Autonomous Robots (2020) 44:627–645
https://doi.org/10.1007/s10514-019-09859-y

Distributed coverage in mobile sensor networks without location information

Marzieh Varposhti[1] · Vesal Hakami[2] · Mehdi Dehghan[3]

Received: 20 April 2018 / Accepted: 7 May 2019 / Published online: 27 May 2019

Abstract
With the recent advances in robotic technologies, field coverage using mobile sensors is now possible, so that a small set of sensors can be mounted on mobile robots and move to desired areas. Compared to static settings, area coverage is more complicated in a mobile sensor network due to the dynamics arising from the continuous movement of the sensors. This complication is even higher in the more realistic case where little or no prior metric information is available about the sensor field. In this paper, we consider the problem of self-deployment of a set of mobile sensors which have no knowledge of the area, the number of nodes, their location, and even the distances to each other. In this restricted setting, we formulate the problem as a multi-player game in which each sensor tries to maximize its coverage while considering the overlapping sensing areas by its neighbors. We propose a distributed learning algorithm for coordinating the movement of the sensors in the field, and prove its convergence to the equilibria of the formulated game. Simulation results demonstrate that for moderate density deployments, the proposed algorithm competes with the existing location-dependent mobility strategies, while outperforming location-free algorithms.

Keywords Mobile sensor networks · Area coverage · Game theory · Self-regulated deployment · Distributed learning · Simplicial complexes

1 Introduction

Mobile sensor networks (MSNs) consist of sensor nodes that can move on their own and interact with the physical environment. Such networks enable a variety of applications in the context of surveillance, target tracking, search and rescue, and real-time monitoring of hazardous material (Yick et al. 2008). Critical to the performance of MSNs is an efficient sensor mobility strategy which is capable of coordinating the motions of the mobile sensors to achieve area coverage with desirable quality of surveillance. In recent years, many sensor mobility strategies have been proposed for area coverage in the context of MSNs (Luo and Chen 2012; Izadi et al. 2015; Yuan et al. 2014). In one taxonomy, these strategies can be categorized into two main groups: *location-based strategies*, and *location-free strategies*.

The location-based mobility strategies (Zou and Chakrabarty 2003; Wang et al. 2006; Fletcher et al. 2010b; Falcon et al. 2010; Zhu and Martínez 2013) consistently rely on the sensor's location information for effective deployment of the nodes in the field. We argue, however, that a location-dependent algorithm is not a cost-effective procedure in terms of real-time computations In particular, the exact location of the sensor nodes should be provided by either a localization algorithm or via a localization device (e.g., GPS). Using localization algorithms needs offline

This is one of the several papers published in *Autonomous Robots* comprising the Special Issue on Multi-Robot and Multi-Agent Systems.

✉ Marzieh Varposhti
varposhti@aut.ac.ir

Vesal Hakami
vhakami@iust.ac.ir

Mehdi Dehghan
dehghan@aut.ac.ir

[1] Department of Computer Engineering, Islamic Azad University of Boroujen, Boroujen, Iran

[2] School of Computer Engineering, Iran University of Science and Technology (IUST), PO Box 16846-13114, Hengam St., Resalat Sq., Tehran, Iran

[3] Department of Computer Engineering and Information Technology, Amirkabir University of Technology, Tehran, Iran

pre-deployment analysis. Moreover, in MSNs, the network topology dynamically changes due to node mobility, which results in frequent re-executions of the localization procedure to keep pace with the node movements. Aside from increased processing load on the sensors, higher signaling overhead would result from the network-wide dissemination of the location information. On the other hand, although GPS-based localization schemes can also be used to determine node locations within a few meters, the cost of GPS devices and the non-availability of GPS signals in confined environments prevent their use in many deployment scenarios. Overall, given the desire to maintain the inherent adaptability of the sensor network, it is essential to develop distributed algorithms that perform coverage without explicit localization information. Also, the location-dependent mobility strategies are not applicable to error prone scenarios where no reliable location information can be obtained.

The literature on MSN coverage also includes self-regulated deployment strategies which guide the movement of the sensors without assuming the availability of the nodes' positions (Esnaashari and Meybodi 2011; Howard et al. 2002). However, the existing strategies are either based on some restricting assumptions or require some form of metric information about the sensor field. For example, the authors in Howard et al. (2002) assume that each node is equipped with a sensor which determines the range and the bearing of the nearby nodes, and (Esnaashari and Meybodi 2011) assumes that an estimation of the size of the area as well as the total number of sensors is available to all nodes.

In this paper, we design a location-free mobility strategy for sensor node deployment and area coverage. We consider a setting where the sensors have no information about the area, the number of the nodes, their location, and even the distances to each other. To the best of our knowledge, such a problem with these restricted assumptions has not been studied yet. We aim at filling this gap with the following contributions:

- In the more realistic case where no metric information is available, and the nodes are not aware of their positions, we resort to a recent technique for building a coordinate-free abstract representation of the network's coverage topology in terms of simplicial complexes (which are mathematical objects from the field of algebraic topology) (Varposhti et al. 2012, 2014, 2015; Silva and Ghrist 2006; Muhammad and Jadbabaie 2007). This representation accurately captures the information of overlapping areas between the sensors, and can be obtained through simple local computations, with no need for a localization procedure.
- With the above representation, each mobile sensor is armed with an abstract local view of its neighborhood. Since we are interested in the automated self-deployment of the nodes, we cast this problem as a non-cooperative game as it is a well-established tool for modeling coordination problems under limited informational assumptions. In particular, we present a formulation of the coverage problem as an exact potential game (Monderer and Shapley 1996) in which each sensor tries to maximize its achievable coverage by keeping the overlapping areas with its neighbors as small as possible. Potential games are a subclass of non-cooperative games with nice properties which can lead to an enhancement of global performance at the equilibrium point.
- When using potential games, an important issue is to devise distributed learning algorithms, using local information and processing abilities, to reach a Nash equilibrium (NE) of the game. A variety of decentralized learning rules have been proposed for optimal action selection in potential games (Marden et al. 2009a, b). These algorithms vary mostly in terms of their information requirements for convergence to equilibrium. Our proposed location-free mobile sensor deployment (LFMSD) algorithm in this paper is a modification of the learning procedure devised for camera sensor networks in Zhu and Martínez (2013). In our LFMSD algorithm, each sensor only remembers its played actions, and its obtained utility values during the last two plays. Also, the number of 1-simplices and 2-simplices of each node is used as a measure to determine the distance that the node should move. This way, the nodes execute a very lightweight procedure to learn their positions of high coverage.
- While LFMSD uses a similar action update law as (Zhu and Martínez 2013) (with a modified exploration step), we note that the existing convergence analysis is very much tied with the specifications of the problem in Zhu and Martínez (2013), and it thus cannot be directly applied to our case. Hence, another contribution of our work pertains to the rigorous convergence analysis of the deployed learning algorithm and derivation of new conditions for convergence to NE.
- We conduct extensive simulation experiments to evaluate the proposed algorithm, and to compare its performance against existing schemes.

The rest of the paper is organized as follows: Sect. 2 gives an overview of sensor deployment strategies in MSNs. In Sect. 3, we present the system model and establish our assumptions regarding the coverage problem. Section 4 begins with a brief review of the relevant game-theoretic notions, and then delves into the problem formulation. This section ends by presenting our proposed distributed algorithm for location-free mobile sensor deployment. Section 5 is dedicated to the convergence analysis of the proposed algorithm. Simulation results are given in Sect. 6. We conclude the paper in Sect. 7.

2 Related works

Wireless sensor networks (WSNs) have been widely studied in recent years, focusing on those with stationary sensor nodes; a recent survey can be found in Rawat et al. (2014). One of the key problems in WSN applications is effective area coverage using stationary sensors, which has been examined in related work from many aspects, such as: field versus path coverage and deterministic versus probabilistic coverage (Zhu et al. 2012). The advances in embedded systems and hardware designs have realized mobile sensors, such as Robomote (Sibley et al. 2002). Using mobile sensors for coverage has recently attracted a lot of attention since network performance can be greatly improved by using just a few of mobile nodes. In this section, we present a brief review of the related works on sensor deployment strategies for area coverage in the context of MSNs.

Wang et al. (2009) categorize the existing movement schemes into three categories as: *coverage-pattern-based* movement, *virtual-force-based* movement and *grid-quorum-based* movement.

Algorithms in the group of coverage-pattern-based movement (Wang et al. 2008; You-Chiun and Yu-Chee 2008) try to relocate the mobile nodes based on a predefined coverage pattern such as regular hexagons.

In virtual-force-based node movement strategies (Howard et al. 2002; Zou and Chakrabarty 2003; Heo and Varshney 2005; Wang et al. 2006; Yuan et al. 2014; Li et al. 2012), the sensor nodes apply virtual forces to their neighboring nodes to repel each other and obstacles after an initial random deployment to maximize the covered area. According to the resultant force vector applied by the neighboring nodes, the direction and the distance that the node should move are calculated. Wang et al. (2006) have proposed distributed protocols for controlling the movement of the sensors, in which Voronoi diagrams are used to detect coverage holes. Connectivity-Preserved Virtual Force (CPVF) (Tan et al. 2009) is an enhanced form of the virtual-force-based method that is designed to maximize sensing coverage with the additional consideration of the connectivity requirements. The CLA-DS algorithm in Esnaashari and Meybodi (2011) is a self-regulated deployment strategy based on cellular learning automata (Meybodi et al. 2003). Neighboring nodes apply forces to each other, and the movement is based on the resultant force vector. Each node is equipped with a learning automaton which decides whether to apply force to its neighbors or not. Therefore, each node in cooperation with its neighboring nodes learns its best location to attain high coverage.

In the group of grid-quorum-based movement methods (Chellappan et al. 2007; Jiang et al. 2008; Wang et al. 2005; Yang et al. 2007; Wu and Yang 2007), the area of the network is partitioned into some small grid cells. Similar to a load balancing problem, each mobile sensor node must find a suitable cell and move to that target cell. The authors in Zhu and Martínez (2013) present payoff-based learning algorithms for coverage in visual mobile sensor networks. They discretize the mission space into a squared lattice. The utility function is defined based on the number of sensors that cover each square. Then each sensor learns how to move to maximize its utility function which corresponds to find its best position based on the location of its neighbors.

Some other algorithms are designed for coverage hole recovery (Tamboli and Younis 2010; Sahoo et al. 2010; Wang et al. 2004; Izadi et al. 2015). In Tamboli and Younis (2010), the algorithm is designed to overcome the problem of created holes caused by failed sensor nodes by temporarily replacing the failed node with one or multiple of its neighbors. In Sahoo et al. (2010) a distributed algorithm is proposed to recover the holes in which the magnitude and the direction of the mobile node is calculated by the laws of vectors. In Wang et al. (2004), the Vector-based Algorithm (VEC) is motivated by the attributes of electro-magnetic particles. The algorithm identifies existing coverage holes in the network, and the virtual force pushes the sensors away from each other if coverage hole exists in either of their Voronoi polygons. The authors in Izadi et al. (2015) have proposed a fuzzy-based self-healing coverage scheme for randomly deployed mobile sensor nodes. They first determine the uncovered sensing areas and then the best mobile nodes are selected to move there in order to minimize the coverage hole. The authors in Bartolini et al. (2010) have proposed a fuzzy-based self-healing coverage scheme for randomly deployed mobile sensor nodes. They first determine the uncovered sensing areas, and then the best mobile nodes are selected to move there in order to minimize the coverage hole. The authors in Cortes et al. (2004) have proposed a distributed control and coordination algorithm for computing the optimal sensor deployment of a class of utility functions that encode optimal coverage and sensing policies. In Fletcher et al. (2010a), a localized algorithm is proposed in which mobile robots carry static sensors and drop them at empty vertices of a virtual grid for full coverage. In Casteigts et al. (2012), the problem of self-deployment of a mobile sensor network with simultaneous consideration to fault tolerance, coverage, diameter, and quantity of movement is investigated.

Many of these algorithms strive to spread sensors to desired positions based on the location information of sensor nodes and their neighbors in order to obtain a stationary configuration such that the coverage is optimized. The main difference is how the desired sensor positions are computed. The reader may note that the algorithm in Esnaashari and Meybodi (2011) is a location-free algorithm, however unlike the case in this paper, the authors in Esnaashari and Meybodi (2011) assume that the sensor nodes know the total number of the nodes as well as the size of the area. In the setting

we consider, the MSN coverage problem is tackled without any sensor knowing its position or its relative distance to other sensors. In addition, the size of the area and the total number of sensor nodes are not known to the nodes. Finally, compared to Esnaashari and Meybodi (2011), it should be noted no theoretical performance/stability guarantee (e.g., in the sense of a game-theoretic equilibrium) has been given in Esnaashari and Meybodi (2011). In fact, the approach in Esnaashari and Meybodi (2011) is a heuristic design based on the notion of learning automata, while LFMSD is founded on a well-established design methodology which promotes game-theoretic formulations for distributed control and optimization (Li and Marden 2013).

