbf{q}). \tag{6}$$

Since $\hat{V}$ is not a Voronoi tessellation, the inequality in (6) will hold strictly over some measurable set of Ω. Therefore:

$$\mathcal{H}(P, V) < \mathcal{H}(P, \hat{V}).$$

■

Assuming that W is determined by the Voronoi tessellation of the points in P then $\mathcal{H}(P, W) = \mathcal{H}(P, V(P)) = \mathcal{H}(P)$ and we have the following result

Proposition 2 *A necessary condition for $\mathcal{H}(P)$ to be minimized is:*

$$\frac{\partial \mathcal{H}(P)}{\partial \mathbf{p}_i} = \frac{\partial \mathcal{H}(\mathbf{p}_i, V_i)}{\partial \mathbf{p}_i} = \int_{V_i} \frac{\partial}{\partial \mathbf{p}_i} f(d(\mathbf{q}, \mathbf{p}_i))\phi(\mathbf{q}) d\mathbf{q} = 0. \tag{7}$$

Proof: By applying the differentiation under the integral sign (see [9]) we can write

$$\begin{aligned}\frac{\partial \mathcal{H}}{\partial x_i} = &\int_{\partial V_i} f(d(\mathbf{q}, \mathbf{p}_{i_0}))\phi(\mathbf{q}) \frac{\partial(\partial V_i)}{\partial x_i} \cdot \mathbf{n}_i ds \\ &+ \int_{V_{i_0}} \frac{\partial}{\partial x_i} f(d(\mathbf{q}, \mathbf{p}_i))\phi(\mathbf{q}) d\mathbf{q} \\ &+ \sum_{j \in \mathcal{N}_i} \int_{\partial V_j} f(d(\mathbf{q}, \mathbf{p}_j))\phi(\mathbf{q}) \frac{\partial(\partial V_j)}{\partial x_i} \cdot \mathbf{n}_j ds\,,\end{aligned}$$

where $\mathbf{p}_i = [x_i, y_i]^T$ in two dimensions, $\mathbf{p}_{i_0}$ is a fixed configuration of agent i, V_{i_0} is the Voronoi region associated to $\mathbf{p}_i = \mathbf{p}_{i_0}$, $\mathbf{n}_i$ and $\mathbf{n}_j$ are the outward facing unit normals of ∂V_i and ∂V_j respectively, and ds is the element of arc length. Due to the property in (4), at the bisector we have $\partial V_i = \partial V_j$ and $d(\mathbf{q}, \mathbf{p}_{i_0}) = d(\mathbf{q}, \mathbf{p}_j)$. Also $\mathbf{n}_j = -\mathbf{n}_i$, and since $\partial V_i = \{\cup_{j \in \mathcal{N}_i}(\partial V_i \cap \partial V_j)\} \cup \{\partial\Omega \cap V_i\}$, and $\frac{\partial(\partial V_j)}{\partial x_i} = 0$ at $\partial\Omega \cap V_i$, we have

$$\frac{\partial(\partial V_i)}{\partial x_i} \cdot \mathbf{n}_i = -\sum_{j \in \mathcal{N}_i} \frac{\partial(\partial V_j)}{\partial x_i} \cdot \mathbf{n}_j.$$

Therefore, we conclude that

$$\frac{\partial \mathcal{H}}{\partial \mathbf{p}_i} = \int_{V_i} \frac{\partial}{\partial \mathbf{p}_i} f(d(\mathbf{q}, \mathbf{p}_i))\phi(\mathbf{q}) d\mathbf{q}\,, \tag{8}$$

which must be equal to zero at a minimum point. ■

In [2] the Euclidean distance is used as d, and $f(d) = d^2$. Moreover, since it is assumed a convex environment it is easy to prove that all Voronoi cells are convex polytopes. In this case the necessary configuration to be at a minimum is obtained when each agent is located exactly at the *centroid* of its own Voronoi cell. The centroid is given by:

$$\mathbf{p}_i^* = \frac{\int_{V_i} \mathbf{q}\phi(\mathbf{q}) d\mathbf{q}}{\int_{V_i} \phi(\mathbf{q}) d\mathbf{q}}. \tag{9}$$

Similarly, we can define a *generalized centroid* for general f and d functions, as follows:

$$\mathbf{p}_i^* = \min_{\mathbf{p}_i \in V_i} \int_{V_i} f(d(\mathbf{q}, \mathbf{p}_i))\phi(\mathbf{q}) d\mathbf{q}. \tag{10}$$

According to Propositions 1 and 2, every robot must be driven to the generalized centroid of its Voronoi region to minimize the functional (1). The resulting partition of the environment is commonly called *Centroidal Voronoi Tessellation* (CVT).

