Voronoi–based Coverage Optimization for Mobile Networks with Limited Sensing Range — A Directional Search Approach

John Stergiopoulos and Anthony Tzes

Abstract— Motion coordination for coverage optimization purposes in mobile networks with limited sensory range is the scope of this paper. The coordination planning is based on Voronoi tessellations by taking into consideration the sensory radius of each agent. Optimization is performed in order to increase the total area covered by the network. An online directional–search algorithm is proposed in order to achieve locally optimal coverage. An agent moves inside its region of responsibility in a way that the total area surveyed by the network increases. The online control action makes the network adaptive to possible changes in the environment. Results are compared to schemes that base their action on Centroidal Voronoi Tessellations (CVT).

Index Terms— Voronoi diagrams, coverage optimization, directional search

I. INTRODUCTION

The provision of an adequate surveillance of a geographic region with a feasible cost dictates the formation of an integrated system with optimum geodesic topology as far as the sensor deployment [1]. Furthermore, the mobile abilities of these sensors can offer adaptation due to changes in the surveyed environment (as in search–and–rescue missions) or alteration of the network [2–4].

Many researchers have studied the problem of keeping the network connected all the time in order to ensure data transfer from one point to another [1, 5]. The problem is examined from a communication point of view based on graph theory. It is profound that such control policies do not achieve optimum area coverage and this is the trade–off to be balanced. In other scenarios studied, the network is supposed to be static and the usage of mobile agents for its re–connection is examined in cases where some nodes break down and the latter is separated in clusters [6]. Coverage control of agents with limited anisotropic sensory has been studied in [7].

Since the coordination of a whole network is cumbersome from a computational point of view especially when the number of nodes is too large, decentralized control techniques based on partitioning of the area of interest into smaller ones (known as Voronoi diagrams [8]) have been studied [9, 10]. In such cases each node is responsible for its area of responsibility rather than the entire space, trying to

The authors are with the Electrical and Computer Engineering Department, University of Patras, Rio 26500, Achaia, GREECE. Corresponding author's e–mail address: tzes@ece.upatras.gr

optimize a criterion [9, 11]. Furthermore, an interesting field for research is to control the transmission power of each of the nodes in the network for reconfiguration purposes by taking into account the lifetime of each of the nodes. Optimum positioning scheme when the nodes' range can vary is studied in [12].

The innovation of this work lays in the fact that the optimization criterion is the total area covered by the network, rather than some symmetry criterion [9]. This problem has been examined from an instantaneous coverage point of view [10], where the agents coordinate their motion such that all points in the environment are surveyed *through time* by an equal amount. Inexistence of analytical expression for the covered area by a set of agents results in inability to find continuous–time control laws based on gradients. The algorithm proposed in this paper surpasses this problem and guarantees (locally) optimal coverage ratio.

The article is organized as follows. In section II the coverage problem from a group of agents is presented and the contribution of Voronoi partitioning of the space in the first is examined. In section III the classical offline optimization problem and the proposed algorithm for the online version are discussed and compared, while in section IV motion coordination strategies based on directional–search optimization are analyzed. Simulation results in section V are presented that show the efficacy of each of the proposed algorithms. Concluding remarks are provided on the last section.

II. PROBLEM FORMULATION

A. Coverage problem setup

Let Ω the region under surveillance be a convex closed set in $\mathbb{R}^2$. Suppose that n is the number of available mobile agents responsible for the sensing coverage of Ω. Let us denote the set $I_K = \{i \in \mathbb{N} : i \leq K\}$ for any $K \in \mathbb{N}$. The agents are considered to move on the $\mathbb{R}^2$ Euclidean configuration space and their positions are denoted as $\mathbf{x}_i \in \mathbb{R}^2$, $i \in I_n$. The following assumption are made for the agents:

Assumption **1.** The agents are supposed to move on the $\mathbb{R}^2$ plane through two control inputs each, while obeying the discrete evolutionary equation (equivalently to [10]):

$$\mathbf{x}_i(k+1) = \mathbf{x}_i(k) + \mathbf{u}_i(k), \ \mathbf{u}_i, \mathbf{x}_i \in \mathbb{R}^2, \ i \in I_n, \qquad (1)$$

subject to saturation limits in the control inputs $\mathbf{u}_i$:

$$\|\mathbf{u}_i\|_\infty \leq \bar{v}, \ i \in I_n. \qquad (2)$$

Assumption **2.** Each agent is supposed to have a uniform circular sensing pattern centered at the agent's position $\mathbf{x}_i$

and is limited by a fixed maximum sensing radius r. The critical sensing radius is the same for all agents since the network is considered homogeneous. Let us denote as C_i the sensing region of each agent i, i.e.:

$$C_i = \left\{ \mathbf{x} \in \mathbb{R}^2 \colon \|\mathbf{x} - \mathbf{x}_i\| \leq r \right\}, \ i \in I_n. \tag{3}$$