3 System model and assumptions

We consider a two-dimensional mission space, where N sensors are initially located at unknown coordinates. Without loss of generality (and primarily for uncluttered notation), we assume that all sensors have a coverage radius of r_c, and that each sensor i can communicate with any sensor whose distance from i is less than the transmission radius r_b. The total area coverage, C, is the union of the individual coverages of all the sensors. Each sensor can move based on a motion vector in four directions. In this work, we have two assumptions:

Assumption 1 $r_b \approx 2r_c$. This assumption is in line with most of the prior works on coverage problem in wireless sensor networks (e.g. Esnaashari and Meybodi 2011). However, we have to state that, theoretically, a more routinely assumed condition to compute the number of 2-simplices is $r_b \leq \sqrt{3}r_c$ (Yan et al. 2014). Assuming a larger transmission radius can lead to the detection of a higher number of overlaps and possibly even to slightly over-count the simplices (in some rare edge cases). More extensive discussion on the impact of the relationship between transmission range and sensing range on the accuracy of the simplex counting procedure can be found in Yan et al. (2014).

Assumption 2 Sensor nodes cannot determine their location in the global coordinate system of the mission field.

Since sensor nodes do not have any location information, they cannot measure the overlapping area. To overcome this problem, we resort to a 3-dimensional graph abstraction of the sensor network, namely the "simplicial complex" of the network, which encapsulates information about the neighbors and overlapping areas (Silva and Ghrist 2006; Hatcher 2002). Recently, simplicial complexes have been used to model the coverage of sensor networks and to solve the coverage problems in sensor networks without location information (Varposhti et al. 2012, 2014, 2015). The simplicial complex of the network is a set consisting of 0-simplice or vertices, 1-simplices or edges, and 2-simplices or triangles, which can be constructed in a distributed manner from the communication graph of the network. Each 0-simplex corresponds to a vertex, and each 1-simplex corresponds to an edge in the communication graph of the network. Each 2-simplex corresponds to the overlap of three sensors. The simplicial complex of the network, R_N, can be generated as follows (Muhammad and Jadbabaie 2007):

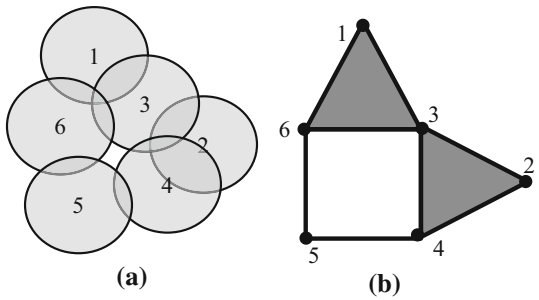

Fig. 1 **a** A sensor network, **b** the simplicial complex of the network

The sensor nodes simultaneously broadcast their identifications. The nodes within the communication radius of each other receive and store this information and pair the received tag with their own tags, which results in the generation of the 1-simplices or the edges in the simplicial complex. Subsequently, the nodes broadcast their list of edges, to form the *2*-simplices. In this way, the number of neighbors and the number of overlapping areas of each node is calculated using only local information exchanges. A brief review of simplicial complexes is provided in "Appendix A".

A sensor network and its corresponding simplicial complex are shown in Fig. 1. There are six 0-simplices, eight 1-simplices, and two 2-simplices. 2-simplex [1 3 6] corresponds to the intersection of disks 1, 3, and 6, and 2-simplex [2 3 4] corresponds to the intersection of disks 2, 3, and 4. Therefore, sensors 1, 2, 4, 6 have one 2-simplex, sensor 3 has two 2-simplices, and sensor 5 does not have any 2-simplices.

We use the number of 2-simplices as a measure of the amount of overlap area. So, each sensor desires to have the minimum number of 1-simplices and 2-simplices.

4 Problem formulation

In this section, we present a game-theoretic formulation of the location-free coverage problem in MSNs. Our formulation is motivated by the self-configuration feature of the game-theoretic design which can derive rich dynamics through the interaction of simple components (Fudenberg and Tirole 1991). In essence, the game-theoretic concept of equilibrium describes a condition of global coordination where all deci-

sion makers are content with the social welfare realized as the consequence of their chosen strategies. The most prominent equilibrium notion in game theory is Nash equilibrium (NE) (Nash 1951). However, the existence and convergence to NE do not always apply to arbitrary utility functions and strategy sets. As such, we resort to an important class of games, namely potential games (Monderer and Shapley 1996) in which at least one pure-strategy NE is guaranteed to exist and can be reached with certain classes of learning dynamics such as iterative best responses.

Given our game-theoretic design, in Sect. 4.1, we give a brief background on the relevant notions from the game theory literature. Then, in Sect. 4.2, we cast the problem of area coverage as a potential game. Finally, in Sect. 4.3, we propose a distributed learning procedure which can be viewed as a self-deployment mechanism for the mobile sensors with guaranteed convergence to NE of the formulated potential game.

4.1 Background on potential games

A strategic game $\Gamma := \langle V, \boldsymbol{A}, U \rangle$ consists of three components: A set of players, V, an action set $\boldsymbol{A}$, and the collection of utility functions U, where the utility function u_i models the preferences of the ith player over action profiles. The profile $\boldsymbol{a} = (a_1, \ldots, a_N) \in (A_1 \times \ldots \times A_N)$ is called a joint action, where a_i is the action of the ith player. The action profile of all players except player i is denoted by a_{-i}, and the set of action profiles for all players other than player i is denoted by $\boldsymbol{A}_{-i} = \prod_{j \neq i} A_j$.

Definition 1 [*Nash equilibrium (NE)*] (Fudenberg and Tirole 1991). Consider the strategic game Γ. An action profile $\boldsymbol{a}^* := (a_i^*, \boldsymbol{a}_{-i}^*)$ is an NE of the game Γ if $\forall i \in V$ and $\forall a_i \in A_i$, it holds that $u_i(\boldsymbol{a}^*) \geq u_i(a_i, \boldsymbol{a}_{-i}^*)$.

Definition 2 [*Exact Potential Game (EPG)*] (Monderer and Shapley 1996). The strategic game Γ is an EPG with potential function $\varphi : \boldsymbol{A} \rightarrow R$ if for every player i, for every $\boldsymbol{a}_{-i} \in \boldsymbol{A}_{-i}$, and for every $a_i, a_i' \in A_i$, it holds that:

$$\varphi(a_i, \boldsymbol{a}_{-i}) - \varphi(a_i', \boldsymbol{a}_{-i}) = u_i(a_i, \boldsymbol{a}_{-i}) - u_i(a_i', \boldsymbol{a}_{-i}) \quad (1)$$

In words, in a potential game formulation, one can identify a special function called the *potential function* , which changes values whenever there is a change in the utility of any single player due to his/her own strategy deviation. As such, we have the following theorem as a standard result in game theory (Monderer and Shapley 1996):

Theorem 1 *If $\varphi(.)$ is a potential function for the game Γ, and $\boldsymbol{a}^* \in \arg\max_{\boldsymbol{a}} \varphi(\boldsymbol{a})$ is a maximizer of the potential function, then $\boldsymbol{a}^*$ is an NE of the game Γ.*

Sometimes the actions available for player i is restricted to a state-dependent subset of A_i, denoted by $F_i(a_i, a_{-i}) \subseteq A_i$, which is the set of feasible actions of player i when the action profile is $(a_i, \boldsymbol{a}_{-i})$. The introduction of F leads to the notion of restricted strategic game $\Gamma_{res} := \langle V, \boldsymbol{A}, U, F \rangle$ which consists of four components: $V, \boldsymbol{A}, U$, and F.

Definition 3 (*Restricted NE*) An action profile $\boldsymbol{a}^*$ is a restricted NE of the restricted strategic game Γ_{res} if $\forall i \in V$ and $\forall a_i \in F_i(a_i^*, \boldsymbol{a}_{-i}^*)$, it holds that $u_i(\boldsymbol{a}^*) \geq u_i(a_i, \boldsymbol{a}_{-i}^*)$.

Definition 4 (*Restricted EPG*) The game Γ_{res} is a restricted EPG with potential function $\varphi(\boldsymbol{a})$ if for every player i, for every $\boldsymbol{a}_{-i} \in \boldsymbol{A}_{-i}$, and for every $a_{-i} \in A_{-i}$, Eq. (1) holds for every $a_i' \in F_i(a_i, a_{-i})$.

The existence of NE in an EPG is guaranteed (Monderer and Shapley 1996). It then follows that any restricted EPG has at least one restricted NE.

4.2 MSN coverage problem as an EPG

In the MSN coverage problem, we are concerned with devising a motion law for repositioning of a number of mobile sensors so that their converged positions in the limit correspond to a deployment with desirable coverage performance. In this section, we present our formulation of this problem in terms of a restricted EPG $\Gamma_{mc} := \langle V, A, U_{mc}, F_{mc} \rangle$. In what follows, we describe the game components in more detail:

- **Player set** V The set of players consists of the N sensors in the mission space, denoted by $V = \{s_1, s_2, \ldots s_N\}$
- **Action set** A The action of a player i is a location $a_i = (x_i, y_i)$. Accordingly, the action set of player i is $A_i = \{a_j \mid a_j$ is a position inside the mission space$\}$. The joint action set across all players is $\boldsymbol{A} = \prod_{i=1}^{N} A_i$.
- **Feasible action set** F_{mc} The feasible action for each agent is determined based on a motion vector in any four cardinal directions; more formally, we have $F_{mc} = \prod_{i=1}^{N} \bigcup_{a_i \in A} F(a_i)$, where $F(a_i)$ is the set of feasible locations that a sensor in location a_i can move to.
- **Utility functions** In general, a global planner would like the sensors to distribute in the mission space uniformly to achieve maximum coverage. Intuitively, at the local sensor-level this translates to the case that each sensor keeps the overlapping areas with nearby sensors as small as possible. Let $n_1(i)$ and $n_2(i)$ denote the number of 1-simplices (i.e., immediate neighbors), and the number of 2-simplices of sensor s_i, respectively. Now, we consider the following welfare function:

$$\varphi(\boldsymbol{a}) = \sum_{i=1}^{N} \left(\frac{1}{n_1(i)+1} + \frac{1}{n_2(i)+1} \right) \quad (2)$$

In light of the discussion in Sect. 3, maximizing $\varphi(\boldsymbol{a})$ corresponds to reducing the total number of overlapping areas.

Remark 1 Other choices of welfare function were also possible. However, our theoretical results in Sect. 5 do not work directly on the explicit form of the welfare (potential) function to provide an estimate of the convergence rate. Our proposed learning rule in Sect. 4.3 prescribes a very simple action update law for the sensors that does not entangle the potential function form. This is as opposed to conventional learning rules such as gradient play (Flåm 2002). In those cases, the functional form of the potential and its properties such as smoothness, differentiability, or Lipchitz continuity can be exploited to provide stronger theoretical results. In our case, without positional information, there is no straightforward formula expressing the number of 1- and 2-simplices as a function of the joint action of the sensor nodes. Instead, the sensors have to perceive the numerical value of their utility via counting the resultant simplices after every motion.

Now, to solve the global optimization problem in a distributed fashion, this is where the notion of EPG comes into play; in an EPG, the sensors (as players) cooperate to accomplish the global goal while pursuing their local interests. In fact, it is well known that if we assign each player a utility function that captures the marginal contribution of that player to the overall welfare function, the resultant game would be EPG, with the welfare function acting also as the potential function of the game (Arslan et al. 2007; Marden et al. 2009a). Now, in light of Theorem 1, it thus remains to design a distributed algorithm for directing the sensor's movements towards the equilibrium of the EPG. To this end, we define the utility of sensor s_i, given any action profile a, as follows:

$$u_i(a) = \varphi(a_i, a_{-i}) - \varphi(a_i^0, a_{-i}) \tag{3}$$

where action a_i^0 is equivalent to sensor s_i turning off its sensing unit. In fact, for computing $\varphi(a_i^0, a_{-i})$, we assume that sensor i is excluded from the society of the sensors. Hence, for each sensor j which is a neighbor of sensor i, the j's neighbor set decreases by one. Also, for such sensor j, all 2-simplices that $[i\,j]$ is its face should be denied. So, the utility of sensor s_i can be computed by (4):

$$u_i(a) = \frac{1}{n_1(i)+1} + \frac{1}{n_2(i)+1} - \sum_{j\in N_1(i)} \frac{1}{(n_1(j))(n_1(j)+1)} + \sum_{j\in N_2(i)} \left(\frac{1}{(n_2(j)+1)} - \frac{1}{(n_2^{\bar{i}}(j)+1)} \right) \tag{4}$$

where $n_2^{\bar{i}}(j)$ is the number of 2-simplices of jth sensor by assuming that ith sensor is off, $N_1(i)$ is the set of neighbors of sensor s_i, and $N_2(i)$ is the set of sensors which are in a triple overlapping with sensor s_i (triple overlapping means that three sensors have overlapped area with each other). Note that the utility function u_i can be computed by local information as it only depends on the action of sensor s_i and its neighbors.

4.3 Location-free mobile sensor deployment

In this section, we propose a distributed learning algorithm for the self-deployment of mobile sensors, which we call *location free mobile sensor deployment* (*LFMSD*). In fact, given our game-theoretic formulation in Sect. 4.3, LFMSD is essentially a distributed learning algorithm that directs the sensor movements towards the equilibrium configuration, where the global objective $\varphi(\boldsymbol{a})$ in (2) is maximized.