B. Continuous-Time Lloyd Algorithm

A classic discrete-time method to compute CVT's is the Lloyd's algorithm [1]. In each iteration this method executes three steps: (i) compute the Voronoi regions; (ii) compute the centroids; (iii) move each point site to the corresponding centroid.

In [2] a continuous-time version of this approach is proposed for kinematic models:$\dot{\mathbf{p}}_i = \mathbf{u}_i$. The following control law guarantees that the system converges to a CVT:

$$\mathbf{u}_i = -k(\mathbf{p}_i - \mathbf{p}_i^*)\,, \tag{11}$$

where k is a positive gain. The control law is a gradient-descent approach, since if d is the Euclidean distance and $f(d) = d^2$, $\frac{\partial \mathcal{H}}{\partial \mathbf{p}_i} = 2\left(\int_{V_i} \phi(\mathbf{q}) d\mathbf{q}\right)(\mathbf{p}_i - \mathbf{p}_i^*)$. It is important to mention that $\mathcal{H}$ is nonconvex which implies that the system will in general converge to a CVT that corresponds to a local minimum. In the rest of the paper we present possible extensions of the method proposed in [2].

III. Heterogeneous Robots

In this section we consider the problem of deploying a team of agents with heterogeneous sensing capabilities to cover an environment. We capture this heterogeneity by modelling each robot (sensor) as a circle (in $\mathbb{R}^2$) $B_i(\mathbf{p}_i, R_{\mathbf{p}_i})$, where $\mathbf{p}_i$ is the center position and $R_{\mathbf{p}_i}$ is the radius. This models omnidirectional sensors that may have limited range and where the quality of the measured information is only acceptable in the region B_i. Our robots are heterogeneous in that the sensor footprints are different. Although the points inside B_i have acceptable quality, this does not necessarily mean that the quality is uniform. The performance may degrade with the distance from the center of B_i. Mathematically, we describe the team task as the minimization of the functional:

$$\mathcal{H}(P, PV) = \sum_{i=1}^{n} \int_{PV_i} [\|\mathbf{q} - \mathbf{p}_i\|^2 - R_{\mathbf{p}_i}^2]\phi(\mathbf{q})d\mathbf{q}\,, \quad (12)$$

where $f(d) = d$ and $d(\mathbf{q}, \mathbf{p}_i) = \|\mathbf{q} - \mathbf{p}_i\|^2 - R_{\mathbf{p}_i}^2$ is the so-called *power distance* [6]. According to Proposition 1 the required tessellation must then be a Voronoi partition according to the power distance. The resulting tessellation is well-known in the literature and it is often called the *Voronoi Diagram in the Laguerre geometry* [10] or the *power diagram* [6].

The power diagram, PV, associates a power region, PV_i, with each circle $(\mathbf{p}_i, R_{\mathbf{p}_i})$ in $\mathbb{R}^2$ defined by:

$$PV_i = \{\mathbf{q} \in \mathbb{R}^2 | d(\mathbf{q}, \mathbf{p}_i) \leq d(\mathbf{q}, \mathbf{p}_j), \forall j \neq i\}\,, \quad (13)$$

where the *power distance* $d(\mathbf{q}, \mathbf{p}_i) = \|\mathbf{q} - \mathbf{p}_i\|^2 - R_{\mathbf{p}_i}^2$.

The power diagram can be viewed as a generalized Voronoi diagram which is closely related to the original Voronoi diagram. Some of its properties are:

Property 1 *The bisector between neighbor power cells PV_i and PV_j is a hyperplane perpendicular to the segment that connects the centers of the circles, B_i and B_j. If the two circles intersect, the bisector passes through the points of intersection. The bisector is defined by the equation:*

$$(\mathbf{p}_i - \mathbf{p}_j)^T \mathbf{q} = \frac{1}{2}(\|\mathbf{p}_i\|^2 - \|\mathbf{p}_j\|^2 - R_{\mathbf{p}_i}^2 + R_{\mathbf{p}_j}^2). \quad (14)$$