The agents are initially deployed randomly in Ω. The goal is to find optimal positions such that the area of the covered region set of Ω is as high as possible. For a closed compact convex set $P \subset \mathbb{R}^2$ let ∂P be its boundary, i.e. the smallest polygon enclosing P. Then ∂P is fully defined by its vertices $p_j = (x_{p_j}, y_{p_j})$, $j \in I_{N_P}$, where N_P is the number of the vertices and x_{p_j}, y_{p_j} are the coordinates of vertex p_j in the base coordinate frame. Then the area–function $\mathscr{A}(\cdot)$ for any P is defined as [13]:

$$\mathscr{A}(P) = \frac{1}{2} \sum_{j=1}^{N_P} \left(x_{p_j} y_{p_{j+1}} - x_{p_{j+1}} y_{p_j} \right), \tag{4}$$

where $p_{N_P+1} \equiv p_1$ and the vertices p_j are set in counter–clockwise order.

The main objective is to position the agents at certain positions such that the total region of Ω surveyed by the network is maximized:

$$\max_{\mathbf{x}_i, \ i \in I_n} \mathscr{A} \left(\Omega \cap \bigcup_{i \in I_n} C_i \right). \tag{5}$$

B. Voronoi spatial tessellations

At each agent a responsibility region is assigned based on the spatial coordinates of the agents in the plane. The set of these regions is well–known as a Voronoi diagram [8]. For the convex closed set Ω and the n cites (agents), the region under surveillance is partitioned in n closed convex subsets V_i, $i \in I_n$, which are defined as:

$$V_i = \left\{ \mathbf{x} \in \Omega \colon \|\mathbf{x} - \mathbf{x_i}\| \leq \|\mathbf{x} - \mathbf{x_j}\|, \ \forall j \in I_n \right\}, \ i \in I_n. \tag{6}$$

The closed set V_i is known as the Voronoi cell of node/agent i. It should be noted that a Voronoi diagram is a full tessellation of $\Omega \subset \mathbb{R}^2$, since $\bigcup_{i \in I_n} V_i = \Omega$. A Voronoi cell is uniquely characterized by the set of its vertices. A point $\mathbf{x} \in \Omega$ can: *a)* either lay in the interior of a Voronoi cell, i.e. $\mathbf{x} \in \text{int}(V_i)$, and node i is responsible for its coverage, or *b)* lay on the boundary of a Voronoi cell, i.e. $\mathbf{x} \in \partial V_i$ and more than one nodes are responsible it.

In this paper the agents should be moved in a way to try and cover ideally the whole space Ω, considering their limited sensing capabilities. Thus, the Voronoi tessellation alone is not sufficient for the motion algorithms, but the sensing regions C_i, $i \in I_n$ should be taken into account. The R–limited Voronoi sets are defined as:

$$V_i^R = \{ \mathbf{x} \in \Omega \colon \mathbf{x} \in V_i \cap C_i \}, \ i \in I_n. \tag{7}$$

An important property of these sets is that, since Ω, V_i, C_i are convex sets $\forall i \in I_n$, then V_i^R are all convex sets, too. However, they do not always apart a full tessellation of Ω, since $\bigcup_{i \in I_n} V_i^R \subseteq \Omega$.

For each agent, the blind regions of its own Voronoi cell (the regions inside the Voronoi cell that do not lay in the sensing region) are defined by:

$$B_i = \{ \mathbf{x} \in \Omega \colon \mathbf{x} \in (V_i - C_i) \}, \ i \in I_n, \tag{8}$$

while the unexploited regions of an agent (parts of the sensing region of the node that do not contribute to coverage of V_i) are defined as:

$$U_i = \left\{ \mathbf{x} \in \mathbb{R}^2 \colon \mathbf{x} \in \left(C_i - V_i^R \right) \right\}, \ i \in I_n. \tag{9}$$