There are many adaptive learning procedures for convergence to equilibria in potential games (Marden et al. 2009a; Monderer and Shapley 1996). These procedures vary mostly in terms of their information requirements. In our setting, each sensor, can determine the utility of the currently taken action only after it gets relocated (i.e., by counting the number of 1-simplices and 2-simplices in the new location it ends up in). However, since a sensor is unaware of the coordinate it would land in, it is unable to evaluate its utility function for the alternative moves. In other words, our EPG formulation corresponds to the case of unknown utilities (Monderer and Shapley 1996). Hence, standard procedures such as best response dynamics or the adaptive play learning algorithm (Young 1993) cannot be employed to solve our problem. In fact, we should design a so-called "payoff-based" learning algorithm (Marden et al. 2009b) where each player adapts its decisions based on only the numerical value of the realized payoffs (utilities). Armed with this understanding, we use a similar procedure as the DISCL algorithm in Zhu and Martínez (2013), albeit with a modified exploration step (Sutton and Barto 1998). DISCL has been proposed in the context of camera sensor networks, and is the most recent variant of payoff-based learning algorithms for potential games which uses a diminishing exploration rate to obtain a stronger NE convergence result.

In what follows, we introduce relevant notations, and give an overview of our LFMSD algorithm. Figure 2 summarizes the steps of the algorithm.

For each sensor s_i at time step t, we define $\tau_i(t)$ as:

$$\tau_i(t) = \begin{cases} t & u_i(a(t)) \geq u_i(a(t-1)) \\ t-1 & else \end{cases} \tag{5}$$

In other terms, $a_i(\tau_i(t))$ is the more successful action of the ith sensor in the last two steps. At each round t, each node i determines the number of its 1-simplices, $n_1^t(i)$ as

The LFMSD Algorithm

1. At $t = 0$ all sensors are placed in D. each sensor s_i communicates with its neighbors and computes $n_1(i)$, and $n_2(i)$, and by local message passing computes $u_i(a(0))$. At $t = 1$, all sensors stick with their actions.
2. At $t > 1$, each sensor s_i updates its state according to the following rules:
   a. Sensor s_i selects a random number r_i that is an integer between 1 and $n_{AF}(i,t)$. It computes $a_i(\tau_i(t))$ and takes note of the exploration rate $\varepsilon(t)$.
   b. With probability $\varepsilon(t)$, sensor s_i chooses an exploratory action uniformly from the set $[r_i \times F_i\,(a_i\,(t))] \setminus \{a_i\,(\tau_i(t))\}$.
   c. With probability $1-\varepsilon(t)$, sensor s_i select its greedy action $a_i\,(\tau_i(t))$.
   d. Sensor s_i moves according to the selected action.
3. At position a_i, each sensor s_i determines n_1 and n_2 of itself and of its neighbors by local message passing, and computes the utility $u_i(a_i)$.

Repeat steps 2 and 3 until convergence.

Fig. 2 The LFMSD algorithm

well as the 2-simplices, $n_2^t(i)$, by exchanging messages with its neighbors. For each sensor s_i, we define $n_{AF}(i,t)$ as:

$$n_{AF}(i,t) = \max\left(n_1^t(i), MS_i(t^-)\right) \tag{6}$$

where $MS_i(t^-)$ is the number of steps that sensor s_i moved as a result of its last action. In fact, $n_{AF}(i,t)$ specifies the distance sensor node i should move at time step t.

As with standard reinforcement learning (Sutton and Barto 1998), LFMSD is basically an iterative execution of two complementary steps: *exploration* and *exploitation*. At each time step, sensor s_i updates its action by either trying some new random action in the feasible action set (exploration) or by selecting the best action from those played in the last two steps (exploitation).

More specifically, Let $\varepsilon(t)$ be a sequence of diminishing exploration rates, i.e., $\{\varepsilon(t)\}_t \to 0$ as $t \to \infty$. At each time t, with probability $\varepsilon(t)$, sensor s_i selects a random integer number r_i which is between 1 and $n_{AF}(i,t)$, and chooses an action uniformly from the set $[r_i \times F_i\,(a_i\,(t))]\backslash\{a_i\,(\tau_i(t))\}$ (i.e., exploration step). With probability $1-\varepsilon(t)$, sensor s_i does not experiment and sets its action to $a_i(\tau_i(t))$ (i.e., exploitation step with greedy action selection). In fact, $r \times F_i(a)$ shows the set of feasible actions of player i where its current location is a, and the selected magnitude of motion is r. In other words, if s_i is located at $(x_i(t), y_i(t))$ at time step t, it moves to $(x_i(t+1), y_i(t+1)) = (x_i(t) + r_i \cos\theta, y_i(t) + r_i \sin\theta)$ with probability $\varepsilon(t)$, where θ is the selected direction, and with probability $1-\varepsilon(t)$, sensor s_i moves to $(x_i(t+1), y_i(t+1)) = (x_i(\tau_i(t)), y_i(\tau_i(t)))$.

Remark 2 LFMSD is an example of an equilibrium learning procedure with synchronous execution steps. However, in the context of potential games, there are other variants of payoff-based learning algorithms which can work in asynchronous fashion (e.g., see Zhu and Martínez 2013; Li and Marden 2013). Based on these results, we could easily come up with an asynchronous variant where at each time step, only one sensor is active and updates its state by either trying some new action or selecting an action from those played in the last two time steps when it was active. However, under the same exploration rate, such asynchronous variant needs a longer time than LFMSD to converge.

Remark 3 Deploying LFMSD in a purely infrastructure-less network (which relies solely on multi-hop communications) is vulnerable to network partitioning, especially when the number of sensors is small. However, achieving joint coverage and connectivity in the absence of any metric information is a much harder problem. In case where highly reliable communications are desired, one way to justify the practical adoption of our algorithm is to restrict its applicability to a limited number of use case scenarios. For example, one may consider a setting where along with the sensor nodes, a few more powerful hub nodes are deployed, which can be either static or mobile (e.g., mini unmanned aerial vehicles (UAVs) that cruise above the ground through different clusters of sensors to collect data). These hubs can be used to collect information from sensors, form a backbone which receives, processes and re-routes sensor data to the end-users. The hubs can also be assumed to be highly functional, with reliable routing protocols for sending information error-free along a multi-hop path to the end-users. However, in this paper we concern ourselves only with area coverage. While the issues of connectivity, route optimization, topology control, and optimal transmission, are equally important, but they lie beyond our scope. Indeed, the purpose of auxiliary communication devices (such as the hubs) is to abstract away these issues.

5 Convergence analysis of LFMSD

In this section, we focus on the convergence properties of the LFMSD algorithm. Our convergence analysis draws on the strong ergodicity results for time-inhomogeneous Markov chains as given in Isaacson and Madsen (1976) and Anily and Federgruen (1987). Due to space limitations, the reader is referred to "Appendix B" for some background concepts on non-stationary Markov chains.

In LFMSD, the play history has length two. Let $z(t) := (\boldsymbol{a}(t-1), \boldsymbol{a}(t))$ be the joint history at time t. Also, denote by $\boldsymbol{B} := \left\{(\boldsymbol{a}, \boldsymbol{a'}) \in \boldsymbol{A} \times \boldsymbol{A} | \boldsymbol{a'}_i \in F(a_i), \forall i \in V\right\}$, the collection of all possible joint histories. After an action profile $\boldsymbol{a}(t)$ is realized at time t, the joint history moves from $z(t)$ to its successor $z(t+1)$. To express the transition probabilities,

consider an arbitrary move $z^t \to z^{t+1}$. We partition the set of players V into two disjoint subsets: $\Lambda(z^t \to z^{t+1})$ and $\Omega(z^t \to z^{t+1})$. Partition Λ entails those players which have chosen their next move to be the best of their two previous actions, i.e., $\Lambda(z^t \to z^{t+1}) := \{i \in V | a_i^{t+1} = a_i^{\tau_i(t)}\}$. Partition Ω denotes the subset of players that make their move randomly from their remaining actions (excluding $a_i^{\tau_i(t)}$), i.e., $\Omega\left(z^t \to z^{t+1}\right) := \left\{ i \in V | a_i^{t+1} \sim \frac{1}{\left|F_i(a_i^t) \backslash \left\{a_i^{\tau_i(t)}\right\}\right|} \right\}$. Therefore, the transition matrix of LFMSD at time t with perturbation $\varepsilon(t)$ is P^{ε}, in which $P_t^{\varepsilon(t)}(z^t, z^{t+1})$ is given by:

$$P_t^{\varepsilon(t)}(z^t, z^{t+1}) = (1-\varepsilon(t))^n \prod_i 1\left(a_i^{t+1} = a_i^{\tau_i(t)}\right) + (1-\varepsilon(t))^{|\Lambda(z^t \to z^{t+1})|} \times \varepsilon(t)^{|\Omega(z^t \to z^{t+1})|} \prod_{\substack{i \in \Omega(z^t \to z^{t+1}) \\ \Omega \neq \emptyset}} \Pr(a_i^t, a_i^{t+1}) \tag{7}$$

where $\Pr(a_i^t, a_i^{t+1})$ is the probability of taking the action of a_i^{t+1} by s_i given that its current location is a_i^t. Since this probability is based on the number of neighbors of s_i, it is independent from the time.

We note that the action adjustment procedure in LFMSD as well as the information history available to each player corresponds to the game-theoretic learning algorithm of DISCL proposed in Zhu and Martínez (2013) for potential games. Therefore, we may establish the convergence of LFMSD by drawing extensively on the analysis given in Zhu and Martínez (2013). As with the standard methodology for convergence analysis of payoff-based learning dynamics to NE of potential games (e.g., Young 1993), the fundamental concept underlying the convergence analysis in Zhu and Martínez (2013) is the stochastic stability of the Markov chain $\{z(t)\}$ (see Definition 5 in the following) (Isaacson and Madsen 1976):

Definition 5 (*Stochastically stable states of Markov chain* P^{ε}) A state z is said to be *stochastically stable* with respect to a family of chains $P^{\varepsilon(t)}$ if $\lim_{t\to\infty} \mu^{\varepsilon(t)}(z) > 0$, where μ^{ε} is a stationary distribution of P^{ε}.

In Zhu and Martínez (2013) Theorem 4.3, it has been proved that the stochastically stable states of the game's underlying Markov chain are the histories consisting entirely of a single strict Nash equilibrium in a given potential game.

However, there exists a technical complication that should be addressed before the above result becomes applicable: With time-varying $\varepsilon(t)$, the evolution of states $z(t) := (a(t-1), a(t))$ corresponds to a time-inhomogeneous Markov chain $\{P_t^{\varepsilon(t)}\}$. Hence, we first need to establish the ergodictiy of this non-stationary Markov chain before applying a convergence proof based on stochastic stability arguments. In the presence of time-inhomogeneity, the erogidicity property of $\{P_t^{\varepsilon(t)}\}$ is concluded only if it is strongly ergodic (with weak ergodicity as a pre-condition).

Once the strong ergodicity of $\{P_t^{\varepsilon(t)}\}$ is established, we can leverage on the main result of Zhu and Martínez (2013) to associate the stochastically stable states of the chain $\{P_t^{\varepsilon(t)}\}$ with the Nash equilibria of the game Γ_{mc}; however, we need to take a slightly different path to establish the pre-conditional ergodicity property of $\{P_t^{\varepsilon(t)}\}$. The reason is that the proof given in (M. Zhu and Martínez 2013) for the weak ergodicity of the game's Markov chain is very much tied with the problem specifications, and it thus cannot be directly applied to our case. In addition to presenting a more general proof for weak ergodicity, we take on a different approach for establishing the strong ergodicity as well, which, different from the one presented in (M. Zhu and Martínez 2013) for DISCL, is much simpler and more straightforward.

Armed with these understandings, Theorem 2 establishes the convergence of LFMSD to a Nash equilibrium configuration. Rather than bore the reader with the daunting details, here we only give an outline of the proof, and defer the detailed argument to "Appendix C".

Theorem 2 *Consider the Markov chain* $\{P_t^{\varepsilon(t)}\}$ *in* (7) *which is induced by the LFMSD algorithm, we have:*

$$\lim_{t\to+\infty} \Pr(z(t) \in \mathrm{diag}(\Re(\Gamma_{mc}))) = 1 \tag{8}$$

where for a generic set χ, $\mathrm{diag}(\chi) := \{(x, x) \in \chi \times \chi | x \in \chi\}$ *and we denote by* $\Re(\Gamma_{mc})$ *the set of restricted NEs of* Γ_{mc}.

Outline of the proof: We prove the theorem in two steps:

- First, we show that the non-stationary Markov chain $\{P_t^{\varepsilon(t)}\}$ is strongly ergodic (c.f., Definition B.3). To this end, we first establish that $\{P_t^{\varepsilon(t)}\}$ is weakly ergodic (c.f., Definition B.4). Next, to verify that $\{P_t^{\varepsilon(t)}\}$ is also strongly ergodic, we rely on the conditions derived by Anily and Federgruen (1987) which provide sufficient conditions for a weakly ergodic non-stationary Markov chain to be strongly ergodic.
- Second, we prove that the distribution of the play histories in LFMSD converges to $\mu^* = \lim_{t\to+\infty} \mu^{\varepsilon(t)}$, with the corollary that in potential games, $\mu^*(z) > 0$ only if z consists of a single strict Nash equilibrium.