Property 2 *Each power cell PV_i is convex or empty. A necessary condition for PV_i to be empty is that the center $\mathbf{p}_i$ is contained in the union of the other circles.*

Property 3 *If a circle, B_i, is not intersected by any other circle, then B_i is entirely contained in PV_i.*

Property 4 *If all circles B_i are identical, $PV_i = V_i$.*

Figure 1 presents three different cases for the power regions for two circles. Figure 1(c) presents the situation where an agent is not located inside its own power region. Thus it is also possible that a robot could have an empty power region. In Figure 1(b), if we have a third robot the same size of the smallest robot inside the power region of the large robot, this third robot contained in the power region of the largest robot would have an empty power region.

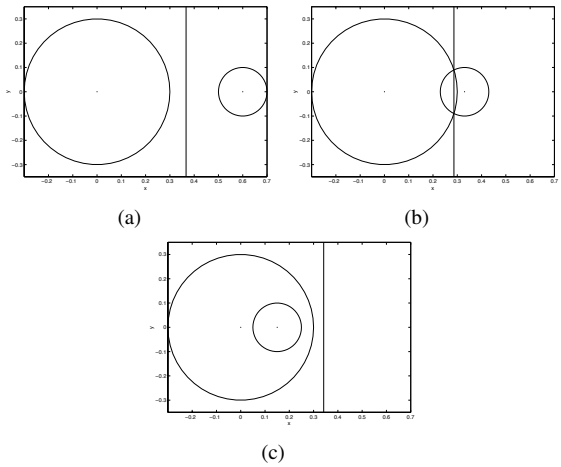


Fig. 1. Power cells. Fig. 1(a) Two disjoint circles. Fig. 1(b) Intersecting circles with both centers inside their corresponding power cells. Fig. 1(c) Intersecting circles with one center outside its corresponding power cell.

The next proposition presents the gradient that will be used by the distributed control laws.

Proposition 3 *The gradient of $\mathcal{H}(P, PV)$ in (12) is given by:*

$$\frac{\partial \mathcal{H}}{\partial \mathbf{p}_i} = 2(\mathbf{p}_i - \mathbf{p}_i^*) \int_{PV_i} \phi(\mathbf{q})d\mathbf{q}\,, \quad (15)$$

where $\mathbf{p}_i^*$ is the centroid of PV_i.

Proof: The proof follows directly from Proposition 2. ■

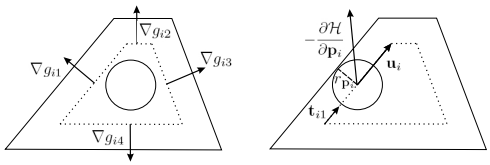
By observing the last proposition we conclude that we can use the same control law in (11). However, one must be careful of the special properties of power diagrams.

As stated in Property 2, a robot that is located in the union of other sensors footprint may have an empty power region. This is not surprising. Since we are designing strategies for heterogeneous teams of robots that act only on local information, robots that have better sensors than others have priority during the deployment. Thus robots with the worst sensors may get trapped in configurations in which they do not contribute to the overall mission. Of course, it is possible to let the "trapped" robot perturb the system by executing a deterministic controller to a region outside the circle in which it is trapped. Practically, we don't find empty power regions in our simulations.

IV. Robots with Finite Size

A practical problem of the unconstrained minimization executed by the pure gradient-descent law in (11) is that actual robots are not point-robots. In this section, we extend the basic results to robots that can be modelled as circular disks, each one with radius $r_{\mathbf{p}_i}$. Also, as in [2], we first assume d is the Euclidean distance and $f(d) = d^2$. Let $\mathcal{F}_{V_i}$ be the *free Voronoi region* defined by the set of points:

$$\mathcal{F}_{V_i} = \{\mathbf{q} \in V_i | \|\mathbf{q} - \mathbf{q}_{\partial V_i}\| \geq r_{\mathbf{p}_i}, \forall \mathbf{q}_{\partial V_i}\}\,, \quad (16)$$



(a) (b)

Fig. 2. Linear constraints for a disk-robot with radius $r_{\mathbf{p}_i}$. The dotted lines represent the facets of $\partial\mathcal{F}_{V_i}$, which are given by $g_{i1}=0$, $g_{i2}=0$, $g_{i3}=0$, and $g_{i4}=0$, with gradients as in Fig. 2(a). Given the active set $\mathcal{A}=\{g_{i1}\}$ in Fig. 2(b) we compute the control input $\mathbf{u}_i$ by projecting the negative gradient of $\mathcal{H}$ onto the unit vector $\mathbf{t}_{i1}$ which is tangent to the active facet.

