2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Self-Deployment Algorithms for Field Coverage in a Network of Nonidentical Mobile Sensors: Vertex-Based Approach

Hamid Mahboubi, Kaveh Moezzi, Amir G. Aghdam and Kamran Sayrafian-Pour

Abstract— **In this paper, efficient deployment algorithms are proposed for a mobile sensor network to enlarge the coverage area. The proposed algorithms calculate the position of the sensors iteratively based on existing coverage holes in the field. To this end, the multiplicatively weighted Voronoi (MW-Voronoi) diagram is used for a network of mobile sensors with different sensing ranges. Under the proposed procedures, the sensors move in such a way that the coverage holes in the network are reduced. Simulation results are provided to demonstrate the effectiveness of the deployment schemes proposed in this paper.**

I. INTRODUCTION

Wireless sensor networks have attracted considerable research interest in the past decade, and have found a broad range of applications in various areas [1], [2], [3]. Recent advances in MEMS (micro-electro-mechanical systems) technology have enabled the design and manufacturing of small and low-cost sensor nodes. Examples of sensor network applications include biomedical engineering, security surveillance, tracking vehicles and environmental monitoring, to name only a few [4], [5], [6], [7]. In this type of system, it is desired to improve network coverage with limited use of resources [8], [9]. An efficient deployment algorithm should take important practical constraints such as limited communication and sensing ranges into account. Furthermore, it is more desirable to have a decentralized decision-making strategy, due to the distributed configuration of the network [9], [10], [11].

In [12], [13], a mobile sensor deployment strategy is introduced to increase network coverage. In [14], distributed control laws are presented to achieve convex equi-partition configuration in mobile sensor networks. An efficient procedure is introduced in [15] to move the sensors in such a way that the maximum error variance and extended prediction variance are minimized. Distributed control laws are provided in [16] for the disk-covering and sphere-packing problems using non-smooth gradient flows. An algorithm is proposed in [17] to monitor an environmental boundary with mobile agents, where the boundary is optimally approximated with a polygon. Decentralized control laws for optimal placement of sensors are presented in [6] for target tracking. A coverage strategy is proposed in [18] based on a localized Voronoi diagram, where each sensor uses the local information of the neighboring sensors to construct its Voronoi region. In [19], a real-time coverage map is provided to compute the position of the sensors accordingly. Most of the existing works in mobile sensor networks (including the above-mentioned papers) assume that all sensors have the same communication and also sensing ranges. While this can significantly simplify the deployment algorithm design, it may not be a realistic assumption in many real-world applications.

In this paper, two distributed deployment algorithms are presented for a network of nonidentical sensors. The multiplicatively weighted Voronoi (MW-Voronoi) diagram is employed to find the coverage holes, where the weight assigned to each sensor is proportional to its sensing radius [20], [21]. Two algorithms are proposed in this work: Maxmin-vertex and Minmax-vertex. Both algorithms are vertex-based in the sense that they use the distance of each sensor from the vertices of its corresponding MW-Voronoi region to calculate the new destination point for the sensor. The algorithms are decentralized, and sensor placement is performed iteratively. Once each destination is computed, new local coverage area of the corresponding sensor (in the previously constructed MW-Voronoi region) is compared to its preceding local coverage area. If the new local coverage area is larger than the preceding one, the sensor moves to the new destination; otherwise, it remains in its current position. If the increase in the local coverage area of each sensor in an iteration does not exceed a certain threshold, the algorithm is terminated (to ensure a finite number of iterations).

The rest of the paper is organized as follows. In Section II, some preliminaries and important notions and definitions are presented. Section III provides the new algorithms for efficient network coverage, as the main contribution of the paper. Simulations are given in Section IV, and finally the concluding remarks are summarized in Section V.

II. BACKGROUND

Let $\mathbf{S}$ be a set of n distinct weighted nodes in the plane denoted by $(S_1, w_1), (S_2, w_2), \ldots, (S_n, w_n)$, where $w_i > 0$ is the weighting factor associated with S_i, for any $i \in \mathbf{n} := \{1, 2, \ldots, n\}$. Partition the plane into n regions such that:

- Each region contains only one node, and
- the nearest node, in the sense of weighted distance, to any point inside a region is the node assigned to that region.