6 Simulation results

To evaluate the performance of LFMSD, several experiments have been conducted in NS-3 (www.nsnam.org). As the MAC layer, we use distributed coordination function (DCF) of 802.11b, and each sensor node sends its packets using CSMA/CA protocol. Networks of different sizes

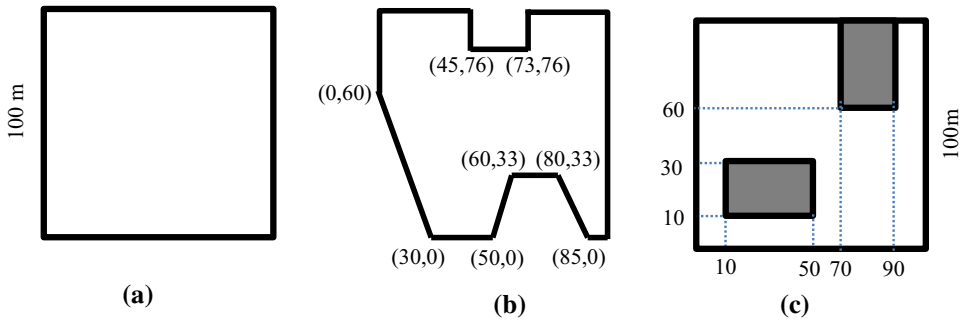

Fig. 3 Simulation area for Experiments 4–6, **a** rectangular area, **b** complex-shaped area, **c** area with narrow passages and obstacles

from $N = 50$ to $N = 200$ sensors are considered for simulations. The exact number of nodes and other simulation parameters such as the coverage radius of the sensors and the dimension/shape of the sensor field are determined specifically for each experiment. In particular, in Experiments 4–6, we perform simulations for three different areas: a rectangular shape area, a complex network area, and an area with two obstacles (and narrow passages) which are shown in Fig. 3.

The performance of LFMSD is compared with that of the potential field-based algorithm abbreviated as PF (Howard et al. 2002), the basic VEC algorithm (Wang et al. 2006), and CLA-DS algorithm (Esnaashari and Meybodi 2011). As briefly reviewed in Sect. 2, PF is in the category of virtual-force-based movement algorithms which works based on the location information of sensor nodes. VEC is a coverage hole recovery algorithm, and is also location-based. CLA-DS works by exerting virtual forces. Similar to LFMSD, it does not require the nodes' location information, but each sensor should be aware of the total number of nodes as well as the size of the area under surveillance. Also, qualitatively, compared to CLA-DS, the LFMSD's merit lies in its reliance on the notion of NE. NE describes a condition of global consensus (coordination), and in the limit, the nodes' joint action space possesses a quiescence property: no node has any incentive to choose a different action. Therefore, reaching an equilibrium can be viewed as all nodes being coordinated in their choice of actions. We realize that not every equilibrium is ideal from the system-wide perspective, but again, in the context of potential games, there is an inherent alignment between the individual and global objectives. On the other hand, CLA-DS does not provide an analytical guarantee of convergence to some global consensus.

In the experiments, we use the following evaluation criteria: coverage (i.e., the percentage of the area covered), average distance travelled by each mobile sensor, and node separation which is defined to be the average distance of all sensors from their nearest neighbor in the deployed sensor network; more formally,

$$\text{Node separation} = \frac{1}{N}\sum_{i=1}^{N} \min_{s_j \in N_1(s_i)} \big(\text{distance}(s_i, s_j)\big) \tag{9}$$

where $N_1(s_i)$ denotes the set of neighbors of sensor s_i. Node separation is an indicator of the overlapping area between the sensing regions of the sensor nodes. The smaller the node separation, the larger will be the overlapping area.

6.1 Experiment 1

For the first experiment, we present a numerical evaluation of our LFMSD algorithm. We consider a 10 m × 10 m square shaped sensor field in which 20 sensors are initially positioned in the middle of the area. The coverage radius of each sensor is 1 m, and the transmission radius is taken to be 2 m. The LFMSD's exploration factor is considered to be:$\varepsilon(t) = 0.1/\ln t$. In this experiment, any configuration in which the sensors do not have overlapping areas maximizes the global potential function, and the corresponding positions of the nodes form an NE of the underlying potential game. Figure 4 demonstrates the stabilized positioning of the nodes at iteration 1000. The reader may note that some sensors are located on the boundaries that might cause wasting their sensing radius. This problem is generally due to the sensors' unawareness of their positions and of the borders of the area. In fact, the utility function of the sensors does not measure "coverage" directly, and it only serves as an implicit indicator. The sensors lack the information to actually measure the size of the area they are exclusively covering; but rather, they seek to minimize the overlap with their neighbors: a behavior which is positively correlated with "coverage". In the particular case of Fig. 4, there are also several other factors contributing to this suboptimal positioning such as: initial deployment, the size of the area, the number of the sensors, their local density, and the step size by which the sensors relocate themselves. Figure 5 shows the evolution of the average number of neighbors of all the sensors which ultimately converges to 0. Finally, Fig. 6 verifies the convergence behavior of the LFMSD algorithm by showing that once the positioning of the mobile sensors stabilizes, the global potential function attains its maximum.

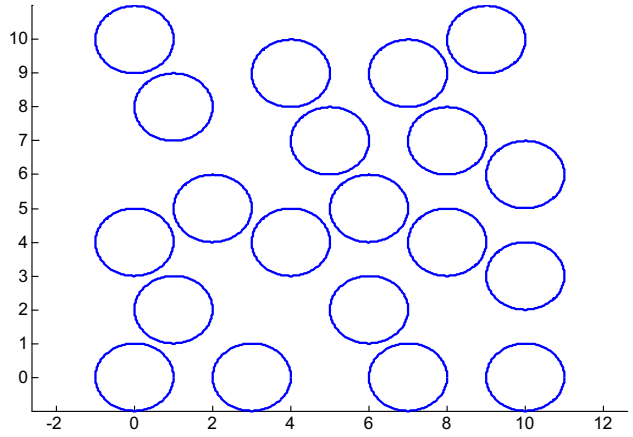

Fig. 4 Final configuration of the network at $t = 1000$

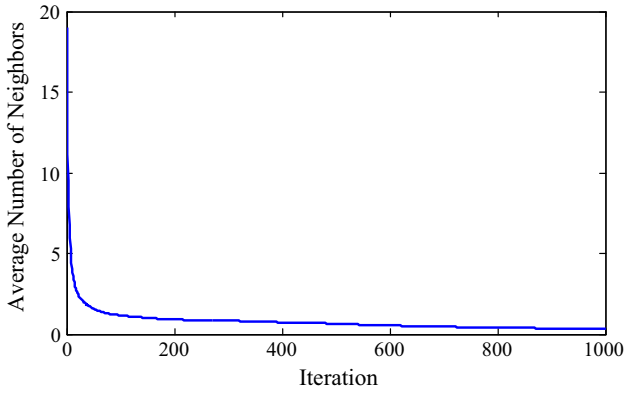

Fig. 5 Average number of neighbors of sensors in a network with N = 20

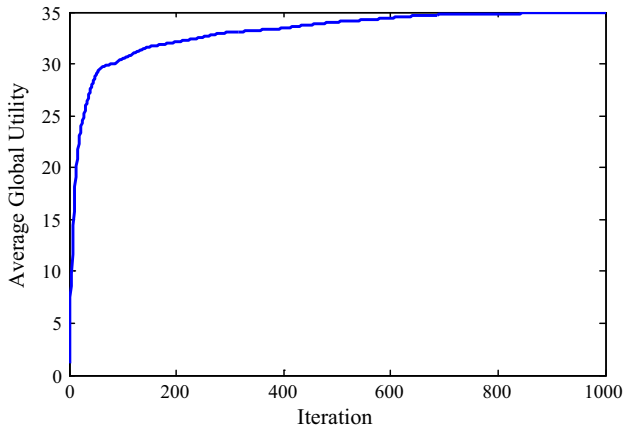

Fig. 6 The evolution of global potential function in a network with N = 20

6.2 Experiment 2

To evaluate the performance of the LFMSD in areas that have obstacles and narrow passages, we consider an office-like indoor area of 15 m × 15 m which is shown in Fig. 7.

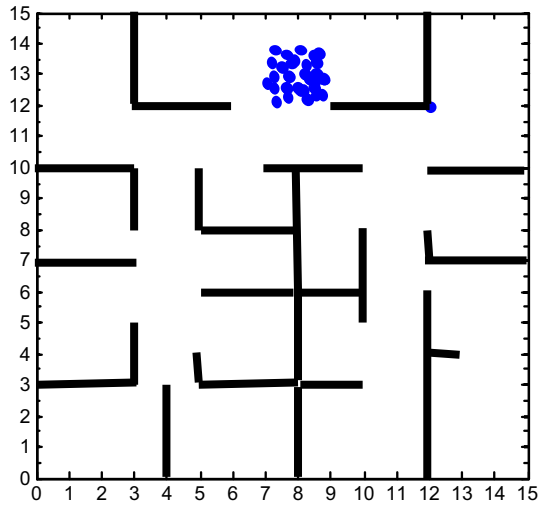

Fig. 7 Initial network configuration

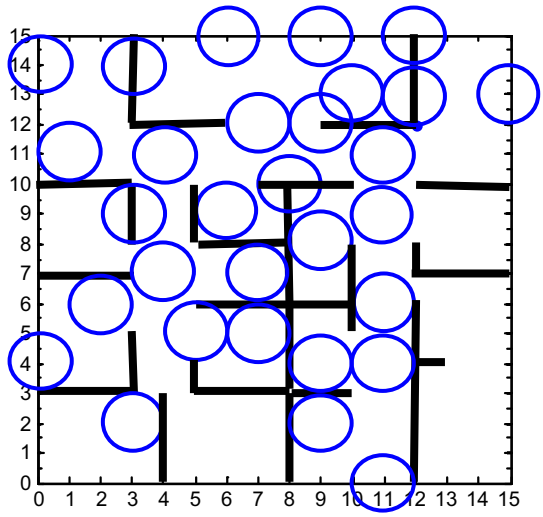

Fig. 8 Final configuration after 3000 iterations

Initially, a total of 30 sensor nodes are located randomly within a 2 by 2 square in the northern part of the area. The sensing radius and the transmission range are assumed to be: 1 m and 2 m, respectively. The final configuration of the network (after 5000 iterations) is shown in Fig. 8. These figures show that the nodes successfully move away from their starting configuration and spread out to cover the area. Also, some have entered the narrow corridors and provided coverage. However, the additional borders can aggravate the wastage of sensing radii. The convergence property is nevertheless unaffected, and we have also plotted the average global utility and the average travelled distance in Figs. 9 and 10, respectively.

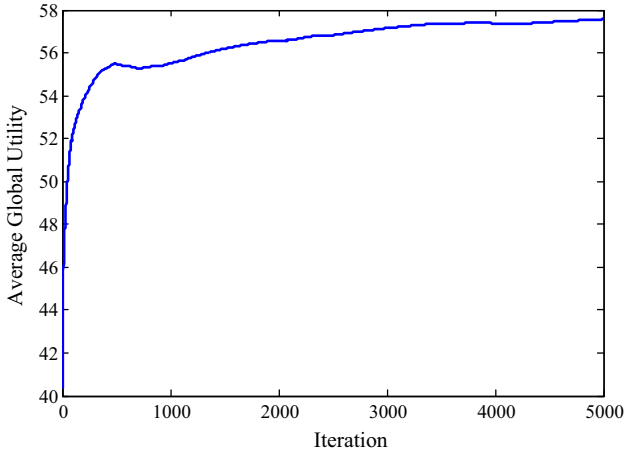

Fig. 9 Average global utility in the area with obstacles

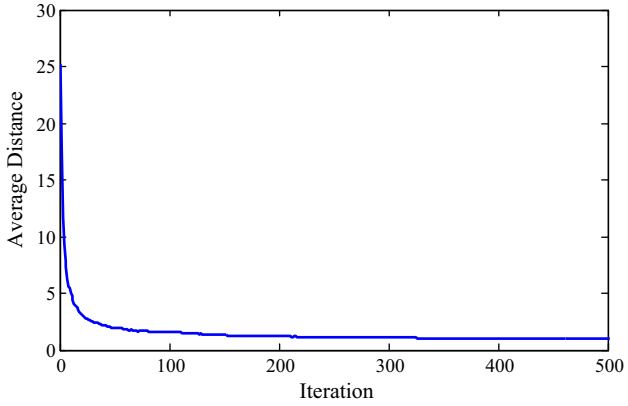

Fig. 10 Average distance travelled within the area with obstacles

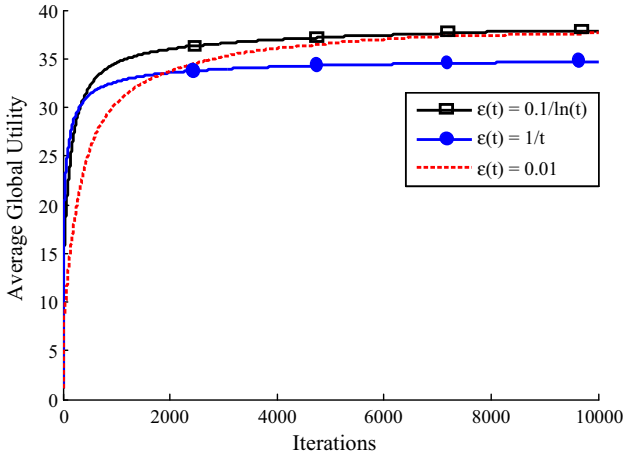

Fig. 11 Evolution of average global potential function for different exploration rates