where $\|\cdot\|$ is the Euclidean norm and $\mathbf{q}_{\partial V_i}$ is a point at the boundary of the Voronoi region, ∂V_i. In fact, the boundaries of the free Voronoi regions, $\partial\mathcal{F}_{V_i}$, are hyperplanes parallel to the hyperplanes that define the boundaries of V_i, located at a distance $r_{\mathbf{p}_i}$ from ∂V_i. It is straight forward to check that such free region is convex.

If the robots start from a safe configuration, a sufficient condition to guarantee collision avoidance is that each robot disk lies in the interior of its own Voronoi region. Therefore, we define a *constrained, location optimization* problem as follows. If robot i is constrained to remain inside its own Voronoi region which is bounded by m facets, there are m linear constraints on the position of its center. Accordingly, we find the robot positions:

$$\begin{aligned}&\min_{\mathbf{p}_i}\mathcal{H}(P,V)\\ &\text{s.t.}\\ &g_{i1}(\mathbf{p}_i)\leq 0\,,\ldots,g_{im}(\mathbf{p}_i)\leq 0\end{aligned}\tag{17}$$

where $g_{il}(\mathbf{p}_i)=0$ defines the lth facet.

Accordingly we choose the control law given by:

$$\mathbf{u}_i=\beta_i\mathbf{h}_i\,,\tag{18}$$

where $\beta_i\propto\left\|\frac{\partial\mathcal{H}}{\partial\mathbf{p}_i}\right\|$ and the vector $\mathbf{h}_i$ is determined by:

$$\begin{aligned}&\min_{\mathbf{h}_i}\left(\frac{\partial\mathcal{H}}{\partial\mathbf{p}_i}\right)^T\mathbf{h}_i\\ &\text{s.t.}\\ &\|\mathbf{h}_i\|=1\,,\nabla g_{ip}^T\mathbf{h}_i\leq 0\,,\ldots,\nabla g_{it}^T\mathbf{h}_i\leq 0\end{aligned}\tag{19}$$

If $\mathcal{A}=\{g_{ip},\ldots,g_{it}\}$ is the set of active constraints $g_{ip}=\ldots=g_{it}=0$, we can compute the controls in (18) as follows:

1) If $\mathcal{A}$ is empty then

$$\mathbf{u}_i=-k\frac{\partial\mathcal{H}}{\partial\mathbf{p}_i}\,,\tag{20}$$

2) Otherwise

$$\mathbf{u}_i=k\pi\left(-\frac{\partial\mathcal{H}}{\partial\mathbf{p}_i},\partial\mathcal{F}_{V_i}\right)\,,\tag{21}$$

where $\pi\left(-\frac{\partial\mathcal{H}}{\partial\mathbf{p}_i},\partial\mathcal{F}_{V_i}\right)$ gives the projection of vector $-\frac{\partial\mathcal{H}}{\partial\mathbf{p}_i}$ along the vector $\mathbf{t}_{il}$ which is the unit vector tangent to the lth facet (see Fig. 2). This tangent vector provides a feasible direction and is such that $-\frac{\partial\mathcal{H}}{\partial\mathbf{p}_i}^T\mathbf{t}_{il}>0$.

Since the region $\mathcal{F}_{V_i}$ is convex and $\partial\mathcal{F}_{V_i}$ is given by plane faces the projection π is guaranteed to return a feasible vector if one exists. Equilibrium is obtained when the first-order Karush-Kuhn-Tucker (KKT) conditions are satisfied.

We can use the algorithm presented in this section with heterogeneous mobile sensors by using (15) and the power diagram, PV, to compute $\frac{\partial\mathcal{H}}{\partial\mathbf{p}_i}$ but (18, 19) computed using the facets from the Voronoi diagram, V.