It should be noted that although $V_i, V_i^R, B_i \subseteq \Omega$, $\forall i \in I_n$ by definition, the same does not always hold for U_i. The aforementioned regions for one node are shown in Fig.1.
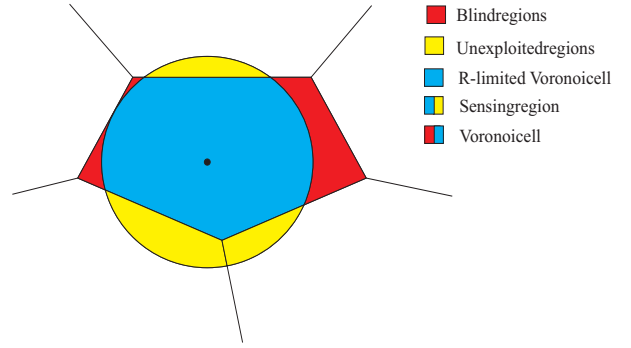

Fig. 1. Characterization of the different regions concerning an agent and its Voronoi cell

Consequently, the total region of Ω surveyed by the network S, can be written as:

$$S = \left\{ \mathbf{x} \in \mathbb{R}^2 \colon \mathbf{x} \in \Omega \cap \bigcup_{i \in I_n} C_i \right\} = \bigcup_{i \in I_n} V_i^R. \tag{10}$$

The main advantage in computation of S via (10) is that, since $V_i^R \cap V_j^R = \emptyset$, $i \neq j$, the area covered by the network can be computed as the summation of the areas of the independent R–limited Voronoi sets.

III. CONTROL POLICIES FOR COVERAGE OPTIMIZATION

A. Offline global optimization strategy

Suppose again $\Omega \subset \mathbb{R}^2$ the convex region to be surveyed by n nodes with limited sensory range as defined by (3). The objective is to find the planar coordinates of each node $\mathbf{x}_i \in \Omega$, $i \in I_n$ that optimize the coverage criterion (5). Since there is no closed form expression for $\mathscr{A}(S)$, a numerical approach must be followed.

This is a standard constrained numerical optimization problem which may converge to possible local extrema. The constraint emerges from the fact that all nodes should lay on the boundary or in the interior of Ω. Let $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n)$ be the vector of the coordinates of all the agents, $\mathbf{X} \in \mathbb{R}^{2n}$. The constraint $\mathbf{x}_i \in \Omega$, $\forall i \in I_n$ can be cast in a linear compact form as $A_{con}\mathbf{X} \leq B_{con}$, where the matrices A_{con}, B_{con} are explicitly defined by the vertices p_j of Ω [14]. Note that $A_{con} \in \mathbb{R}^{nN_\Omega} \times \mathbb{R}^{2n}$, $B_{con} \in \mathbb{R}^{nN_\Omega}$.

Thus, the offline version of the coverage problem can be defined as the solution of the following constrained optimization:

$$\max_{\mathbf{X}} \mathscr{A}(S), \qquad A_{con}\mathbf{X} \leq B_{con}, \qquad (11)$$

where S is given by (10). This is a computationally intensive optimization problem to solve due to the inexistence of an analytical expression for $\mathscr{A}(S)$, the large number of local extrema of it and the time it takes (for even a small number of nodes).

B. Online methods — convergence to local extrema

The most significant disadvantage in the offline version described in III-A is its lack of adaptation. Consequently, in case the region of interest changes or if an agent/node runs out of energy, then a new optimization should be performed for the "new" optimal positioning of the nodes, which may be disastrous in cases of emergency. Online algorithms are faster from a time–to–solve point of view and are adaptive since the optimization is performed online by the network itself. However, global optimal solution is not guaranteed, due to the large number of local extrema of the problem.

One of the main differences, compared to the offline method, is that we are not interested in defining only the final optimal positioning of the nodes, but also the whole path they follow (their positions at each time step), such that (if possible) the total coverage is an increasing function of time. What needs to be ensured is *i*) stability of the network's motion and *ii*) locally optimum coverage in the final state.

Accomplishment of the first objective can be ensured if *only one* node moves at each time step in a way that the total coverage increases. The selection of the node–to–move can be performed randomly among the set of them. The motion is performed into small maneuvers (ε–motions), such that the Voronoi partitioning of the space does not alter significantly.

As for the second objective, the selected agent that is to move at each time step first finds the direction at which it should move inside its own Voronoi cell, such that the area of the total region surveyed by the network will be increased (due to its motion). In case that motion of the node results in less coverage area, then another node is chosen, until one agent able to increase coverage is found. The procedure ends when all nodes' motions lead in less coverage than that in the previous time step or when the mean of the coverage contribution in the last M steps is insignificant, when M is a parameter chosen by the coordinator.