H. Mahboubi, K. Moezzi and A. G. Aghdam are with the Department of Electrical & Computer Engineering, Concordia University, 1455 de Maisonneuve Blvd. W., EV005.139, Montréal, Québec H3G 1M8 Canada, {h_mahbo,k_moezz,aghdam}@ece.concordia.ca

K. Sayrafian-Pour is with the National Institute of Standards and Technology (NIST), 100 Bureau Drive, Stops 8920 and 8910 Gaithersburg, MD 20899 USA {ksayrafian}@nist.gov

This work has been supported by the National Institute of Standards and Technology (NIST), under grant 70NANB8H8146.

The diagram obtained by the partitioning described above is called the *multiplicatively weighted Voronoi diagram* (MW-Voronoi diagram) [21]. Analogous to conventional Voronoi diagram, the mathematical characterization of each region obtained by the above partitioning is as follows:

$$\bar{\Pi}_i = \{Q \in \mathbb{R}^2 \mid w_j d(Q, S_i) \leq w_i d(Q, S_j), \forall j \in \mathbf{n} - \{i\}\} \tag{1}$$

for any $i \in \mathbf{n}$, where $d(Q, S_i)$ is the Euclidean distance between Q and S_i.

According to (1), any point Q in the i-th MW-Voronoi region $\bar{\Pi}_i$ has the following property:

$$\frac{d(Q, S_i)}{d(Q, S_j)} \leq \frac{w_i}{w_j}, \quad \forall i \in \mathbf{n}, \ \forall j \in \mathbf{n} - \{i\} \tag{2}$$

Definition 1. Similar to conventional Voronoi diagram, the nodes S_i and S_j ($i, j \in \mathbf{n}, i \neq j$) in an MW-Voronoi diagram are called neighbors if $\bar{\Pi}_i \cap \bar{\Pi}_j \neq \emptyset$. The set of all neighbors of S_i, $i \in \mathbf{n}$, is denoted by $\mathbf{N}_i$ and is formulated below:

$$\mathbf{N}_i = \{S_j \in \mathbf{S} \mid \bar{\Pi}_i \cap \bar{\Pi}_j \neq \emptyset, \ \forall j \in \mathbf{n}\} \tag{3}$$

Definition 2. Consider a sensor S_i with the sensing radius r_i and the corresponding MW-Voronoi region $\bar{\Pi}_i$, $i \in \mathbf{n}$, and let Q be an arbitrary point inside $\bar{\Pi}_i$. The intersection of the region $\bar{\Pi}_i$ and a circle of radius r_i centered at Q is referred to as the *coverage area with respect to (w.r.t.) Q*. The coverage area w.r.t. the location of the sensor S_i is called the *local coverage area* of that sensor.

Definition 3. The *Apollonian circle* of the segment AB, denoted by $\Omega_{AB,k}$, is the locus of all points E such that $\frac{AE}{BE} = k$ [22].

To construct the i-th MW-Voronoi region, first the Apollonian circles of the neighboring partitions are found for the i-th sensor. In other words, the Apollonian circles $\Omega_{S_iS_j,\frac{w_i}{w_j}}$ are found for all $S_j \in \mathbf{N}_i$. The smallest region (created by the above circles) containing the i-th node is, in fact, the i-th MW-Voronoi region (e.g., see Fig. 1). An example of a MW-Voronoi diagram with 15 sensors is sketched in Fig. 2.

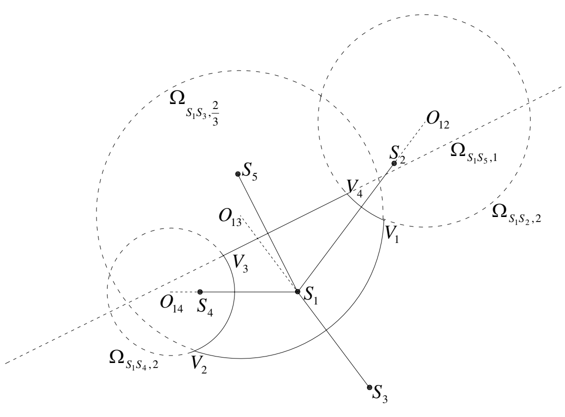


Fig. 1. The MW-Voronoi region for a sensor S_1 with four neighbors $S_2, \ldots, S_5$ [3].

The MW-Voronoi diagram is the main tool for sensor deployment in this paper. Each sensor has a sensing area which is a circle whose size can be different for distinct sensors. Let each sensor in the field be denoted by a node with a weight equal to its sensing radius, and sketch the MW-Voronoi region for each sensor. From the characterization of the MW-Voronoi regions provided in (1), it is straightforward to show that if a sensor cannot detect a phenomenon in its corresponding region, no other sensor can detect it either. This means that in order to find the "so-called" coverage holes (i.e., the undetectable points in the network), it would suffice to compare the MW-Voronoi region of each node with its local coverage area.