V. Nonconvex Environments

In [11], nonconvex environments are addressed in the problem of deploying a team of robots to achieve full visibility of the environment. Differently, in the present work we consider the following problem in a nonconvex environment: *minimize the coverage functional (1), such that d is the geodesic distance and $f(d)=d^2$*:

$$\mathcal{H}(P,W)=\sum_{i=1}^{n}\mathcal{H}(\mathbf{p}_i,W_i)=\sum_{i=1}^{n}\int_{W_i}d(\mathbf{q},\mathbf{p}_i)^2\phi(\mathbf{q})d\mathbf{q}.\tag{22}$$

Let Ω be a compact region in $\mathbb{R}^2$ with boundary, $\partial\Omega$, determined by a simple polygon with m sides and set of vertices $V=\{\mathbf{v}_1,\ldots,\mathbf{v}_m\}$. Also, assume that $P=\{\mathbf{p}_1,\ldots,\mathbf{p}_n\}\subset\Omega$. By the geodesic distance, $d(\mathbf{o},\mathbf{w})$, between two points $\mathbf{o}$ and $\mathbf{w}$, we mean the length of the shortest path, $s(\mathbf{o},\mathbf{w})$, between $\mathbf{o}$ and $\mathbf{w}$, entirely contained in Ω. In fact, it is well known that such a path is formed by the sequence of segments $\{\overline{\mathbf{o}\mathbf{v}_{r1}},\overline{\mathbf{v}_{r1}\mathbf{v}_{r2}},\ldots,\overline{\mathbf{v}_{rl-1}\mathbf{v}_{rl}},\overline{\mathbf{v}_{rl}\mathbf{w}}\}$, where $\mathbf{v}_{ri}$'s are reflex vertices of $\partial\Omega$. By reflex vertices, we mean the vertices with internal angle greater than 180 degrees. According to Proposition 1 we require a Voronoi tessellation computed according to the geodesic metric. Such a geodesic Voronoi tessellation can be computed by means of the algorithm proposed in [12]. Some properties of the geodesic distance in simple polygons are [12]:

Property 5 *As Ω is closed and bounded by a simple polygon, $s(\mathbf{o},\mathbf{w})$ exists and is unique for every pair of points in Ω. Moreover, $s(\mathbf{o},\mathbf{w})$ is piecewise linear with "breakpoints" at reflex vertices of $\partial\Omega$.*

Property 6 *The function $d(\mathbf{o},\mathbf{w})$ is continuous in both $\mathbf{o}$ and also $\mathbf{w}$. Furthermore, this function is continuously differentiable except at a set of measure zero Γ.*

The following proposition allows for devising distributed control laws.

Proposition 4 *If $\mathbf{w}\in\Omega\backslash\Gamma$ is a point where the gradient $\nabla_{\mathbf{w}}d(\mathbf{w},\mathbf{q})$ exists, then this gradient is given by:*

$$\frac{\partial d}{\partial\mathbf{w}}=-\mathbf{z}_{\mathbf{w},\mathbf{q}}\,,\tag{23}$$

where $\mathbf{z}_{\mathbf{w},\mathbf{q}}$ is a unit vector directed along the first segment of the shortest path $s(\mathbf{w},\mathbf{q})$.

Proof: Let the shortest path, $s(\mathbf{w},\mathbf{q})$, between the points $\mathbf{w}$ and $\mathbf{q}$, be given by the sequence of segments $\{\overline{\mathbf{w}\mathbf{v}_{r1}}, \overline{\mathbf{v}_{r1}\mathbf{v}_{r2}}, \ldots, \overline{\mathbf{v}_{rl-1}\mathbf{v}_{rl}}, \overline{\mathbf{v}_{rl}\mathbf{q}}\}$. The path length is then determined by: $d(\mathbf{w},\mathbf{q}) = \|\mathbf{v}_{r1}-\mathbf{w}\| + \|\mathbf{v}_{r2}-\mathbf{v}_{r1}\| + \ldots + \|\mathbf{v}_{rl}-\mathbf{v}_{rl-1}\| + \|\mathbf{q}-\mathbf{v}_{rl}\|$, where $\mathbf{v}_{ri}$'s are reflex vertices of the polygon $\partial\Omega$. Therefore,

$$\frac{\partial d}{\partial \mathbf{w}} = -\frac{(\mathbf{v}_{r1}-\mathbf{w})}{\|\mathbf{v}_{r1}-\mathbf{w}\|}. \tag{24}$$

■

From this proposition and from (8), a distributed control law that minimizes $\mathcal{H}$ in (22) can be computed by:

$$\mathbf{u}_i = -k\frac{\partial \mathcal{H}}{\partial \mathbf{p}_i} = 2k\int_{V_i} d(\mathbf{p}_i,\mathbf{q})\phi(\mathbf{q})\mathbf{z}_{\mathbf{p_i},\mathbf{q}}d\mathbf{q}. \tag{25}$$

It is important to mention that at the points where we have discontinuities in the gradient, which are also reflex vertices of the polygon $\partial\Omega$, we must use generalized gradients [13].