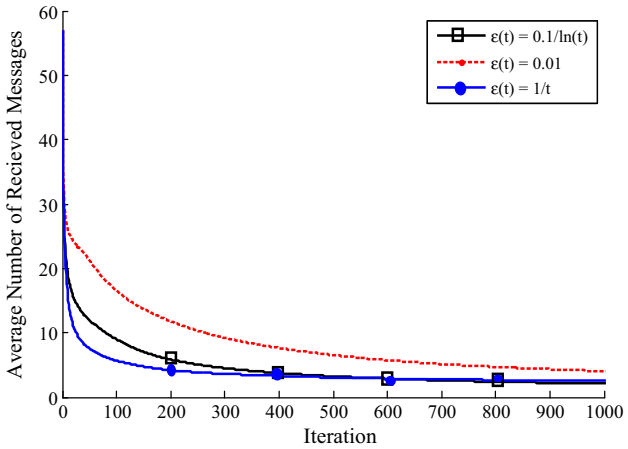

Fig. 12 Average number of received messages for different exploration

6.3 Experiment 3

In this experiment, we investigate the impact of the exploration rate parameter in the convergence of LFMSD. We use the same simulation settings as in Experiment 1, but play with different values of $\varepsilon(t) = 0.01,\ 1/t,\ 0.1/\ln t$. Figure 11 shows the evolution of the average global potential function for 10,000 iterations. As can be seen, for $\varepsilon(t) = 0.01$ and $0.1/\ln t$, the limiting value of the average global utility is the same at iteration 10,000. However, the convergence speed for the time-varying exploration rate happens to be higher compared to when a very small constant rate (i.e. 0.01) is consistently applied throughout the execution. The difference between the two exploration regimes is most noticeable when the number of iterations is small. Intuitively, this is because in the case of a diminishing rate, the algorithm benefits from a relatively large exploration at the start, and this allows for a quicker identification of the unknown environment. As time progresses, the exploration rate is decreased, allowing the algorithm to exploit the information collected and converge to some desired configuration. In fact, in cases where the density of the sensor nodes is high within a small sub-area, it is reasonable to begin with a large exploration rate which diminishes with time. This leads to a more rapid separation of nodes by allowing them to make more "adventurous" moves at the initial stages of the algorithm before finally yielding to a more "prudent" stabilizing behavior.

The average number of received messages and the average distance travelled by each node are shown in Figs. 12 and 13, respectively. When $\varepsilon(t) = 0.01$, the nodes' relocation probability is low, and node density decreases by a lower rate in comparison with $\varepsilon(t) = 1/t$ and $\varepsilon(t) = 0.1/\ln t$. Therefore, in this case, the sensor nodes interact with a higher number of neighbors over a longer period of time, which results in higher amount of message passing.

6.4 Experiment 4

In this experiment, we compare LFMSD with PF, VEC, and CLA-DS algorithms in terms of the node separation crite-

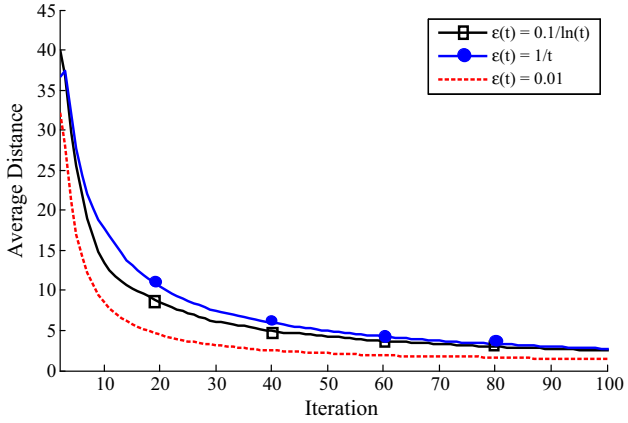

Fig. 13 Average distance travelled for different exploration rates

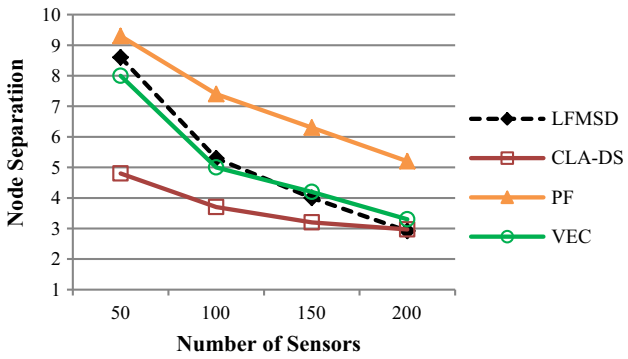

Fig. 14 Node separation within the rectangular network area

rion defined in (9). The sensor nodes are initially positioned randomly within a square region of 10 m × 10 m centered in the middle of the network area. The global sensor field is assumed to be 100 m × 100 m. The experiment is performed for $N = 50, 100, 150$, and 200 sensors. The coverage radius of the sensors is 5 m, and the transmission radius is taken to be 10 m. We choose $\varepsilon(t) = 0.1/\ln(t)$ for the exploration factor in LFMSD, and let the algorithms run for 50,000 iterations. All the results are averaged over 50 runs.

According to the plots shown in Figs. 14 and 15, compared to CLA-DS and VEC, the total overlapping area between the sensor nodes in LFMSD is smaller in a low-density sensor network. However, as shown in Fig. 16, when there are obstacles in the area, PF and VEC outperform LFMSD and CLA-DS, especially in high-density sensor networks. This is because PF and VEC are capable of determining the borders of obstacles. However, "node separation" alone is not a sufficient metric for comparison, and the next experiment contrasts the algorithms in terms of the achievable coverage.

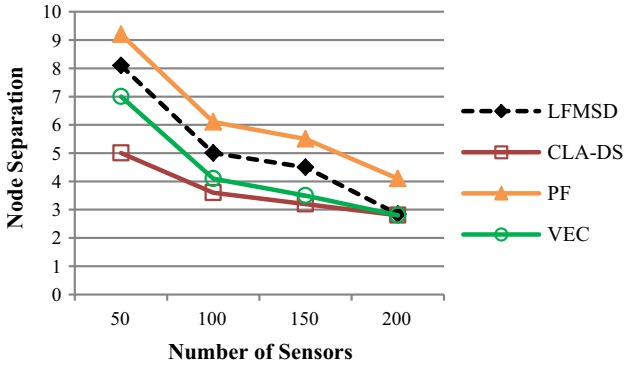

Fig. 15 Node separation within the complex network area

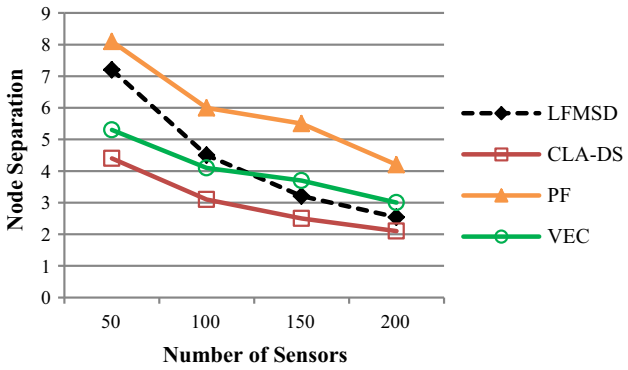

Fig. 16 Node separation within the network area with obstacles

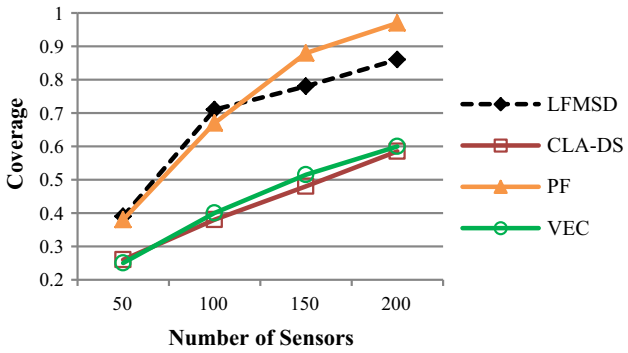

Fig. 17 Coverage performance within the rectangular network area

6.5 Experiment 5

In this experiment, we compare the coverage performance of LFMSD with the other algorithms. We apply the same simulation settings as in Experiment 4. Figures 17, 18 and 19 show the coverage plots for the different areas depicted in Fig. 3. As can be seen, VEC and CLA-DS perform almost similarly in terms of coverage, with LFMSD outperforming both. In the next subsection, we note that LFMSD's higher coverage in comparison with VEC and CLA-DS is obtained at the cost of a higher travelled distance by the sensors.

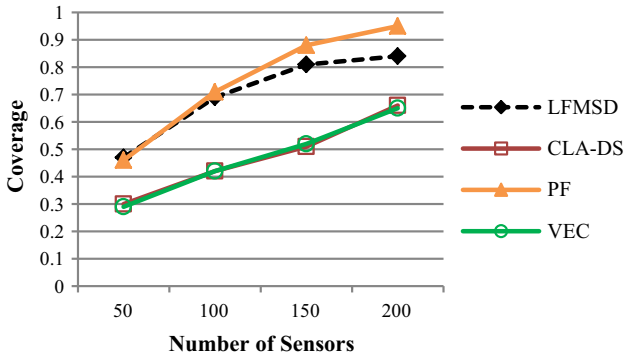

Fig. 18 Coverage performance within the complex-shaped area

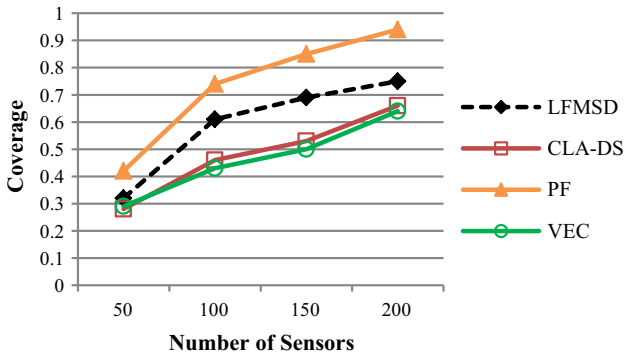

Fig. 19 Coverage performance within the network area with obstacles

It is worth mentioning that VEC's coverage performance, as also pointed out in Wang et al. (2004), is very much dependent upon the initial sensor deployment and the assumed communication range. VEC works best when the initial deployment follows a uniform distribution, while its performance is negatively affected when the sensors are densely positioned in the initial configuration (as is the case in our Experiment 5). In fact, as the cells' Voronoi are very small, the Voronoi regions are covered by the sensors in the central cells in such a way that they will no longer move. Although the sensors at the edge continue moving, but, after just a few steps, their distances to their neighbors reach d_{avg}. Thus, they also stop, and the algorithm effectively halts.

In rectangular and complex-shaped areas, LFMSD, despite being location-free, achieves the same coverage as PF in low-density sensor networks. However, in high-density sensor networks, PF works better than LFMSD. The reason is that in high-density sensor networks, some sensors may be superposed in the NE such that full coverage cannot be achieved. In Fig. 20a, b, we show one final configuration of 150 sensor nodes resultant from running LFMSD and PF, respectively. Overall, this experiment showcases that the proposed learning algorithm is very much effective in positioning the sensors nodes such that the number of overlapping areas is considerably reduced. In the area with obstacles,

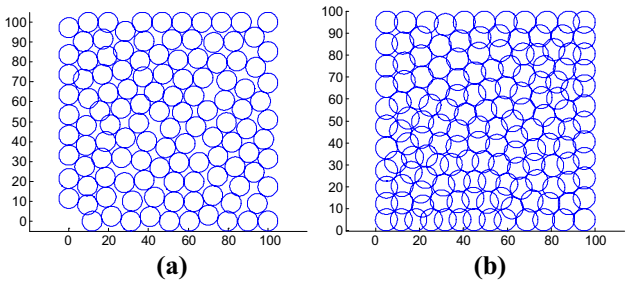

Fig. 20 **a** Final configuration of LFMSD with 150 sensor nodes. **b** Final configuration of PF with 150 sensor nodes

however, the performance of LFMSD decrease, whereas PF outperforms all others owing to its capability in determining obstacles. LFMSD, on the other hand, has no knowledge of the area, and some sensors may get stuck in the boundaries of the obstacles, wasting part of their sensing radius in the process. Despite this deficiency, Fig. 19 shows that LFMSD can compete with the other algorithms which draw on extra information.