Notation 1. Consider a circle of radius r centered at O, denoted by $\Omega(O, r)$ hereafter, and a point V in the plane. The intersections of Ω and the extension of VO from O is denoted by $T^V_{\Omega(O,r)}$. The other intersection point of $\Omega(O, r)$ and VO (or its extension) is denoted by $\bar{T}^V_{\Omega(O,r)}$.

Notation 2. As mentioned before, the boundary curves of an MW-Voronoi region are the segments of some Apollonian circles. The set of all such Apollonian circles for the i-th MW-Voronoi region is denoted by $\mathbf{\Omega}_i$. The sets $\bar{\mathbf{\Omega}}_i$ and $\tilde{\mathbf{\Omega}}_i$ are then defined as follows:

$$\bar{\mathbf{\Omega}}_i = \{\Omega \in \mathbf{\Omega}_i | S_i \in \Omega\}$$

$$\tilde{\mathbf{\Omega}}_i = \{\Omega \in \mathbf{\Omega}_i | S_i \notin \Omega\}$$

Assumption 1. In this paper, it is assumed that there is no obstacle in the field. Therefore, the sensors can move to any desired location without obstacle avoidance concerns using existing techniques, e.g. [9], [23], [24], [25].

Assumption 2. The sensors are supposed to be capable of localizing themselves with sufficient accuracy in the field (using, for instance, the methods proposed in [1], [26]).

Assumption 3. The communication range of the sensors is bounded (and not necessarily the same for all sensors). This is a limiting factor for the sensors, potentially preventing them from reaching their neighbors, which can result in incorrect Voronoi regions around some of the sensors. Consequently, such a limitation can negatively affect the detection of coverage holes.

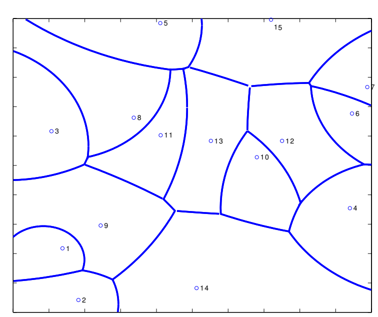


Fig. 2. An example of the MW-Voronoi diagram for a group of 15 nonidentical sensors in a network.

III. Deployment Protocols

In this section, two different protocols are developed for a distributed sensor network. The proposed algorithms are iterative, where in each iteration every sensor first broadcasts its sensing radius and location to other sensors, and then constructs its MW-Voronoi region based on the received information. It checks the region subsequently to detect the possible coverage holes. If any coverage hole exists, the sensor calculates its target location (but does not move there) in such a way that by moving there the coverage hole would be eliminated, or at least its size would be reduced by a certain threshold. Once the new target location is calculated, the coverage area w.r.t. this location (in the previously constructed MW-Voronoi region) is obtained. If this coverage area is greater than the current one, the sensor moves to the new location; otherwise it remains in its current position. In order to terminate the algorithm in finite time, a proper threshold ϵ is defined such that if the increase in the local coverage area by each sensor is not sufficiently large (as specified by ϵ), there is no need to continue the iterations.

Notation 3. In the remainder of this paper, $\mathcal{V}$ represents an MW-Voronoi diagram with n regions (each one corresponding to a node). Furthermore, the number of vertices of the i-th region is denoted by m_i, for any $i \in \mathbf{n}$.

Definition 4. The corner points of the i-th MW-Voronoi region (i.e., the intersection points of its boundary curves) are denoted by $\mathbf{V}_i = \{V_{i1}, V_{i2}, \ldots, V_{il}\}$. These points are called the MW-Voronoi vertices for the i-th region.

A. Maxmin-Vertex Strategy

The main idea behind this strategy is that normally for ideal network coverage, none of the sensors should be too close to any of the vertices of its Voronoi region. The Maxmin-vertex strategy selects the destination for each sensor as a point inside the corresponding MW-Voronoi region whose distance from the nearest vertex is maximized. This point will be referred to as the *Maxmin-vertex centroid*, and will be denoted by $\bar{O}_i$ for the i-th MW-Voronoi region, $i \in \mathbf{n}$. Let the distance between this point and the nearest vertex on the i-th region to it be represented by $\bar{r}_i$. Denote with $C(O_i, r_i)$ a circle of radius r_i centered at the point O_i. The Maxmin-vertex circle is defined next.