Definition 1 *(Clarke's Generalized Gradient): For a locally Lipschitz function $g : \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}$ define the generalized gradient of g at $(\mathbf{x},t)$ by*

$$\partial g(\mathbf{x},t) = \overline{co}\{\lim \nabla g(\mathbf{x}_i,t_i) | (\mathbf{x}_i,t_i) \to (\mathbf{x},t)\}, \tag{26}$$

where $\overline{co}$ is the convex closure, $(\mathbf{x}_i,t_i) \notin \Gamma$ which is the set of measure zero where the gradient is not defined.

Assume that agent $\mathbf{p}_i$ is located exactly at a reflex vertex $\mathbf{v}_{ri}$. Let $\partial_F\mathcal{H}(\mathbf{p}_i,V_i) \subset \partial\mathcal{H}(\mathbf{p}_i,V_i)$ be the subset of generalized gradients, ξ, at the reflex vertex, $\mathbf{v}_{ri}$, that have feasible directions, *i. e.*, $\exists\delta \in (0,\delta_{max}]$ such that $\mathbf{v}_{ri} - \delta\xi \in \Omega$. We can use the following control law at the reflex vertices:

$$\mathbf{u}_i = -k\xi, \tag{27}$$

where k is a positive gain, and ξ could be any vector belonging to $\partial_F\mathcal{H}(\mathbf{p}_i,V_i)$.

VI. Simulation Results

In this section we present simulations to verify the three extensions to the work in [2]. First we show in Figure 3 the result of applying the algorithm described in Section III to deal with heterogeneous, point robots. We show a team of nine agents with three possible footprints. Two agents have a large circular footprint of radius 0.08 units, one agent has a medium-size circular footprint of radius 0.05 units, and the remaining robots have the smallest circular footprint of radius 0.02 units. In the left figures we present configurations of the group and in the right we present trajectories that show how the robots evolved from the configuration shown in the panel immediately to the left to the next configuration in the panel below. The trajectories in Figure 3(h) end at the optimal configuration. The corresponding power diagram is also presented. The density function is a smooth cubic spline centered at the point $\mathbf{q}_c = [0.5,0.5]^T$:

$$\phi(\mathbf{q}-\mathbf{q}_c,h) = \begin{cases} 1-\frac{3}{2}\kappa^2+\frac{3}{4}\kappa^3 & \text{if} \quad 0\le\kappa\le 1, \\ \frac{1}{4}(2-\kappa)^3 & \text{if} \quad 1\le\kappa\le 2, \\ 0 & \text{otherwise}, \end{cases} \tag{28}$$