In the policy presented above it is profound that each agent should be able to compute its own Voronoi cell. It should be noted that since we are interested in the sensing coverage of a compact region Ω, none of the Voronoi sets is an open set. In order to be able to do this, an agent should known the positions of its Delaunay neighbors [8], which are all the agents that have a common Voronoi edge with that node. In this paper it is supposed that either a supervisor–coordinator informs periodically the agents about the position of their Delaunay neighbors or all nodes have communication capabilities with transmission power such

that they can exchange information with their Delaunay neighbors, at each time step.

IV. ANALYSIS OF ONLINE METHODS — A DIRECTIONAL SEARCH APPROACH

In this section the general online optimization strategy described in III-B, which takes into account the limited sensory range of the agents, is clarified step by step. The nodes are initially deployed in Ω. In order to prevent oscillatory phenomena in the network behavior, only one node is moved at a time. At this point, what is not clearly yet defined are: *i*) the direction at which a node is to move and *ii*) the exact motion of the selected agent to move at each time step. In this paper, two schemes are proposed concerning the aforementioned issues, which both base their action on directional search strategies.

According to the first, the selected agent that is to move at each time step performs an optimization for defining the direction at which it should move inside its own Voronoi cell. The criterion to maximize is the slope of the area of the total region surveyed by the network. Thus, it is guaranteed that not only the total area will be increased comparing to the previous time step (due to the motion of the node), but with the highest possible rate, too. This problem can easily be solved by scanning the node's heading perimetrically around itself.

According to the second algorithm, the selected agent at each time step does not perform any optimization at all, but move towards the direction of the centroid of its R–limited Voronoi. This strategy needs by far less computational effort compared to the first one; however, since its roots lay in CVT theory, it does not guarantee optimum coverage, but optimizes some symmetry criterion, instead. Comparing to other CVT algorithms though [12, 15], the one chosen in this article is the one that contributes most to coverage.

A. Steepest descent scheme

As mentioned above, what needs to be defined clearly is the appropriate direction towards which the node should move, along with the manner of its motion. According to this algorithm, the selection of the appropriate heading is made considering the one that leads in the highest area covered by the network. Thus, it results in a steepest descent optimization algorithm which is performed for one node at a time in order to define the direction at which it will move. It is profound that in case the node lays in a position which is a local maximum for its own coverage contribution, then the first is selected to stand still and another agent from the random list is chosen to fulfill the objective. When all nodes' motions lead in less coverage ratio than the current one, then their current positions are the optimal ones that one seeks.

The problem is that their is no analytical expression for the total area covered, and thus the gradients have to be computed in a discrete manner. Let j be the node that is to move. Discrete gradient implementation implies that the points at which the "new" coverage ratio will be computed,

should not be too far from the current node's position $\mathbf{x}_j(k)$ or equivalently:

$$\left\| \mathbf{x}_j(k) - \hat{\mathbf{x}}_j(k+1) \right\| \leq \varepsilon, \tag{12}$$

where $\hat{\mathbf{x}}_j(k+1)$ is the estimated node's position at the next time step and $0 < \varepsilon \leq \bar{v}$ is to be kept small, so that the current tessellation of Ω is not altered significantly (Note: the inequality $\varepsilon \leq \bar{v}$ emerges from (2)). The reason for using an estimated position is that during the optimization stage for determining the suitable heading, one does not know yet which direction will be selected for the node's motion. As a conclusion, the optimal heading computed with this method is given as:

$$\theta(k) = \arg\max \left\{ \sum_{i \in I_n} \mathscr{A}\left(\hat{V}_i^R(k+1)\right) - \sum_{i \in I_n} \mathscr{A}\left(V_i^R(k)\right) \right\}, \tag{13}$$

subject to:

$$\sum_{i \in I_n} \mathscr{A}\left(\hat{V}_i^R(k+1)\right) > \sum_{i \in I_n} \mathscr{A}\left(V_i^R(k)\right). \tag{14}$$

Constraints (14) have been inserted since we are interested in only increasing the coverage ratio performed by the network. It should be noted that the reason for computing the network coverage area as the summation of the areas of the independent R–limited Voronoi sets (instead of the the area of their union) is because the latter are disjoint by definition, i.e. $V_i^R \cap V_j^R = \emptyset$, $\forall (i,j) \in I_n \times I_n$, $i \neq j$. Of course, in case that there is no valid solution in (13)–(14), then another node is chosen as a candidate to move. It should be noted that the constrained optimization problem (13)–(14) is computationally tractable, since the search is performed for a single parameter $\theta(k)$.