6.6 Experiment 6

In this experiment, the LFMSD algorithm is compared with other algorithms in terms of the average distance travelled by each sensor node. We use the same simulation settings as in Experiment 4. According to Figs. 21, 22 and 23 the VEC algorithm turns out to be the most efficient. In comparison with CLA-DS, PF, and VEC, the average distance moved by each sensor in LFMSD is higher. In LFMSD, the movements of a sensor are mostly governed by the number of its neighbors, and hence the distance travelled by each node increases in dense networks. In general, LFMSD has shown a better performance in the complex-shaped area primarily because it has a smaller size, and the nodes travel smaller distances. Also, according to Fig. 23, when there are obstacles in the area, there is an increase in the sensor movements. In fact, since the obstacles bound the sensors in certain directions, exploring the area would need much more locomotion.

In fact, in the absence of location information, the sensors would be oblivious to the small continuous changes made in their physical position as they could only perceive and (possibly) react to the changes in the abstract level (i.e., in the number of 1- and 2-simplices). This way, each sensor is constantly seeking a position where the area under its coverage has minimal overlap with the other sensors. In most experimental scenarios, this behavior has led to an acceptable coverage performance, but as evidenced by the simulations, the proposed scheme turns out to be more costly in terms of the total distance travelled by the sensors. We believe this is a reasonable price to pay for operating under such restricted information settings.

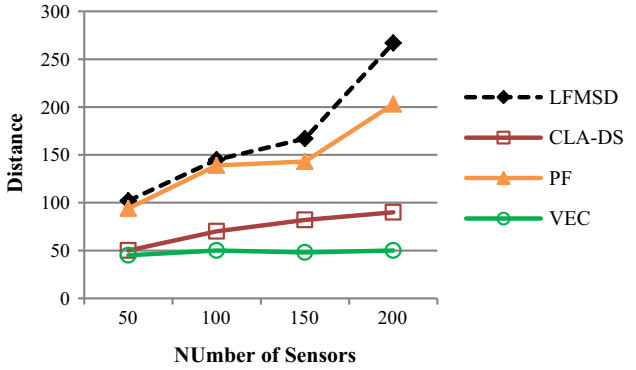

Fig. 21 Average travelled distance within the rectangular network area

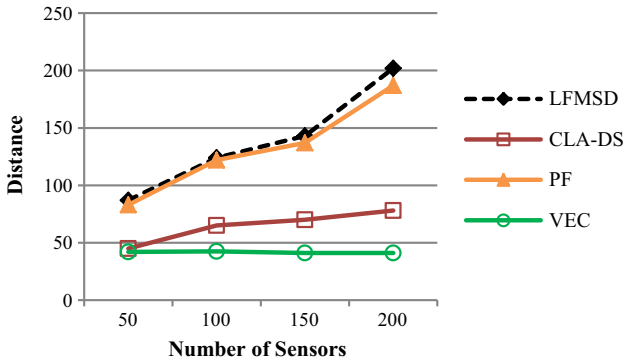

Fig. 22 Average travelled distance within the complex-shaped area

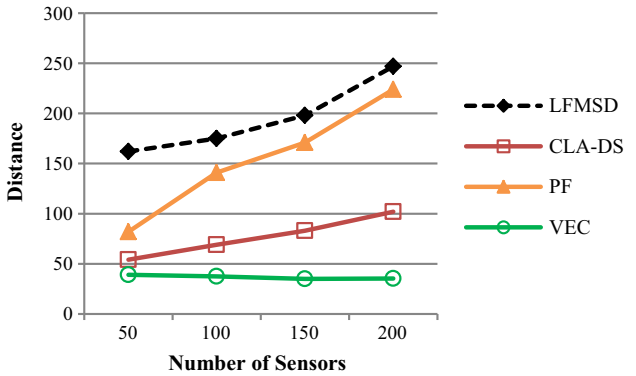

Fig. 23 Average travelled distance within the network area with obstacles

6.7 Experiment 7

In this experiment, we investigate how error in location information can deteriorate the coverage performance in different deployment algorithms. To this end, we assume that rather than the exact locations, only a noise-smeared version of the coordinates is available, as indicated by the following equations:

$$\begin{cases} x_i^{noisy} = x_i + \delta \times Rand_i(x^{\max}) \\ y_i^{noisy} = y_i + \delta \times Rand_i(y^{\max}) \end{cases}$$

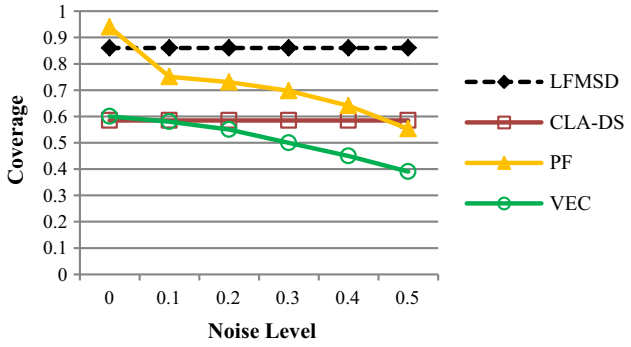

Fig. 24 Coverage performance under noisy location information

where δ is a scaling factor to signify the magnitude of a uniform random noise. In the experiment, we vary the precision of the location information by using different values (from 0.1 to 0.5) for δ. (x_i, y_i) is the exact coordinate of the ith sensor. (x^{max}, y^{max}) corresponds to the farthest location in the network area at which sensor nodes can be located, and $Rand_i(x^{max})$ and $Rand_i(y^{max})$ are random numbers selected uniformly from ranges $[-x^{max}/2, x^{max}/2]$ and $[-y^{max}/2, y^{max}/2]$, respectively.

In this experiment, a total of $N = 200$ sensor nodes are initially deployed within a square region (with side length 10) at the center of the complex-shaped area (depicted in Fig. 3). Figure 24 shows the coverage performance under noisy location information. As expected, LFMSD and CLA-DS are completely robust against the errors in the location information, as they are location-free algorithms with no use for the knowledge of sensor coordinates. The PF algorithm, on the other hand, has the most vulnerability to imprecise location information. In particular, the nodes' movements in PF are governed by Newton's second law of motion, which is in turn determined by the Euclidean distance between the nodes. Also, the movements prescribed in VEC are based on the distance between the sensor nodes; hence, it suffers from the inaccuracy of location information.

6.8 Experiment 8

In this experiment, we study how variations in communication range can affect the achievable coverage. We consider a total of $N = 100$ sensor nodes (with sensing range 5 m), initially deployed within a square region (with side length 10) at the center of the rectangular-shaped area (depicted in Fig. 3). We model the variation in r_b by the following equation: $r_b^{var} = (1 + \delta(t))r_c$ where $\delta(t)$ represents a random perturbation process taking values between 0 and 1. We assume that the probability of occurring a perturbation is P^{var}. We experiment with different values of P^{var} from 0.1 to 0.5. We realize that this is a somehow stylized model of variations in communication range, and more realistic mod-

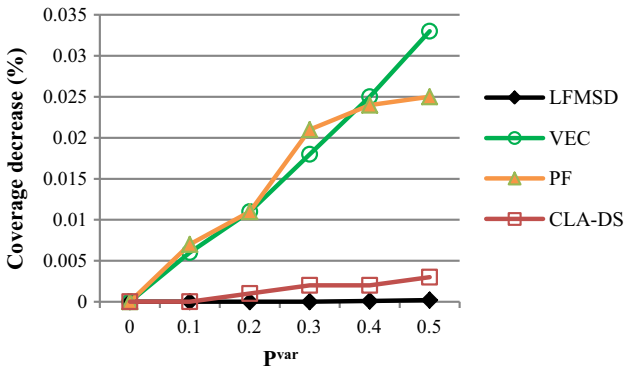

Fig. 25 Coverage decrease versus the variations in communication range

els could also be considered (e.g., based on channel fading). However, for the sake of exposition, this simple model suffices our purposes in this experiment.

Figure 25 shows that LFMSD and CLA-DS are fairly robust against the variations in communication range. Also, if, on average, the communication range approaches its theoretical optimum $\sqrt{3}r_c$, it would be to the benefit of our algorithm (c.f., Assumption 1). VEC, on the other hand, is the most vulnerable algorithm. The reason is that when the communication range is lower than 10 m, some sensors cannot determine their Voronoi neighbors, so the constructed Voronoi diagram is not accurate.

7 Conclusion

In this paper, we proposed a distributed procedure for the automated deployment of a set of mobile sensors in a field in order to maximize the covered area. A great advantage of our design is that it does not use any information about the physical layout of the environment, the number of nodes, their location, and even the distances of the sensors from each other. Instead, we resorted to the recent technique of forming low-cost simplicial complexes to give each sensor an abstract local view of the coverage topology. Given this limited informational assumption, we were motivated to formulate the problem of location-free coverage as a constrained exact potential game. In this game, the utility of each sensor has an inverse relationship with the number of 1-simplices and 2-simplices as a measure of the sensor's total number of overlapping areas with its neighbors. To shape the behavior of the mobile sensors towards an equilibrium deployment, we equipped each sensor with a distributed adaptive learning algorithm (i.e., LFMSD). We rigorously proved that by executing LFMSD, the final positioning of the sensors correspond to a Nash equilibrium. Through simulation experiments, it was demonstrated that LFMSD can compete with location-dependent algorithms.

Appendix A

In this section, we briefly review the concepts of simplicial complexes. In-depth information regarding algebraic topology and simplicial complexes can be found in Hatcher (2002).

Let $\{v_0, \ldots, v_k\}$ be a geometrically independent set in $\mathbb{R}^N$, the k-simplex $[v_0\, v_1 \ldots v_k]$ is the set of all points x of $\mathbb{R}^N$ such that $x = \sum_{i=0}^{k} \lambda_i v_i$, where $\sum_{i=0}^{k} \lambda_i = 1$, and $\lambda_i \geq 0$ for all i. The points $v_0, \ldots, v_k$ are called vertices. Simplex A is a face of simplex B, if the vertices of A form a subset of the vertices of B. A simplicial complex is a finite collection of simplices, K, which are properly joined and have the property that each face of a member of K is also a member of K. An oriented simplicial complex is a simplicial complex with ordering on every k-simplex.

Given a set of points, χ, in a metric space and a fix $\varepsilon > 0$, the Rips complex of χ, $R_\epsilon(\chi)$, is the abstract simplicial complex whose k-simplices correspond to unordered $(k+1)$-tuples of points in χ which are pairwise within distance ε of each other.

Two continuous functions mapping one topological space to another are called homotopic if one can be continuously deformed into the other. Such deformation is called a homotopy between the two functions. Given two spaces X and Y, we say they are homotopy equivalent or of the same homotopy type if there exist continuous maps $f\colon X \to Y$ and $g\colon Y \to X$ such that gof is homotopic to the identity map id_X and fog is homotopic to id_Y.

A set X is contractible if the identity map on X is homotopic to a constant map. In other words, a contractible space is one that can be continuously shrunk to a point.

Theorem A.1 (Cech theorem) (Bott and Tu 1995) *If a collection of sets and all their nonempty finite intersections are contractible, then the union of those sets has the homotopy type as the nerve complex.*

Theorem A.2 (Silva and Ghrist 2006) *The Rips complex of a sensor network with parameter ε, R_ε, is a subcomplex of the nerve complex corresponding to disks of radius $\varepsilon \Big/ \sqrt{3}$ centered at its vertices.*

Therefore, $r_b \leq r_c\sqrt{3}$ leads to $R_{r_b} \subseteq C_{r_c}$.

Appendix B

Definition B.1 Let P^0 be the transition matrix of a time-homogeneous Markov chain $\{P_t^0\}$ on a finite space X, and

P^ε be the transition matrix of the perturbed Markov chain $\{P_t^\varepsilon\}$. $\{P_t^\varepsilon\}$ follows P^0 with probability $1-\varepsilon$, and does not follow P^0 with probability ε. $\{P_t^\varepsilon\}$ is a regular perturbation of $\{P_t^0\}$ if $\forall x, y \in X$:

1. For some $n > 0$, $\forall \varepsilon \in (0, n]$, the Markov chain $\{P_t^\varepsilon\}$ is irreducible and aperiodic
2. $\lim_{\varepsilon \to 0^+} P_{xy}^\varepsilon = P_{xy}^0$
3. If $P_{xy}^\varepsilon > 0$ for some ε, then there exist a real number $r(x \to y) \geq 0$ such that $0 < \lim_{\varepsilon \to 0^+} P_{xy}^\varepsilon \Big/ \varepsilon^{r(x \to y)} < \infty$.
4. $r(x \to y)$ is called the resistance of the transition from x to y.

Theorem B.2 (Young 1993) *Let $\{P_t^\varepsilon\}$ be a regular perturbation of $\{P_t^0\}$, and $\mu(\varepsilon)$ be the unique stationary distribution of $\{P_t^\varepsilon\}$ for each $\varepsilon > 0$. Then $\lim_{\varepsilon \to 0^+} \mu_\varepsilon = \mu_0$ exists, and $\mu(0)$ is a stationary distribution of $\{P_t^0\}$. The stochastically stable states are precisely those states contained in the irreducible classes with minimum stochastic potential.*

Definition B.3 The Markov chain (Yuan et al.) is strongly ergodic if there exist a probability distribution μ^* on X such that for any initial distribution μ_0 on X and any $m \in \mathbb{Z}^+$, $\lim_{k \to +\infty} \mu_0^T P(m, k) = (\mu^*)^T$, where $P(m, k) := \prod_{t=m}^{k-1} P(t)$, $0 \leq m < k$. If (Yuan et al.) is strongly ergodic, then (Yuan et al.) in distribution is convergent (Isaacson and Madsen 1976).

