

Mobile Sensors Deployment Subject to Location Estimation Error

Hamid Mahboubi, *Senior Member, IEEE*, Mojtaba Vaezi, *Member, IEEE*,
and Fabrice Labeau, *Senior Member, IEEE*

Abstract—Voronoi-based mobile sensor deployment algorithms require the knowledge of sensors' location to guarantee a simple, reliable coverage detection, and they miss the mark if the location is inaccurate. But, in practice, it is often too expensive to include a GPS receiver in each node, and location information is inaccurate as sensors estimate locations from the messages they receive. We study sensor deployment algorithms in the presence of location estimation error, for sensors with non-identical sensing ranges. We propose a set of Voronoi-based diagrams, named *guaranteed Voronoi diagrams*, that guarantee *single-cell-based coverage hole detection algorithms*, provided that upper bounds on localization errors are assumed. Although inaccuracy of location information would appear to deteriorate the total coverage, our simulation results demonstrate that the proposed algorithms can exploit this inaccuracy to improve the network coverage. Hence, even if the location information is exactly known at each node, assuming some error margins improves the network coverage if *guaranteed Voronoi diagrams* are used.

Index Terms—Mobile sensors deployment, coverage hole, guaranteed Voronoi diagram, estimation error.

I. INTRODUCTION

WITH a diverse range of applications in environmental monitoring and surveillance, *wireless sensor networks* (WSNs) are expected to revolutionize environmental sensing and play an important role in the future society [1]–[4]. *Mobile sensor networks* are advantageous to their static counterparts as they can cope with *topology changes* [5]–[7]. Topological changes are unavoidable in the applications where sensors are deployed in an *ad hoc* manner and move either because of the environment causes, such as wind and current, or because their objective is to track a moving target [8]. Also, significant topological changes can result simply from the malfunctioning of some sensor nodes, e.g., due to power failure [9].

Required for fulfilling the sensing tasks, *area coverage* is a key design parameter in any WSN. As such, when the network topology changes, maintaining or increasing the total sensing

coverage is very important because each sensor can obtain a limited view of the environment, both in range and accuracy, as it can only cover a limited physical area. Mobile sensors adapt to the network topology and increase the area coverage by moving toward the correct places.

Movement-assisted sensor deployment protocols based on *Voronoi diagrams* are shown to reduce the coverage holes, both for *identical sensors* [10]–[13] and *non-identical sensors*, where the sensing radii of the sensors are different [14]–[16]. Voronoi-based diagrams facilitate finding the coverage holes by dividing the field into regions such that, in each region, any point out of the *sensing disk* of the corresponding sensor is a coverage hole. This means that the point cannot be covered by other sensors. In this method, a *single-cell-based coverage analysis* is sufficient. A variant of the Voronoi diagram, known as *weighted Voronoi diagram* [16], has been then proposed to make such an analysis valid for non-identical sensors.

We study mobile sensor deployment in a network where the prior locations of nodes are not known¹, their sensing radii are non-identical, and their location information is not accurate. The aforementioned diagrams and corresponding deployment algorithms are based on the assumption that the exact location of all sensors is known at each node. In practice, however, it is often too expensive to include a GPS receiver in each sensor node. Instead, each sensor estimates the location of its neighboring nodes by using a localization technique [17], [18], or it receives the location information by communication via noisy channels [19]. Hence, this information is not usually exact, which is challenging because the system performance depends on the accuracy of the sensors' position.

In this paper, we introduce three Voronoi-based diagrams for single-cell-based coverage holes detection in the presence of localization error, collectively called *guaranteed Voronoi diagram*. We then elaborate on how the regions shape and construction differ from one algorithm to another.

These diagrams are next used to propose sensor deployment algorithms that use *farthest point (FP)* and *minmax point (MP)* [10], [20] strategies for sensor movement. The combination of the three diagrams with the two movement strategies results in six possible deployment algorithms. These algorithms can be used both when the location information is accurate or when it has a certain error, as the former is a special case of the latter. We evaluate the performance of the algorithms in terms of coverage, energy consumption and convergence

Copyright © 2015 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

H. Mahboubi and F. Labeau are with the Department of Electrical and Computer Engineering, McGill University, Montreal, QC H3A 0E9, Canada, hamid.mahboobi@mail.mcgill.ca; fabrice.labeau@mcgill.ca.

M. Vaezi is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544, USA, mvaezi@princeton.edu.

This work has been supported by Hydro-Québec, the Natural Sciences and Engineering Research Council of Canada and McGill University in the framework of the NSERC/Hydro-Québec/McGill Industrial Research Chair in Interactive Information Infrastructure for the Power Grid.