Having computed the suitable heading $\theta(k)$ for node j, the latter has to move towards that direction. It is profound that the amount of distance the node moves should be that at which the prediction was made, considering (12) and (13). As a result, the control inputs of the agent are given by:

$$\mathbf{u}_j(k) = \begin{pmatrix} \varepsilon \cos\theta(k) \\ \varepsilon \sin\theta(k) \end{pmatrix}, \tag{15}$$

where constraints (2) obviously hold and $\mathbf{u}_i(k) = \mathbf{0}$, $i \in I_n$, $i \neq j$.

B. CVT–based scheme

Although the category of algorithms that base their action on centroids converge in solutions that optimize some "symmetry" criterion, they are presented in order to emphasize their contribution to coverage problems. The main difference compared to [12] is that a node is chosen to move *towards* that centroid iff the network's coverage is increased by that motion; otherwise another node is chosen to move.

The most common algorithm for quasi–symmetric spatial distribution of the nodes in a sensor network is to move each node to the centroid of its own Voronoi cell. However, this algorithm is not suitable in cases where the sensors have limited range, since it takes into account only the spatial characteristics of the network and thus is not suit for coverage purposes.

A more appropriate version of this algorithm is to move each node to the centroid of its own R–limited Voronoi cell. Taking into consideration that the sensing pattern of all nodes is symmetrical, it is profound that in the case when $C_i \subseteq V_i \Leftrightarrow C_i \equiv V_i^R$, the centroid of V_i^R coincides with the node itself, i.e. $\mathscr{C}\left(V_i^R\right) \equiv \mathbf{x}_i$.

For a compact convex closed set $P \subset \mathbb{R}^2$ let the smallest enclosing polygon ∂P consist of N_P vertices p_j, $j \in I_{N_P}$ and its area denoted as $\mathscr{A}(P)$. The planar coordinates of the centroid of this set $\mathscr{C}(P) = \left(x_{\mathscr{C}(P)}, y_{\mathscr{C}(P)}\right)$ are then given by [13]:

$$x_{\mathscr{C}(P)} = \frac{1}{6\mathscr{A}(P)} \sum_{j=1}^{N_P} (x_{p_j} + x_{p_{j+1}})(x_{p_j} y_{p_{j+1}} - x_{p_{j+1}} y_{p_j}), \tag{16}$$

$$y_{\mathscr{C}(P)} = \frac{1}{6\mathscr{A}(P)} \sum_{j=1}^{N_P} (y_{p_j} + y_{p_{j+1}})(x_{p_j} y_{p_{j+1}} - x_{p_{j+1}} y_{p_j}), \tag{17}$$

where $p_{N_P+1} \equiv p_1$, $\mathscr{A}(P)$ is defined in (4) and the vertices p_j are considered in counter–clockwise order.

In order to emphasize in the intuitive physical properties of an algorithm that coordinates the nodes' motion towards the centroid of their R–limited Voronoi cells, imagine that the sensing radius of an agent increases with a steady rate; there will be a time where ∂C_i will intersect ∂V_i and consequently $U_i \neq \emptyset$. Let the centroid of the unexploited regions be denoted as $\mathscr{C}(U_i)$. By definition of the area centroid, since $C_i = V_i^R \cup U_i$, $\mathscr{A}\left(V_i^R \cap U_i\right) = 0$, and $\mathscr{C}(C_i) = \mathbf{x}_i$, $\forall i \in I_n$, one can verify that:

$$\mathscr{C}\left(V_i^R\right)\mathscr{A}\left(V_i^R\right) + \mathscr{C}(U_i)\mathscr{A}(U_i) = \mathbf{x}_i\mathscr{A}(C_i). \tag{18}$$

Furthermore $\mathscr{A}(C_i) = \mathscr{A}(U_i) + \mathscr{A}\left(V_i^R\right)$ holds and thus the centroid of V_i^R can be computed by (18) as:

$$\mathscr{C}\left(V_i^R\right) = \mathbf{x}_i\left(\frac{\mathscr{A}(U_i) + \mathscr{A}\left(V_i^R\right)}{\mathscr{A}\left(V_i^R\right)}\right) - \mathscr{C}(U_i)\left(\frac{\mathscr{A}(U_i)}{\mathscr{A}\left(V_i^R\right)}\right) \tag{19}$$

Note that when $\mathscr{A}(U_i) = 0$, then $\mathscr{C}\left(V_i^R\right) \equiv \mathbf{x}_i$.