Definition B.4 The Markov chain (Yuan et al.) is weakly ergodic if $\forall x, y, z \in X$ and $\forall m \in \mathbb{Z}^+$, $\lim_{k \to +\infty} \big(P_{xz}(m, k) - P_{yz}(m, k)\big) = 0$

Theorem B.5 (Isaacson and Madsen 1976) *The Markov chain (Yuan et al.) is weakly ergodic if and only if there is a strictly increasing sequence of positive numbers k_i, such that*

$$\sum_{i=0}^{+\infty} (1 - \lambda P(k_i, k_{i+1})) = +\infty,$$

where $\lambda(P) := 1 - \min_{1 \leq i, j \leq n} \sum_{k=1}^n \min\big(P_{ik}, P_{jk}\big)$.

Theorem B.6 (Isaacson and Madsen 1976) *A Markov chain (Yuan et al.) is strongly ergodic if*

1. The Markov chain (Yuan et al.) is weakly ergodic.
2. $\forall t$, there exists a stochastic vector $\mu^{(t)}$ on X such that $\mu^{(t)}$ is the left eigenvector of the transition matrix $P(t)$ with eigenvalue 1.
3. The eigenvector $\mu^{(t)}$ satisfy $\sum_{t=0}^{+\infty} \sum_{z \in X} \left| \mu_z^{(t)} - \mu_z^{(t+1)} \right| < +\infty$.

Appendix C

In this section, we prove the ergodicity properties of the Markov chain underlying the LFMSD learning algorithm. To this end, we first show that the game's Markov chain satisfies the "regular perturbation" property (Lemma C.1). This way, we can be sure that all non-zero entries of P^ε are of the order $O(\varepsilon^r)$ for some $r \geq 0$. In our problem, r corresponds to the number of sensors which choose their move via exploration rather than via better reply; i.e., $r = |\Omega|$. Next (in Lemma C.3), we basically follow the standard methodology based on Theorem B.5 for proving the weak ergodicity of the non-stationary Markov chain $P_t^{\varepsilon(t)}$. Basically, the proof entails decomposing the infinite product $\prod_t P_t^{\varepsilon(t)}$ into blocks of matrices and showing that the sum of the "ergodic coefficients" of all blocks is infinite. More specifically, the ergodic coefficient of any stochastic matrix $\mathbb{P}$ denoted by $erg(\mathbb{P})$ is given by $erg(\mathbb{P}) \stackrel{\text{def}}{=} \min_{ij} \sum_k \min\big(\mathbb{P}_{ik}, \mathbb{P}_{jk}\big)$ and in order to prove that the ergodic coefficients of all product blocks sums up to infinity, we will use our earlier result (Lemma C.1) to show that for any product block $\mathbb{P}$, $\min\big(\mathbb{P}_{ik}, \mathbb{P}_{jk}\big)$ is bounded from below by ε^r.

Lemma C.1 *:$\{P_t^\varepsilon\}$ is a regular perturbation of $\{P_t^0\}$.*

Proof To prove $\{P_t^\varepsilon\}$ is a regular perturbation of $\{P_t^0\}$, we show that conditions 1, 2, and 3 stated in "Appendix B" hold for all $z^1, z^2 \in B$. □

Examining condition (1): Since each sensor s_i in location a_i can stay in its position, and its motion to the position with distance 1 from its current position is allowed, so the reachable set from any $z^0 \in B$ is B. So, the Markov chain $\{P_t^\varepsilon\}$ is irreducible on the space B.

Each sensor can stay in its current location, so any state in diag A has period 1. For any state $z := (a^0, a^1)$ the following two paths are feasible:

$$(a^0, a^1) \to (a^1, a^0) \to (a^0, a^1)$$
$$(a^0, a^1) \to (a^1, a^1) \to (a^1, a^0) \to (a^0, a^1)$$

Therefore, the period of state z is 1. Therefore, $\{P_t^\varepsilon\}$ is aperiodic.

Examining condition (2): It is not difficult to see that $\lim_{\varepsilon \to 0^+} P_{z^1 z^2}^\varepsilon = P_{z^1 z^2}^0$.

Examining condition (3): According to (7), it is obvious that $0 < \lim_{\varepsilon \to 0} \frac{P_{z^1 z^2}^\varepsilon}{\varepsilon^{|\Omega(z_1 \to z_2)|}} < +\infty$. Therefore, the resistance of transition $z^1 \to z^2$ that we show it by $r(z^1 \to z^2)$ is $\Omega(z^1 \to z^2)$.

Therefore, all the conditions required for being regular perturbation of $\{P_t^\varepsilon\}$ are established.

According to (7), it is obvious that $0 < \lim_{\varepsilon(t)\to 0} \frac{P_t^{\varepsilon(t)}(z^1,z^2)}{\varepsilon(t)^{|\Omega(z_1\to z_2)|}} < +\infty$. Therefore, for sufficiently large t,

$$\underline{\alpha}(z^1,z^2)\varepsilon_t^{\Omega(z^1,z^2)} < P_t^{\varepsilon(t)}(z^1,z^2) < \bar{\alpha}(z^1,z^2)\varepsilon_t^{\Omega(z^1,z^2)} \quad (10)$$

Let $\alpha = \min_{x,\, y\in B}\Big(\underline{\alpha}(x,\, y)\Big)$, then

$$P_t^{\varepsilon(t)}(z^1,z^2) > \alpha\varepsilon_t^{r(z^1,z^2)} \quad (11)$$

Lemma C.2 *Stochastically stable states are exactly the states in* $\mathrm{diag}(\Re(\Gamma_{mc}))$.

Proof This Lemma can be proved in the same way as Zhu and Martínez (2013). □

Lemma C.3 $\{P^{\varepsilon(t)}\}$ *is weakly ergodic.*

Proof Let $z* := (a*,\, a*) \in \mathrm{diag}(\Re(\Gamma_{mc}))$ be a stochastically stable state which is in the recurrent class E^*. Since $\{P_t^{\varepsilon(t)}\}$ is irreducible, there is a path from any state z to z^*. Let $z \Rightarrow z*$ be a path from z to z^* in markov chain $\{P_t^{\varepsilon(t)}\}$, and $NT(z \Rightarrow z*)$ be the minimum number of transitions in path $z \Rightarrow z*$. Let $h = \max_z \min_{z\Rightarrow z*} (NT(z \Rightarrow z*))$. Since there is a transition from z^* to itself, there is a path from each state z to z^* with length h. Consider path $z \to z^1 \to \cdots \to z^h = z^*$. According to (11) $P_t^{\varepsilon}(z^i,\, z^{i+1}) > \alpha\varepsilon^{\Omega(z^i,z^{i+1})}$, where z^i and z^{i+1} are two consecutive states in the mentioned path. Hence,

$$P_{(t,t+h)}^{\varepsilon(t)}(z,\, z^*) \geq P_t^{\varepsilon(t)}(z,\, z_1)P_{t+1}^{\varepsilon(t+1)}(z_1,\, z_2)\cdots$$
$$P_{t+h-1}^{\varepsilon(t+h-1)}(z_{h-1},\, z^*) > \alpha^h\varepsilon_t^{\Omega(z,z_1)}\varepsilon_{t+1}^{\Omega(z_1,z_2)}\cdots\varepsilon_{t+h-1}^{\Omega(z_{h-1},z*)}$$
$$> \alpha^h\varepsilon_{t+h-1}^{\sum_{i=0}^{h-1}\Omega(z_i,z_{i+1})} \geq \alpha^h\varepsilon_{t+h-1}^{\rho}$$

for all $z \in B$, where $P_{(m,l)} := \prod_{t=m}^{l-1} P_t$, $0 \leq m < l$. Consequently, by choosing a subsequence such that $k_{n+1} - k_n = h$, for sufficiently large n, it holds that

$$P_{(k_n,k_{n+1})}^{\varepsilon(t)}(z,\, z*) > \alpha^h\varepsilon_{k_{n+1}}^{\rho} \quad \forall z \in B$$

Therefore,

$$\min\{P_{(k_n,k_{n+1})}^{\varepsilon(t)}(x,\, z*),\, P_{(k_n,k_{n+1})}^{\varepsilon(t)}(y,\, z*)\} > \alpha^h\varepsilon_{k_{n+1}}^{\rho} \quad \forall x,\, y \in B$$
$$\sum_{z\in B}\min_{x,\,y}\{P_{(k_n,k_{n+1})}^{\varepsilon(t)}(x,\, z),\, P_{(k_n,k_{n+1})}^{\varepsilon(t)}(y,\, z)\} > \alpha^h\varepsilon_{k_{n+1}}^{\rho}$$
$$\Rightarrow \min\sum_{z\in B}\min_{x,\,y}\{P_{(k_n,k_{n+1})}^{\varepsilon(t)}(x,\, z),\, P_{(k_n,k_{n+1})}^{\varepsilon(t)}(y,\, z)\} > \alpha^h\varepsilon_{k_{n+1}}^{\rho}$$
$$\Rightarrow \sum_{n=0}^{+\infty}\min\sum_{z\in B}\min_{x,\,y}\{P_{(k_n,k_{n+1})}^{\varepsilon(t)}(x,\, z),\, P_{(k_n,k_{n+1})}^{\varepsilon(t)}(y,\, z)\}$$
$$> \alpha^h\sum_{n=0}^{+\infty}\varepsilon_{k_{n+1}}^{\rho} = \alpha^h\sum_{n=0}^{+\infty}\big(0.1\big/\log(k_{n+1})\big)^{\rho} = +\infty$$

Therefore, according to Theorem B.5 $\{P_t^{\varepsilon(t)}\}$ is weakly ergodic if $\sum_{n=0}^{+\infty}\varepsilon_{k_{n+1}}^{\rho} = +\infty$. For example $\varepsilon(t) = \big(1\big/t\big)^{\frac{1}{\rho}}$ is one choice. □

Lemma C.4 $:\{P^{\varepsilon(t)}\}$ *is strongly ergodic.*

Proof To prove the strong ergodicity of $\{P^{\varepsilon(t)}\}$ we use Theorem 2 of Anily and Federgruen (1987) and show that the weakly ergodic markov chain $\{P_t^{\varepsilon(t)}\}$ is strongly ergodic. According to this theorem, first, we construct an extension $\bar{\varepsilon}(x)$ of the sequence $\varepsilon(t)$; second, we construct a regular extension $\bar{P}^{\bar{\varepsilon}(x)}(x)$ of $\{P_t^{\varepsilon}\}$; third, we show that all entries of the regular extension $\bar{P}^{\bar{\varepsilon}(x)}(x)$ are members of a closed class of asymptotically monotone functions.

Definition C.5 Let $\{f(t)\}$ be a sequence with $f(t) \in R^m$. The (vector) function $\bar{f}(x) : (0,\, 1] \to R^m$ is an extension of the sequence if $\bar{f}(x_t) = f(t)$ for some sequence $\{x_t\}$ with $\lim_{t\to\infty} x(t) = 0$.

To construct an extension $\bar{\varepsilon}(x)$ of the sequence $\varepsilon(t)$, for all $i \in N$, we define $\bar{\varepsilon}_i(x)$ as $\bar{\varepsilon}_i(x) = 0.1x$, and $x_t = 1\big/\log(t+1)$. Obviously, $\bar{\varepsilon}_i(x) = \frac{0.1}{\log(t+1)} = \varepsilon_i(t)$.

Definition C.6 Let $\bar{P}(.)$ be an extension of a nonstationary Markov chain $\{P_t^{\varepsilon(t)}\}$. $\bar{P}(.)$ is a regular extension of $\{P_t^{\varepsilon(t)}\}$ if a positive real number x^* exists such that the collection of subchains of $\bar{P}(x)$ is identical for all $x < x*$.

Let $\bar{P}^{\bar{\varepsilon}_i(x)}(x)$ be an extension to the Markov chain $\{P_t^{\varepsilon(t)}\}$. According to (7), $\overline{P}^{\bar{\varepsilon}(x)}(z,\, z')$ is positive, so considering $x^*=1$, it is obvious that the set of transitions with strictly positive probabilities is identical for all $x < x^*$. Now, we should show that every entry function in $\bar{P}^{\bar{\varepsilon}_i(x)}(x)$ belongs to a closed class of asymptotically monotone functions F.

Definition C.7 A class $F \subset C^1$ of functions defined on $(0,\, 1]$ is CAM (closed class of asymptotically monotone functions) if

1. $f \in F \Rightarrow f' \in F$ and $-f \in F$
2. $f,\, g \in F \Rightarrow (f+g) \in F$ and $(f.g) \in F$
3. All $f \in F$ change signs finitely often in on $(0,\, 1]$.