Definition 5. The Maxmin-vertex circle of a region in the MW-Voronoi diagram $\mathcal{V}$ is defined as the largest circle centered inside that region such that all of the vertices of the region are either outside the circle, or on it. This circle is, in fact, $C(\bar{O}_i, \bar{r}_i)$ for the i-th region, $i \in \mathbf{n}$.

Remark 1. If an MW-Voronoi region has exactly one boundary curve, then this curve is a circle which is also the Maxmin-vertex circle in the Maxmin-vertex strategy.

A number of lemmas and theorems are presented next, but the proofs are omitted due to space restrictions.

Lemma 1. *Suppose the i-th region ($i \in \mathbf{n}$) of the MW-Voronoi diagram $\mathcal{V}$ has more than one boundary curve. If the Maxmin-vertex circle passes through exactly one vertex, say V_{i1}, then $\bar{O}_i$ is $T_{\Omega}^{V_{i1}}$ for some $\Omega \in \mathbf{\Omega}_i$; otherwise, the Maxmin-vertex circle passes through at least two vertices.*

Lemma 2. *Consider an MW-Voronoi diagram $\mathcal{V}$, and assume that the Maxmin-vertex circle of one of the regions, say region i ($i \in \mathbf{n}$) passes through exactly two vertices, say $\bar{V}_{i1}$ and $\bar{V}_{i2}$. Then $\bar{O}_i$ is the intersection point of the perpendicular bisector of $\bar{V}_{i1}\bar{V}_{i2}$ and the boundary of the i-th MW-Voronoi region.*

Definition 6. For convenience of notation, the circle passing through two vertices V_p and V_q of region i in the MW-Voronoi diagram $\mathcal{V}$, centered at the intersection of the perpendicular bisector of V_pV_q and the curve V_kV_l, is denoted by $\Omega_{p,q}^{k,l}$, $k, l, p, q \in \mathbf{m}_i$. Also, the circle passing through one vertex V_r of MW-Voronoi region i, centered at $T_{\Omega}^{V_r}$, is denoted by $\Theta_{\Omega}^{V_r}$, for any $r \in \mathbf{m}_i$ and $\Omega \in \mathbf{\Omega}_i$.

Theorem 1. *Consider an MW-Voronoi diagram $\mathcal{V}$, and suppose that the i-th region ($i \in \mathbf{n}$) has more than one boundary curve. Let $\hat{\mathbf{C}}_i$ and $\check{\mathbf{C}}_i$ be the sets of all circles $\Omega_{p,q}^{k,l}$, $\forall k, l, p, q \in \mathbf{m}_i$ and $\Theta_{\Omega}^{V_r}$, $\forall r \in \mathbf{m}_i$, $\Omega \in \mathbf{\Omega}_i$, respectively, whose centers are on the boundary curve of the MW-Voronoi region i, and do not enclose any of the vertices of the MW-Voronoi region. Let also $\tilde{\mathbf{C}}_i$ be the set of all circumcircles of any three vertices, centered inside the MW-Voronoi region or on its boundary, which do not enclose any of the vertices of the MW-Voronoi region. Define $\mathbf{C}_i := \hat{\mathbf{C}}_i \cup \check{\mathbf{C}}_i \cup \tilde{\mathbf{C}}_i$; then $C(\bar{O}_i, \bar{r}_i)$ belongs to $\mathbf{C}_i$, and is the largest circle in this set.*

Using the result of Theorem 1, one can develop a procedure with a complexity of order $O(m_i^3)$ to calculate the Maxmin-vertex centroid in the i-th MW-Voronoi region, where m_i is the number of the vertices of the corresponding region. Since typically an MW-Voronoi region does not have ”too many” vertices, the computational complexity for calculating the Maxmin-vertex centroid is usually not very high.

B. Minmax-Vertex Strategy

The idea behind the Minmax-vertex technique is that normally for optimal coverage, each sensor should not be ”too far” from any of its Voronoi vertices. The Minmax-vertex strategy selects the target location for each sensor as a point inside the corresponding MW-Voronoi region whose distance from the farthest vertex is minimized. This point will be referred to as the *Minmax-vertex centroid*, and will be denoted by $\check{O}_i$ for the i-th region, $i \in \mathbf{n}$. Furthermore, the distance between this point and the farthest vertex from it will be represented by $\check{r}_i$. The Minmax-vertex circle is defined next.