A important remark is that the unexploited regions of an agent's sensing region w.r.t its own Voronoi cell act as repulses when the agent is moving towards the centroid of its R–limited Voronoi cell, while the magnitude of the "artificial repulsive force" is proportional to the area of these unexploited regions. Thus one can conclude that this algorithm tends to maximize the "symmetry" of the unexploited regions around an agent itself.

As in IV-A, one should defined the appropriate direction towards which the node should move, along with the control inputs, by using the algorithm presented in this section. Let again $j \in I_n$ be the index of the agent that is selected (in a random manner) to move, and $\mathbf{x}_j(k)$ be its spatial coordinates at this time step. Let $\theta(k)$ be the appropriate heading, which in this case is given as:

$$\theta(k) = \text{angle}\left(\mathscr{C}\left(V_j^R(k)\right) - \mathbf{x}_j(k)\right), \tag{20}$$

where $\text{angle}(\cdot)$ is the heading function of the vector given as argument.

By taking into account the selected heading direction $\theta(k)$ one could select the control inputs via the following control law:

$$\mathbf{u}_j(k) = \left(\begin{array}{c} \min\left\{K\bar{\upsilon}, \left\|\mathbf{x}_j(k) - \mathscr{C}\left(V_j^R(k)\right)\right\|\right\}\cos\theta(k) \\ \min\left\{K\bar{\upsilon}, \left\|\mathbf{x}_j(k) - \mathscr{C}\left(V_j^R(k)\right)\right\|\right\}\sin\theta(k) \end{array} \right), \quad (21)$$

where $0 < K \leq 1$ is a factor to suppress high values of $\bar{\upsilon}$, that may lead to significant change in the Voronoi topology from one step to another. This alternative offers a special advantage over (15) in case where the centroid lays near the node, avoiding so any oscillatory undesired response in the node's positioning between consecutive steps of the algorithm.

Overall, the suggested algorithms fall along the concept of a single–step ahead predictive control laws. The convergence can certainly be improved if a multi–step ahead cost function is employed; however, the problem becomes computationally intractable due to the exploding number of constraints (14) and prediction–horizon related results $(\theta(k), \theta(k+1), \ldots)$.

V. SIMULATION RESULTS

Simulation studies were carried out in order to show the efficacy of the proposed schemes. The region Ω to be surveyed is a convex set in $\mathbb{R}^2$. During network evolution the covered area is always increasing until it converges to an extremum solution. The agents are initially deployed in Ω, while their kinematics are described by (1). The control inputs are determined online as explained in sections IV-A and IV-B, depending on the strategy used in each case, while their maximum values were selected as $\bar{\upsilon} = 5\frac{\text{cm}}{\text{sample}}$.

In the simulation studies following, the number of agents is $n = 12$ with critical sensing radius equal to $r = 2\text{m}$. The maximum theoretically achievable sensing area is $\sum_{i \in I_n} \mathscr{A}(C_i) = n\pi R^2 = 150.8\text{m}^2$, while the area of the region of interest is $\mathscr{A}(\Omega) = 226.37\text{m}^2$. The convex area under surveillance is that presented in [1, 12]. Ideally, if possible, the agents should be able to cover the region without any overlapping, and thus covering $\frac{\sum_{i \in I_n} \mathscr{A}(C_i)}{\mathscr{A}(\Omega)} = 66.6\%$ of Ω. In both studies, the main algorithm used is that described in section III-B. Each time step was considered a 10msec time interval. The algorithm stops when all nodes' motions lead in less coverage or when:

$$\sum_{\ell=k-M+1}^{k} \mathscr{A}(S(\ell)) - \mathscr{A}(S(\ell-1)) = \mathscr{A}(S(\ell)) - \mathscr{A}(S(\ell-M)) < \delta, \quad (22)$$

where S is defined in (10), M is chosen to 50 samples and δ is set equal to $\delta = 1\text{cm}^2$. In both scenarios the agents' initial configuration is that shown in Fig. 2.

The evolution of the network in time, its final configuration, along with the coverage percentage ratio w.r.t. time, when the control scheme presented in section IV-A (IV-B) was applied are shown in Fig. 3 (4). The straight line in the percentage covered area figures (c) represents the maximum possible coverage ratio, which in our case is 66.6%.