Let F be the class of functions that each $f \in F$ is of the form $\sum_{k=1}^{K} c_k(\log(x+1))^{b_k}V_k(x)^{g_k}$ where c_k is a real number, and b_k and g_k are integer numbers. It is not difficult to check that the class F is a closed class of asymptotically monotone functions. According to (7), all elements of the transition matrix P_t^{ε} are functions in the class F. Therefore, $\{P_t^{\varepsilon}\}$ is strongly ergodic. □

Proof of Theorem 2 According to lemma C.4, $\{P_t^{\varepsilon}\}$ is strongly ergodic. Thus, according to theorem B.2 the limiting distribution is $\mu^* = \lim_{t\to+\infty}\mu^{(t)}$. In addition, we

can prove that the stochastically stable states of $\{P_t^{\varepsilon}\}$ are contained in the set $\mathrm{diag}(\Re(\Gamma_{mc}))$ in the same way as Zhu and Martínez (2013). Therefore, the support of μ^* is contained in $\mathrm{diag}(\Re(\Gamma_{mc}))$, that corresponds to $\lim_{t\to+\infty} p(z(t) \in diag\Re(\Gamma_{mc})) = 1$. □

References

Anily, S., & Federgruen, A. (1987). Ergodicity in parametric nonstationary Markov chains: An application to simulated annealing methods. *Operations Research, 35*(6), 867–874.

Arslan, G., Marden, J. R., & Shamma, J. S. (2007). Autonomous vehicle-target assignment: A game-theoretical formulation. *Journal of Dynamic Systems, Measurement, and Control, 129*(5), 584–596.

Bartolini, N., Calamoneri, T., La Porta, T., & Silvestri, S. (2010). Mobile sensor deployment in unknown fields. In *INFOCOM, 2010 Proceedings IEEE, 2010* (pp. 1–5). IEEE.

Bott, R., & Tu, L. (1995). *Differential forms in algebraic topology*. New York: Springer.

Casteigts, A., Albert, J., Chaumette, S., Nayak, A., & Stojmenovic, I. (2012). Biconnecting a network of mobile robots using virtual angular forces. *Computer Communications, 35*(9), 1038–1046.

Chellappan, S., Bai, X., Ma, B., Xuan, D., & Xu, C. (2007). Mobility limited flip-based sensor networks deployment. *IEEE Transactions on Parallel and Distributed Systems, 18*(2), 199–211.

Cortes, J., Martinez, S., Karatas, T., & Bullo, F. (2004). Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation, 20*(2), 243–255. https://doi.org/10.1109/tra.2004.824698.

Esnaashari, M., & Meybodi, M. R. (2011). A cellular learning automata-based deployment strategy for mobile wireless sensor networks. *Journal of Parallel and Distributed Computing, 71*(7), 988–1001. https://doi.org/10.1016/j.jpdc.2010.10.015.

Falcon, R., Li, X., Nayak, A., & Stojmenovic, I. (2010). The one-commodity traveling salesman problem with selective pickup and delivery: An ant colony approach. In *2010 IEEE congress on evolutionary computation (CEC), 2010* (pp. 1–8). IEEE.

Flåm, S. D. (2002). Equilibrium, evolutionary stability and gradient dynamics. *International Game Theory Review, 4*(04), 357–370.

Fletcher, G., Li, X., Nayak, A., & Stojmenovic, I. (2010a). Backtracking based sensor deployment by a robot team. In *2010 7th annual IEEE communications society conference on sensor mesh and ad hoc communications and networks (SECON), 2010a* (pp. 1–9). IEEE.

Fletcher, G., Li, X., Nayak, A., & Stojmenovic, I. (2010b). Randomized robot-assisted relocation of sensors for coverage repair in wireless sensor networks. In *2010 IEEE 72nd vehicular technology conference fall (VTC 2010-Fall), 2010b* (pp. 1–5). IEEE.

Fudenberg, D., & Tirole, J. (1991). *Game theory* (Vol. 1). Cambridge: MIT Press Books.

Hatcher, A. (2002). *Algebraic topology*. Cambridge: Cambridge Univ. Press.

Heo, N., & Varshney, P. K. (2005). Energy-efficient deployment of intelligent mobile sensor networks. *IEEE Transactions on Systems, Man, and Cybernetics, 35*(1), 78–92.

Howard, A., Mataric, M. J., & Sukhatme, G. S. (2002). Mobile sensor network deployment using potential fields: A distributed scalable solution to the area coverage problem. In *International symposium on distributed autonomous robotics systems, Fukuoka, Japan, 2002*.

Isaacson, D. L., & Madsen, R. W. (1976). *Markov chains, theory and applications* (Vol. 4). New York: Wiley.

Izadi, D., Abawajy, J., & Ghanavati, S. (2015). An alternative node deployment scheme for WSNs. *IEEE Sensors Journal, 15*(2), 667–675. https://doi.org/10.1109/jsen.2014.2351405.

Jiang, Z., Wu, J., Kline, R., & Krantz, J. (2008) Mobility control for complete coverage in wireless sensor networks. In *28th international conference on distributed computing systems workshops, 2008. ICDCS'08.* (pp. 291–296). IEEE.

Li, J., Zhang, B., Cui, L., & Chai, S. (2012). An extended virtual force-based approach to distributed self-deployment in mobile sensor networks. *International Journal of Distributed Sensor Networks, 8*(3), 1–14.

Li, N., & Marden, J. R. (2013). Designing games for distributed optimization. *IEEE Journal of Selected Topics in Signal Processing, 7*(2), 230–242.

Luo, R. C., & Chen, O. (2012). Mobile sensor node deployment and asynchronous power management for wireless sensor networks. *IEEE Transactions on Industrial Electronics, 59*(5), 2377–2385. https://doi.org/10.1109/tie.2011.2167889.

Marden, J. R., Arslan, G., & Shamma, J. S. (2009a). Cooperative control and potential games. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 39*(6), 1393–1407.

Marden, J. R., Young, H. P., Arslan, G., & Shamma, J. S. (2009b). Payoff-based dynamics for multiplayer weakly acyclic games. *SIAM Journal on Control and Optimization, 48*(1), 373–396.

Meybodi, M. R., Beigy, H., & Taherkhani, M. (2003). Cellular learning automata and its applications. *Sharif Journal of Science and Technology, 19*(25), 54–77.

Monderer, D., & Shapley, L. S. (1996). Potential games. *Games and economic Behavior, 14*(1), 124–143.

Muhammad, A., & Jadbabaie, A. (2007) Decentralized computation of homology groups in networks by gossip. In *American control conference, ACC '07, 9–13 July 2007* (pp. 3438–3443).

Nash, J. (1951). Non-cooperative games. *Annals of Mathematics*, *54*(2), 286–295.

Rawat, P., Singh, K. D., Chaouchi, H., & Bonnin, J. M. (2014). Wireless sensor networks: A survey on recent developments and potential synergies. *The Journal of supercomputing, 68*(1), 1–48.

Sahoo, P. K., Tsai, J.-Z., & Ke, H.-L. (2010) Vector method based coverage hole recovery in wireless sensor networks. In *2010 2nd international conference on communication systems and networks (COMSNETS), 2010* (pp. 1–9). IEEE.

Sibley, G. T., Rahimi, M. H., & Sukhatme, G. S. (2002) Robomote: A tiny mobile robot platform for large-scale ad-hoc sensor networks. In *IEEE international conference on robotics and automation, 2002. Proceedings. ICRA'02* (Vol. 2, pp. 1143–1148). IEEE.

Silva, V. D., & Ghrist, R. (2006). Coordinate-free coverage in sensor networks with controlled boundaries via homology. *International Journal of Robotics Research, 25*(12), 1205–1222. https://doi.org/10.1177/0278364906072252.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction* (Vol. 1). Cambridge: MIT Press.

Tamboli, N., & Younis, M. (2010). Coverage-aware connectivity restoration in mobile sensor networks. *Journal of Network and Computer Applications, 33*(4), 363–374.

Tan, G., Jarvis, S. A., & Kermarrec, A.-M. (2009). Connectivity-guaranteed and obstacle-adaptive deployment schemes for mobile sensor networks. *IEEE Transactions on Mobile Computing, 8*(6), 836–848.

Varposhti, M., Dehghan, M., & Safabakhsh, R. (2012) Camera selection without location information: A topological approach. In *IEEE symposium on computers and communications, ISCC'12, Cappadocia, Turkey, 2012* (pp. 376–381). IEEE.

Varposhti, M., Dehghan, M., & Safabakhsh, R. (2014). Distributed topological camera selection without location information. *IEEE Sensors Journal, 14*(8), 2579–2589. https://doi.org/10.1109/jsen.2014.2309797.

Varposhti, M., Dehghan, M., & Safabakhsh, R. (2015) A Distributed homological approach to location-independent area coverage in wireless sensor networks. *Wireless Personal Communications*, *83*(4), 3075–3089.

Wang, G., Cao, G., & La Porta, T. (2004) Movement-assisted sensor deployment. In *INFOCOM 2004. 23rd annual joint conference of the IEEE computer and communications societies, 7–11 March 2004* (Vol. 4, pp. 2469–2479, Vol. 2464). https://doi.org/10.1109/infcom.2004.1354668.

Wang, G., Cao, G., & La Porta, T. (2006). Movement-assisted sensor deployment. *IEEE Transactions on Mobile Computing, 5*(6), 640–652.

Wang, G., Cao, G., La Porta, T., & Zhang, W. (2005) Sensor relocation in mobile sensor networks. In *Proceedings of the 24th annual joint conference of the IEEE computer and communications societies, INFOCOM, 13–17 March 2005* (Vol. 4, pp. 2302–2312).

Wang, B., Lim, H. B., & Ma, D. (2009). A survey of movement strategies for improving network coverage in wireless sensor networks. *Computer Communications, 32*(13–14), 1427–1436. https://doi.org/10.1016/j.comcom.2009.05.004.

Wang, Y.-C., Hu, C.-C., & Tseng, Y.-C. (2008). Efficient placement and dispatch of sensors in a wireless sensor network. *IEEE Transactions on Mobile Computing, 7*(2), 262–274. https://doi.org/10.1109/tmc.2007.70708.

Wu, J., & Yang, S. (2007). Optimal movement-assisted sensor deployment and its extensions in wireless sensor networks. *Simulation Modelling Practice and Theory, 15*(4), 383–399. https://doi.org/10.1016/j.simpat.2006.11.006.

Yan, F., Martins, P., & Decreusefond, L. (2014). Accuracy of homology based coverage hole detection for wireless sensor networks on sphere. *IEEE Transactions on Wireless Communications, 13*(7), 3583–3595. https://doi.org/10.1109/twc.2014.2314106.

Yang, S., Li, M., & Wu, J. (2007). Scan-based movement-assisted sensor deployment methods in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems, 18*(8), 1108–1121.

Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. *Computer Networks, 52*(12), 2292–2330.

You-Chiun, W., & Yu-Chee, T. (2008). Distributed deployment schemes for mobile wireless sensor networks to ensure multilevel coverage. *IEEE Transactions on Parallel and Distributed Systems, 19*(9), 1280–1294. https://doi.org/10.1109/tpds.2007.70808.

Young, H. P. (1993). The evolution of conventions. *Econometrica: Journal of the Econometric Society*, *61*(1), 57–84.

Yuan, S., Bing, W., Zhijie, S., Pattipati, K. R., & Gupta, S. (2014). Distributed algorithms for energy-efficient even self-deployment in mobile sensor networks. *IEEE Transactions on Mobile Computing, 13*(5), 1035–1047. https://doi.org/10.1109/tmc.2013.46.

Zhu, C., Zheng, C., Shu, L., & Han, G. (2012). A survey on coverage and connectivity issues in wireless sensor networks. *Journal of Network and Computer Applications, 35*(2), 619–632. https://doi.org/10.1016/j.jnca.2011.11.016.

Zhu, M., & Martínez, S. (2013). Distributed coverage games for energy-aware mobile sensor networks. *SIAM Journal on Control and Optimization, 51*(1), 1–27.

Zou, Y., & Chakrabarty, K. (2003). Sensor deployment and target localization based on virtual forces. In *22nd annual joint conference of the IEEE computer and communications, INFOCOM 2003, 30 March–3 April 2003* (Vol. 2, pp. 1293–1303).

Marzieh Varposhti received her B.Sc. and M.Sc. from Isfahan University, Isfahan, Iran in 2005 and 2008 respectively, and Ph.D. from Amirkabir University of Technology (AUT), Tehran, Iran in 2015. She is currently an assistant professor of computer engineering at Azad university. Her research interests are in wireless networks and sensor networks.

Vesal Hakami received his B.Sc. degree in Computer Engineering (Software) and his M.Sc. and Ph.D. degrees in Information Technology (Computer Networks), all from Amirkabir University of Technology (AUT), Tehran, Iran, in 2004, 2008 and 2015, respectively. Following graduation, he has served as a research consultant in Iran Telecommunications Research Center (ITRC), working on standardization issues for future wireless networks. In 2016, he joined as an Assistant Professor to the Department of Computer Engineering, Iran University of Science and Technology (IUST), Tehran, Iran. His current research mainly focuses on cognitive control of computer networks using stochastic control theory, and game-theoretic learning.

Mehdi Dehghan received his B.Sc. in Computer engineering from Iran University of Science and Technology (IUST), Tehran, Iran in 1992, and his M.Sc. and Ph.D. from Amirkabir University of Technology (AUT), Tehran, Iran in 1995, and 2001, respectively. Since 1995, he has been a research scientist at Iran Telecommunication Research Center (ITRC) working in the area of Quality of Service provisioning and Network Management. He joined Computer Engineering Department of Amirkabir University of Technology in 2004.