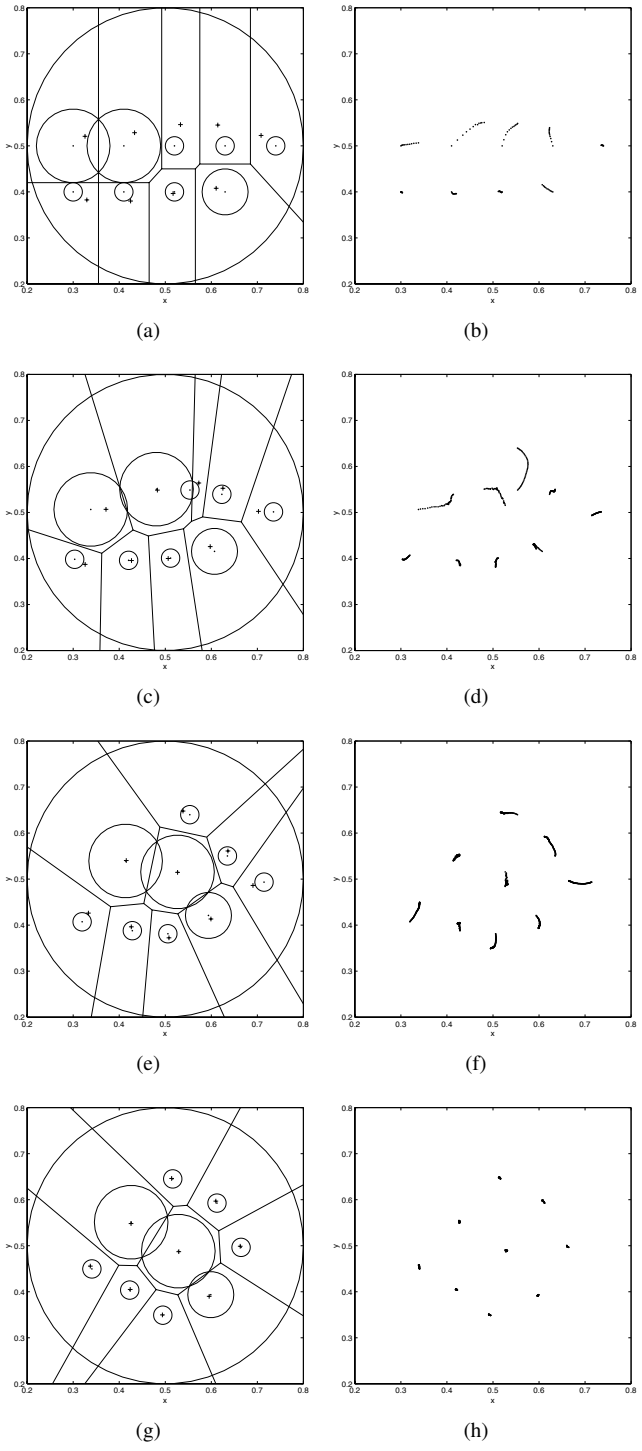
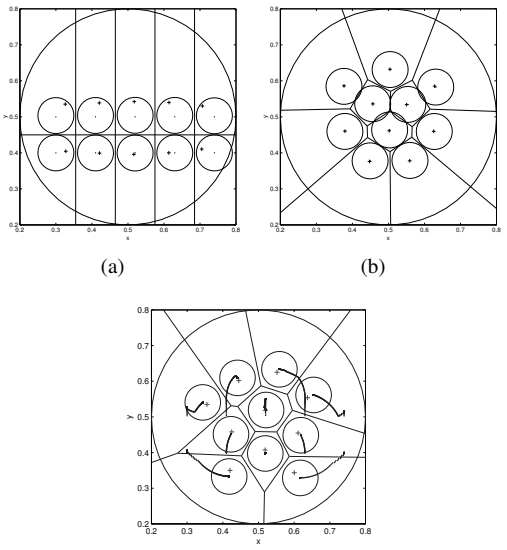


Fig. 3. Simulation results for heterogeneous agents in a environment with density function defined as in (28). In the panels on the left, the large circle with diameter 0.6 units represents the density function support. A team of nine robots, starting from an initial configuration (Fig. 3(a)), with intermediate configurations (Figs. 3(c) and 3(e)), converge to a centroidal formation (Fig. 3(g)). Figs. 3(b), 3(d), 3(f), and 3(h) correspond to the trajectories followed by the robots starting from the configuration in the figure presented in the left. The crosses represent the centroids of the corresponding power regions. Small circles associated with robots represent the footprints of the sensors carried by the robots.



(a) (b) (c)

Fig. 4. Simulation results for finite size agents in an environment with density function defined as in (28). A team of ten finite size agents start from an initial configuration (Fig. 4(a)) and have final configuration that implies collisions as in Fig. 4(b) when executing the unconstrained minimization as in [2]. By using the technique proposed in Section IV the agents reach a final configuration (Fig. 4(c)) without colliding (see trajectories). The crosses represent the centroids of the corresponding Voronoi regions. The largest circle represents the density function support while the other circles represent the shape of the robots.

where $\kappa = \|\mathbf{q} - \mathbf{q}_c\|/h$. It can be observed that the function support is determined by $2h$. In the simulation $h = 0.15$.

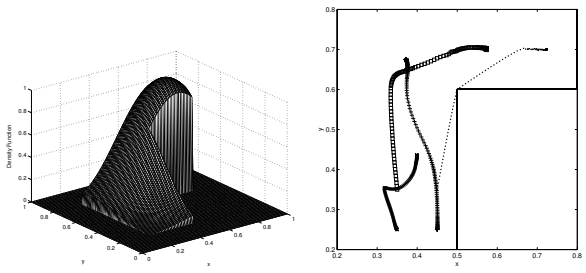
We can verify that the final configuration is obtained when a centroidal tessellation is obtained, as expected. It is interesting to note that the best sensors converged to the area where better coverage is required, *i. e.*, the area where we find the higher values of the density function.