The value of ε was selected equal to $\bar{\upsilon}$, since the latter is not considered large enough. As far as concerns the final
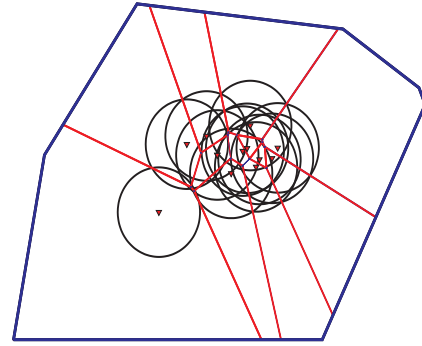

Fig. 2.   Initial state of the network for both simulation studies

nodes state, it is obvious that the agents have positioned themselves such that there is no overlapping between their sensing patterns, while attaining optimum coverage, for both cases. As far as concerns Fig. 3(c), the initial coverage percentage ratio is 21.57%, while in almost 10sec (1000 samples) the latter has reached the maximum possible coverage percentage ratio 66.6%. It is also obvious that the covered area by the network is a strictly increasing function of time, as expected, since as it has been mentioned, a node does not move unless its motion contributes to total coverage.

As for Fig. 4(c), in the first 20sec (2000 samples) of simulation the network has achieved coverage percentage ratio 65%, while maximum coverage percentage ratio is achieved after 40sec more. The covered area by the network is a strictly increasing function of time, as expected in this scenario, too. It should be noted that the maximum possible coverage ratio is the same for both simulations, since it only depends on the nodes' common sensing radius r and the number of nodes n.

Judging from the results obtained by simulation studies, one can see that optimum coverage is obtained faster when using the algorithm of section IV-A, compared to that when using that of IV-B. Indeed, this was absolutely expectable, since according to the first algorithm, the moving node drives towards the direction where maximum coverage will be achieved. Apart from that, when using the algorithm of section IV-B, due to the control law (21), when the network state is near–to–optimal, the nodes perform micro–maneuvers. Finally, comparing Fig. 4(b) and Fig. 3(b), one can see that usage of the control scheme in section IV-B tends to keep the network more cohesive.

Comparing this work to previous ones, it is seen that the network tries to optimize the area covered by itself, rather than some symmetry criterion [9, 12, 15]. The fact that only one node moves at a time parts the main difference, while proof of optimal area achievement is guaranteed by this way.

VI. CONCLUSIONS

In this paper control strategies are presented for optimizing the area covered by a homogeneous network consisted of mobile agents, where the coordination planning is based on Voronoi tessellations. Two directional–search online control
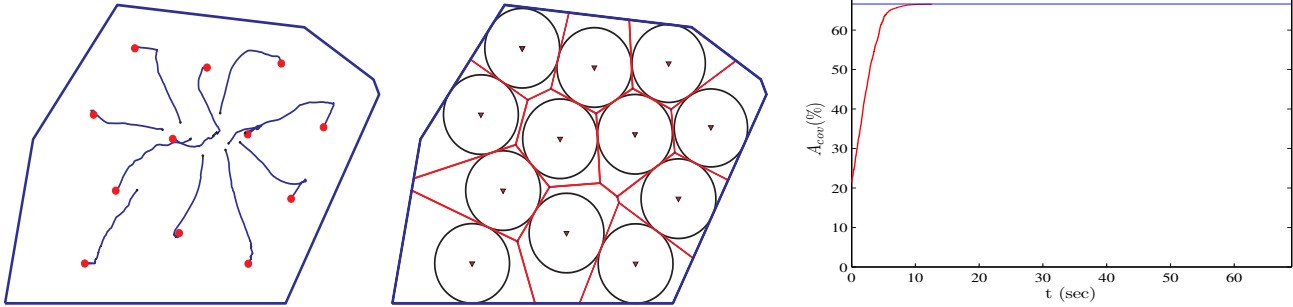
Fig. 3. Control scheme presented in section IV-A: (a) Evolution of the agents' positions (the dots represent final positions), (b) Final state of the network, (c) Percentage of covered area w.r.t. time
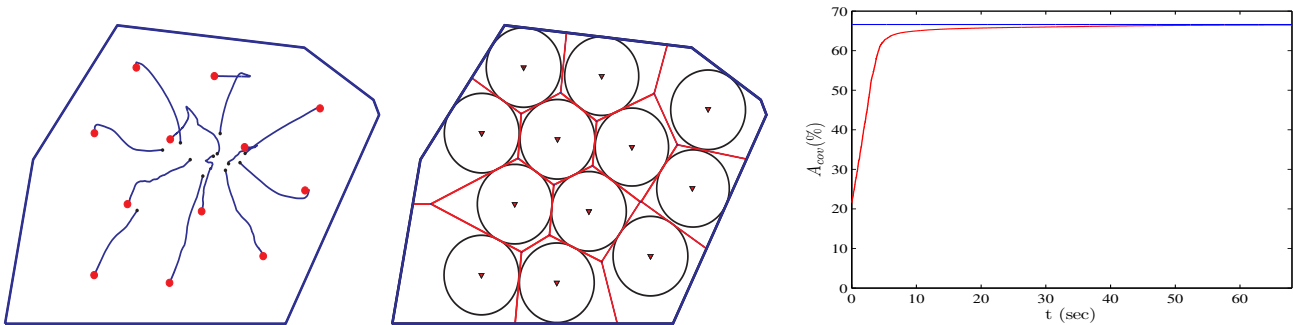