Definition 7. The Minmax-vertex circle of an MW-Voronoi region is defined as the smallest circle centered inside the region such that all of the vertices of the region are either inside the circle or on it. This circle is, in fact, $C(\check{O}_i, \check{r}_i)$, for the i-th region.

Remark 2. If an MW-Voronoi region has exactly one boundary curve, then this curve is a circle which is also the Minmax-vertex circle for that region in the Minmax-vertex strategy.

Lemma 3. *If an MW-Voronoi region has more than one boundary curve, then the corresponding Minmax-vertex circle passes through at least two vertices.*

Lemma 4. *Consider an MW-Voronoi diagram $\mathcal{V}$, and assume that the Minmax-vertex circle of one region, say region i ($i \in \mathbf{n}$) passes through exactly two vertices, say $\check{V}_{i1}$ and $\check{V}_{i2}$. Then $\check{O}_i$ is the intersection point of the perpendicular bisector of $\check{V}_{i1}\check{V}_{i2}$ and the boundary of the i-th MW-Voronoi region.*

Theorem 2. *Given the MW-Voronoi diagram $\mathcal{V}$, let $\hat{\mathbf{W}}_i$ be the set of all circles $\Omega^{k,l}_{p,q}$, $\forall k, l, p, q \in \mathbf{m}_i$, whose centers are on the boundary of region i, and all vertices of the region are either inside or on them. Let also $\tilde{\mathbf{W}}_i$ be the set of all circumcircles of any three vertices, centered inside region i or on its boundary, with all the vertices of the region either inside or on them. Define $\mathbf{W}_i := \hat{\mathbf{W}}_i \cup \tilde{\mathbf{W}}_i$; then $C(\check{O}_i, \check{r}_i)$ belongs to $\mathbf{W}_i$, and is the smallest circle in this set.*

Using the result of Theorem 2, one can develop a procedure with a complexity of order $O(m_i^3)$ to calculate the Minmax-vertex centroid of the i-th MW-Voronoi region. As in the case of the farthest point calculation, the computational complexity for calculating the Minmax-vertex centroid is normally not very high.

Remark 3. It is worth mentioning that for the case when the sensing radii of the sensors are the same, the Minmax-vertex algorithm becomes the Minimax algorithm proposed in [9]. In other words, the Minmax-vertex algorithm proposed here is the generalized form of the one reported in the literature.

Remark 4. If the assigned location of a sensor is too far, the real coverage may not be increasing as the sensor gets closer to its destination. Therefore, as proposed in [9], the sensor may select the midpoint or $3/4$ point between its current location and the assigned location in such cases, in order to achieve better coverage.

Remark 5. In order to prevent the sensors from oscillatory movements, each sensor can check its new movement direction. If it is not in the opposite direction of the previous movement, then it moves to the target location; otherwise, it does not move [9].

IV. Simulation Results

Example 1: In this example, 27 sensors are randomly deployed in a $50\text{m} \times 50\text{m}$ flat space: 15 with a sensing radius of 6m, 6 with a sensing radius of 5m, 3 with a sensing radius of 7m, and 3 with a sensing radius of 9m. Moreover, the communication range of each sensor is assumed to be $10/3$ times greater than its sensing range. Fig. 3 shows an operational example of Minmax-vertex algorithm. Three snapshots are provided, and in each one both local coverage of the sensors (filled circles) and the MW-Voronoi regions are depicted. The initial coverage is 68.5%, but after the first round it is improved to 83.3%, and the final coverage is 95.8%. It can be observed from this figure that in the final round the sensors are distributed more evenly than the initial deployment, and that the coverage increases significantly.

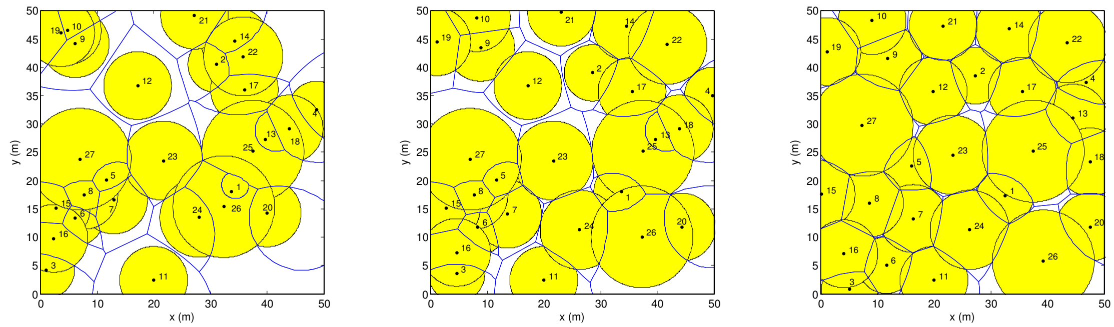
Remark 6. It is important to note that an analytical solution to the sensor deployment problem for optimal coverage is mathematically too complex to compute. This issue has also been pointed out in the existing literature, and the performance of any sensor deployment technique is typically evaluated by running a number of simulations with random settings [9], [10], [23]. This approach for the evaluation and comparison of sensor deployment strategies will be adopted in the remaining simulations in order to measure the effectiveness of the proposed techniques.