Figure 4 shows the result obtained when solving the constrained minimization proposed in Section IV for a team of ten disk-shaped robots. The density function is the same as before. In this example, collisions are observed when the unconstrained control law (11) is used (see Figure 4(b)). However, the trajectories shown in Figure 4(c) which are obtained by using the control law (18) are free of collisions. Note that in this case not all agents converge to their centroids (see Figure 4(c)). This is the price the robots have to pay to guarantee safety (no collisions).

The extension to address nonconvex environments is verified in Figure 5. A group of four agents cover a L-shape environment according to the density function in Figure 5(a). The peak of the function is located at the point $[0.7, 0.7]^T$. This density function is a cubic spline inside the nonconvex domain and zero outside. One can conclude from Figure 5(b) that the robots follow trajectories that do not leave the nonconvex domain as desired. As expected, the robots moved to the region where better coverage is required.

VII. Conclusions

We addressed the problem of deriving optimal distributed control laws to cover nonconvex domains with a team of



(a) (b)

Fig. 5. Simulation results for a L-shape environment. Given a density function defined in a nonconvex domain (Fig. 5(a)), a group of four robots executes the approach proposed in Section V and moves according to the trajectories in Fig. 5(b).

heterogeneous mobile sensors with finite sizes. This paper incorporates three novel extensions into the previous work [1], [2] to address: (a) sensors with circular footprints of different radii, (b) disk-shaped robots, and (c) nonconvex polygonal environments. The extensions are based on the use of different distance functions, power distance and geodesic distance, and the incorporation of constraints to allow collision avoidance. Extensions to spheres (in $\mathbb{R}^3$) are straight forward.

VIII. Acknowledgments

We gratefully acknowledge partial support from NSF grants IIS-0427313, NSF IIP-0742304, and IIS-0413138, ARO grant W911NF-05-1-0219, ONR grant N00014-07-1-0829, and CNPq – Brazil.

References

[1] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inform. Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[2] J. Cortez, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Trans. Robot. and Automat.*, vol. 20, no. 2, pp. 243–255, 2004.

[3] J. Cortez, S. Martinez, and F. Bullo, "Spatially-distributed coverage optimization and control with limited-range interactions," *ESIAM: Control, Optimisation and Calculus of Variations*, vol. 11, pp. 691–719, 2005.

[4] M. Schwager, J. McLurkin, and D. Rus, "Distributed coverage control with sensory feedback for networked robots," in *Proceedings of Robotics: Science and Systems*, Philadelphia, USA, 2006, pp. 1–8.

[5] A. Kwok and S. Martínez, "Deployment algorithms for a power constrained mobile sensor network," in *Proc. IEEE Int. Conf. Robot. Automat.*, Pasadena, USA, 2008, pp. 140–145.

[6] F. Aurenhammer, "Power diagrams: Properties, algorithms and applications," *SIAM Journal on Computing*, vol. 16, no. 1, pp. 78–96, 1987.

[7] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, 2nd ed., ser. Wiley Series in Probability and Statistics. New york: Wiley, 2000.

[8] Q. Du, V. Faber, and M. Gunzburger, "Centroidal Voronoi tessellations: Applications and algorithms," *SIAM Review*, vol. 41, no. 4, pp. 637–676, December 1999.

[9] H. Flanders, "Differentiation under the integral sign," *American Mathematical Monthly*, vol. 80, no. 6, pp. 615–627, 1973.

[10] H. Imai, M. Iri, and K. Murota, "Voronoi diagram in the Laguerre geometry and its applications," *SIAM Journal on Computing*, vol. 14, no. 1, pp. 93–105, 1985.

[11] A. Ganguli, J. Cortez, and F. Bullo, "Distributed deployment of asynchronous guards in art galleries," in *Proc. of the American Control Conference*, Minneapolis, USA, 2006, pp. 1416–1421.

[12] B. Aronov, "On the geodesic Voronoi diagram of point sites in a simple polygon," *Algorithmica*, vol. 4, no. 1, pp. 109–140, 1989.

[13] F. Clarke, *Optimization and Nonsmooth Analysis*. New York: Wiley, 1983.