Fig. 4. Control scheme presented in section IV-B: (a) Evolution of the agents' positions (the dots represent final positions), (b) Final state of the network, (c) Percentage of covered area w.r.t. time

schemes were proposed in order to achieve optimum coverage. Due to the nature of the latter, the total area covered by the agents is a strictly increasing function of time. Online action makes the network flexible and reconfigurable to environmental changes. Considering heterogeneous networks or unequal surveillance of regions inside a node's pattern consist possible extension of this work.

REFERENCES

[1] J. Cortés, S. Martínez, and F. Bullo "Spatially–Distributed Coverage Optimization and Control with Limited–Range Interactions," in *ESAIM: Control, Optimisation and Calculus of Variations*, vol. 11, no. 4, 2005, pp. 691–719.

[2] A. Ahmadzadeh, A. Jadbabaie, V. Kumar, and J. Pappas, "Multi–UAV Cooperative Surveillance with Spatio–Temporal Specifications," in *Proc. of the 45th IEEE Conference on Decision and Control*, San Diego, CA, USA, 2006, pp. 5293–5298.

[3] N. E. Leonard, D. A. Paley, F. Lekien, R. Sepulchre, D. M. Fratantoni, and R. E. Davis "Collective Motion, Sensor Networks, and Ocean Sampling," in *Proc. of the IEEE*, vol. 95, no. 1, 2007, pp. 48–74.

[4] Y. Wang, and I. I. Hussein "Cooperative Vision–Based Multi–Vehicle Dynamic Coverage Control for Underwater Applications," in *Proc. of the 16th IEEE International Conference on Control Applications*, Singapore, 2007, pp. 82–87.

[5] X. Y. Li, P. J. Wan, and O. Frieder, "Coverage in Wireless Ad Hoc Sensor Networks," in *IEEE Transactions on Computers*, vol. 52, no. 6, 2003, pp. 753–763.

[6] A. Panousopoulou, and A. Tzes "Utilization of Mobile Agents for Voronoi–Based Heterogeneous Wireless Sensor Network Reconfiguration," in *Proc. of the 2007 European Control Conference*, Kos, Greece, 2007, pp. 4276–4284.

[7] K. Laventall, and J. Cortés "Coverage Control by Robotic Networks with Limited–Range Anisotropic Sensory" in *Proc. of American Control Conference*, Seattle, Washington, USA, 2008, pp. 2666–2671.

[8] F. Aurenhammer, and R. Klein "Voronoi Diagrams", Handbook of Computational Geometry, Chapter 5, Elsevier Science Publishing, 2000, pp. 201–290.

[9] S. Martínez, J. Cortés, and F. Bullo "Motion–Coordination with Distributed Information," in *IEEE Control Systems Magazine*, vol. 27, Aug. 2007, pp. 75–88.

[10] I. I. Hussein, and D. M. Stipanović "Effective Coverage Control for Mobile Sensor Networks With Guaranteed Collision Avoidance," *in IEEE Transactions on Control Systems Technology*, vol. 14, no. 4, 2007, pp. 642–657.

[11] R. M. Murray "Recent Research in Cooperative Control of Multivehicle Systems" *in ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, 2007, pp. 571–583.

[12] J. Cortés, and F. Bullo "Coordination and Geometric Optimization via Distributed Dynamical Systems," *in SIAM Journal on Control and Optimization*, vol. 44, no. 5, 2005, pp. 1543–1574.

[13] J. Cortés, S. Martínez, T. Karatas, and F. Bullo "Coverage Control for Mobile Sensing Networks" *in IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, 2004, pp. 243–255.

[14] S. Boyd, and L. Vandenberghe "Convex Optimization," *Cambridge University Press*, Chapter 2, 2004, pp. 21–66.

[15] Q. Du, V. Faber, and M. Gunzburger "Centroidal Voronoi Tessellations: Applications and Algorithms," *in SIAM Review*, vol. 41, no. 4, 1999, pp. 637–676.