Example 2: Different settings will be examined in this example, where the two algorithms proposed in Section III are applied to a flat space of size $50\text{m} \times 50\text{m}$. The results presented in this section for field coverage are all the average values obtained by using 20 random initial deployments for the sensors. Furthermore, while the horizontal axis in all figures in this section represents a discrete parameter, the graphs are displayed as continuous curves for clarity.

Assume first there are 36 sensors: 20 with a sensing radius of 6m, 8 with a sensing radius of 5m, 4 with a sensing radius of 7m, and 4 with a sensing radius of 9m. Moreover, the communication range of each sensor is assumed to be $10/3$ times greater than its sensing range; e.g., a sensor with a sensing range of 6m has a communication range of 20m. The coverage factor (defined as the ratio of the covered area to the overall area) of the sensors in each round is depicted in Fig. 4 for the two algorithms proposed in this paper. It can be observed from this figure that both strategies result in a satisfactory coverage level of the target field in the first few rounds of the corresponding algorithms. The resultant curves also show that the Minmax-vertex algorithm performs better than the Maxmin-vertex algorithm as far as coverage is concerned.

It is desired now to compare the performance of the proposed algorithms in terms of the number of deployed sensors n. To this end, consider three more setups: n=18, 27, and 45 (in addition to n=36 discussed above). Let the changes in the number of identical sensors in the new setups be proportional to the changes in the total number of sensors (e.g., for the case of n=27 there will be 15 sensors with a sensing radius of 6m, 6 with a sensing radius of 5m, 3 with a sensing radius of 7m, and 3 with a sensing radius of 9m). Fig. 5 provides the coverage results for different number of sensors. It can be seen from this figure that the network coverage in Minmax-vertex algorithm is larger than that in Maxmin-vertex algorithm for different number of sensors.

The time it takes for the sensors to provide the desired coverage level is another important factor for measuring the efficiency of the algorithms. Since the deployment time of the sensors in each round is almost equal for all algorithms, the number of rounds required for the sensors to reach a certain coverage level is used to evaluate time efficiency. It is shown in Fig. 6 that in both algorithms the number of rounds (required to meet a certain termination condition) increases by increasing the number of sensors up to a certain value (which varies for different algorithms), and then starts to decrease by adding more sensors. This is due mainly to



(a) (b) (c)

Fig. 3. Snapshots of the execution of the movement of the sensors under the Minmax-vertex algorithm; (a) initial coverage; (b) field coverage after the first round, and (c) final coverage.

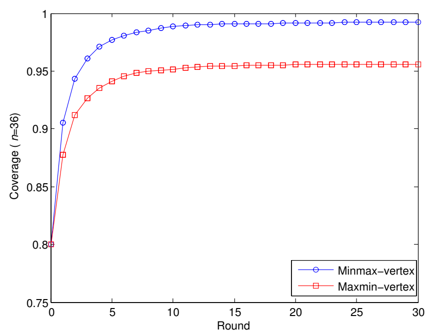


Fig. 4. Network coverage per round for 36 sensors.

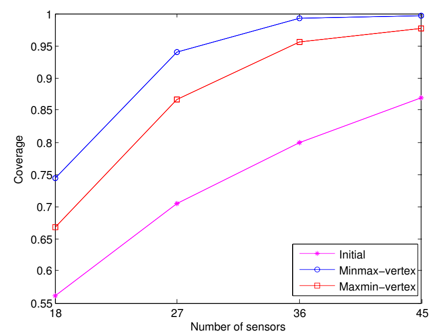


Fig. 5. Network coverage for different number of sensors using the proposed algorithms.

the fact that when there are a small number of sensors in the field, the MW-Voronoi regions are large in comparison with the corresponding sensing circles. Hence, there is a good chance that each sensor's local coverage area is completely inside its MW-Voronoi region, which means that the sensor does not need to move in order to increase its coverage area. On the other hand, when there are a large number of sensors in the field, there is a good chance that each sensor covers its MW-Voronoi region, which implies that the termination condition will be satisfied in a short period of time. It is also to be noted that when there are a relatively small number of sensors, in the Maxmin-vertex algorithm the number of rounds required for the proper termination is less than that in the Minmax-vertex algorithm. Hence, for a relatively small number of sensors, Maxmin-vertex algorithm is a good candidate for field coverage as far as deployment time is concerned.

Another important means of assessing the performance of sensor deployment algorithms is the energy consumption of the sensors. The consumed movement energy is known to be directly related to the traveling distance of the sensors, as well as the number of times they stop. Thus, to compare the proposed methods in terms of energy consumption, the traveling distance and the number of movements should be taken into consideration. Fig. 7 depicts the average moving distance for different number of sensors. This figure shows that by increasing the number of sensors, the average moving distance is decreased in both scenarios. This is due to the fact that in both algorithms when the number of sensors increases, the MW-Voronoi regions become smaller. As a result, the distance between each sensor and its destination point in the corresponding MW-Voronoi region decreases, which leads to a decrease in the average moving distance. It can be seen from Fig. 7 that the average moving distance of both algorithms are more or less the same when there are large

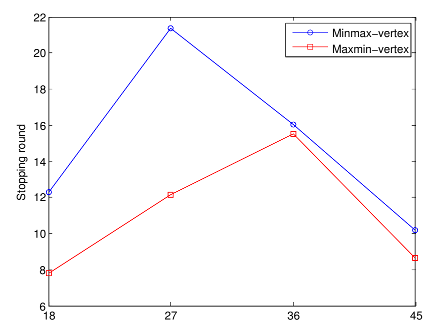


Fig. 6. The number of rounds required to reach the termination conditions for different number of sensors using the proposed algorithms.

number of sensors in the field. The number of movements versus the number of sensors is depicted in Fig. 8, where it is shown that if the number of sensors increases from 18 to 27, the number of movements increases as well. It can be observed from this figure that in general when the number of sensors is more than 27, the number of movements decreases. This is due to the fact that for large number of sensors the MW-Voronoi regions become smaller, and hence the sensors will likely cover their MW-Voronoi regions. As a result, the coverage holes will be covered in a shorter period of time, decreasing the number of movements.

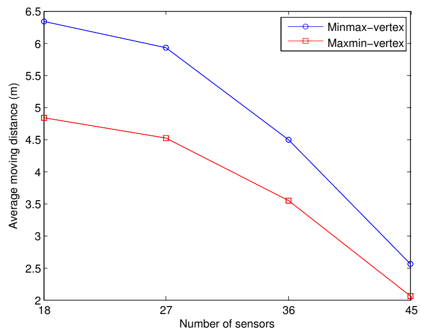


Fig. 7. The average distance each sensor travels for different number of sensors using the proposed algorithms.

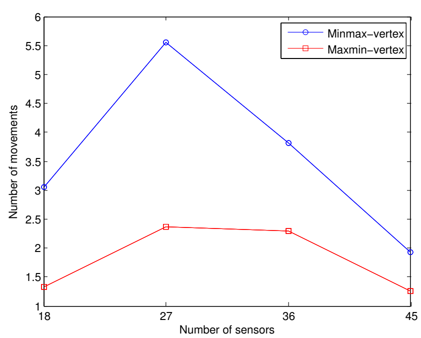


Fig. 8. The number of movements required for different number of sensors using the proposed algorithms.

V. Conclusions

This paper presents efficient sensor deployment algorithms to increase coverage in mobile sensor networks. The problem is addressed in the most general case, where the sensing radii of different sensors are not the same. A multiplicatively weighted Voronoi (MW-Voronoi) diagram is then employed to develop two vertex-based distributed deployment algorithms. Under these algorithms, the sensors move iteratively to fill coverage holes in the network. The algorithms are based on some known facts about the general characteristics of an ideal sensor configuration (e.g., each sensor should not be too far or too close to any of the vertices of its corresponding MW-Voronoi region). Simulation results are presented to compare the proposed approaches for different number of sensors in the field. Furthermore, it is shown that both algorithms improve network coverage.

References

[1] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," in *Proceedings of 6th Annual International Conference on Mobile Computing and Networking*, 2000, pp. 56–67.

[2] G. J. Pottie and W. J. Kaiser, *Wireless Integrated Network Sensors*. ACM New York, NY, USA, 2000.

[3] H. Mahboubi, K. Moezzi, A. G. Aghdam, K. Sayrafian-Pour, and V. Marbukh, "Self-deployment algorithms for coverage problem in a network of mobile sensors with unidentical sensing range," in *Proceedings of IEEE Global Communications Conference*, 2010, pp. 1–6.

[4] M. Moh, B. J. Culpepper, L. Dung, T.-S. Moh, T. H., and C.-F. Su, "On data gathering protocols for in-body biomedical sensor networks," in *Proceedings of IEEE Global Communications Conference*, 2005, pp. 2991–2996.

[5] K. A. C. Baumgartner, S. Ferrari, and T. A. Wettergren, "Robust deployment of dynamic sensor networks for cooperative track detection," *IEEE Sensors Journal*, vol. 9, 2009.

[6] S. Martinez and F. Bullo, "Optimal sensor placement and motion coordination for target tracking," *Automatica*, vol. 4, pp. 661–668, 2006.

[7] Y. Wang and C.-H. Wu, "Robot-assisted sensor network deployment and data collection," in *Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 2007, pp. 467–472.

[8] H. Mahboubi, A. Momeni, A. G. Aghdam, K. Sayrafian-Pour, and V. Marbukh, "Optimal target tracking strategy with controlled node mobility in mobile sensor networks," in *Proceedings of American Control Conference*, 2010, pp. 2921–2928.

[9] G. Wang, G. Cao, and T. F. L. Porta, "Movement-assisted sensor deployment," *IEEE Transactions on Mobile Computing*, vol. 5, pp. 640–652, 2006.

[10] A. Howard, M. J. Matarić, and G. S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks," *Autonomous Robots*, vol. 13, pp. 113–126, 2002.

[11] H. Mahboubi, K. Moezzi, A. G. Aghdam, and K. Sayrafian-Pour, "Self-deployment algorithms for field coverage in a network of nonidentical mobile sensors," in *Proceedings of IEEE International Conference on Communications*, 2011, to appear.

[12] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 2, pp. 243–255, 2004.

[13] J. Cortes, "Area-constrained coverage optimization by robotic sensor networks," in *Proceedings of 47th IEEE Conference on Decision and Control*, 2008, pp. 1018–1023.

[14] M. Pavone, E. Frazzoli, and F. Bullo, "Distributed policies for equitable partitioning: Theory and applications," in *Proceedings of 47th IEEE Conference on Decision and Control*, 2008, pp. 4191–4197.

[15] R. Graham and J. Corts, "Asymptotic optimality of multicenter voronoi configurations for random field estimation," *IEEE Transactions on Automatic Control*, vol. 1, pp. 153–158, 2009.

[16] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coordination and geometric optimization via distributed dynamical systems," *SIAM Journal on Control and Optimization*, vol. 5, pp. 1543–1574, 2005.

[17] S. Susca, S. Martinez, and F. Bullo., "Monitoring environmental boundaries with a robotic sensor network," *IEEE Transactions on Control Systems Technology*, vol. 2, pp. 288–296, 2008.

[18] A. Boukerche and X. Fei, "A Voronoi approach for coverage protocols in wireless sensor networks," in *Proceedings of IEEE Global Communications Conference*, 2007, pp. 5190–5194.

[19] J. Luo and Q. Zhang, "Probabilistic coverage map for mobile sensor networks," in *Proceedings of IEEE Global Communications Conference*, 2008, pp. 357–361.

[20] E. Deza and M. M. Deza, *Encyclopedia of Distances*. Springer, 2009.

[21] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. Wiley, 2000.

[22] A. V. Akopyan and A. A. Zaslavsky, *Geometry of Conics*. American Mathematical Society, 2007.

[23] G. Wang, G. Cao, P. Berman, and T. F. L. Porta, "A bidding protocol for deploying mobile sensors," *IEEE Transactions on Mobile Computing*, vol. 6, pp. 563–576, 2007.

[24] D. E. Koditschek, *Robot planning and control via potential functions*. MIT Press Cambridge, MA, USA, 1989.

[25] Q. Li, M. D. Rosa, and D. Rus, "Distributed algorithms for guiding navigation across a sensor network," in *Proceedings of 9th Annual International Conference on Mobile Computing and Networking*, 2003, pp. 313–325.

[26] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AOA," in *Proceedings of 22nd IEEE INFOCOM*, 2003, pp. 1734–1743.