¹For example, sensor deployment where sensor nodes are dropped from a plane over the field [17].

time. All algorithms are shown to largely improve the total coverage in a few iterations. The results also indicate that the movement strategy is more determinant than the diagram, and the performance of the MP-based algorithms is better than that of the FP-based algorithms in terms of total coverage area. This is, however, achieved at the expense of more movements (starting/braking) and larger travel distances for the sensors, which in turn increase the energy consumption of the sensors.

Another important contribution of this paper is to show that assuming some bounded errors on the sensors' locations can increase the overall network coverage, if the proposed algorithms are used. This is attributed to the reduced overlapping coverage areas, due to separating the boundaries of the sensors' regions via introducing error bounds. In this sense, even if the sensors' locations are exactly known at each node, considering certain error margins can improve the network coverage based on the proposed algorithms.

The paper is organized as follows. To get an understanding of the Voronoi-based diagrams, we review them in Section II. This sets the stage to propose *guaranteed* Voronoi-based diagrams in Section III. In Section IV we study region construction for the proposed diagrams. We use these diagrams for sensor deployment in Section V, and we present numerical results in Section VI. This is followed by our concluding remarks in Section VII.

II. BACKGROUND

In mathematics, a *Voronoi diagram* or Voronoi partition of a collection of points (called seeds, nodes, etc.) is a partition of a plane into cells (polygons) such that each cell contains exactly one node and every point in a given polygon is closer to its generating node than to any other node. Voronoi diagrams are popular in discovering the coverage holes in sensor deployment algorithms [10]–[16].

Let $\mathcal{F} \subset \mathbb{R}^2$ be a field containing n sensors s_i , $1 \leq i \leq n$, with sensing ranges r_i , $r_i > 0$, located at planar coordinate $p_i = (x_i, y_i)$. We denote this set of sensors by $\mathcal{S} = \{s_i\}$, where $i \in \mathcal{N}_S := \{1, 2, \dots, n\}$. Throughout this paper we assume $i, j \in \mathcal{N}_S$ and $i \neq j$, unless otherwise stated. We use $d(a, b)$ or $\|b - a\|$ to denote the *Euclidean distance* between two points a and b .

Definition 1. The *coverage area* of a sensor (s_i, r_i) located at p_i is a *disk* of radius r_i centered at p_i . This is called the *sensing disk*, and its border the *sensing circumcircle*. A point q is covered by this sensor if and only if $d(q, p_i) \leq r_i$.

Definition 2. For a given deployment of sensors in a field \mathcal{F} , the *total coverage hole* is defined as the collection of all points in \mathcal{F} that are out of the sensing circumcircles of all sensors.

A key requirement of any effective sensor deployment is to maximize the total coverage of a desired field (\mathcal{F}) for a given number of sensors (n). Commonly, the field is divided into n *cells*, each corresponding to one sensor, and the *local coverage* of each cell is maximized separately. Described shortly in this section, Voronoi-based diagrams are popular for this purpose.

A. Identical Sensors

Consider a set of sensors with the same sensing radii, i.e., $r_i = r_j = r$. A *Voronoi diagram* (VD) partitions the field into n cells in a way that any point in each cell can be only sensed by its corresponding sensor. More precisely, the region corresponding to sensor i is defined by

$$\Pi_i^{\text{VD}} = \{q \in \mathcal{F} \mid d(q, p_i) \leq d(q, p_j)\}. \quad (1)$$

Obviously, if s_i is not able to sense a given point q in its own region, i.e., if $d(q, p_i) > r_i$, then other sensors are not able to do so, either. Hence, to find the coverage holes of the field \mathcal{F} , it suffices to check the coverage holes at each cell individually. Such a *single-cell-based* coverage hole detection simplifies the hole detection process and is suitable to develop distributed, self-deployment protocols for sensor networks [10].

B. Non-Identical Sensors

The above partitioning assumes the sensing ranges of the sensors are equal, i.e., $r_i = r_j = r$. However, in many applied WSNs the sensors are *non-identical* [21], i.e., their sensing radii are different. In such networks, the partitioning (1) is obviously inefficient as a point not covered by its sensor may be covered by a neighboring sensor. To benefit from the simplicity of single-cell-based coverage hole detection, *weighted Voronoi diagrams* are proposed in [15]. These diagrams are developed by modifying the distance metrics with positive weights. Some commonly-used weighted Voronoi diagrams, in the context of WSNs, are reviewed here.

1) *Multiplicatively Weighted Voronoi Diagram (MWVD)*: This diagram is developed based on multiplying the distance between points by positive weights [22]. In the context of sensor deployment, the multiplicatively weighted distance of a point q from the sensor (s_i, r_i) is defined as $d_{\text{MW}}(q, s_i) := \frac{1}{r_i} \times d(q, p_i)$, where p_i represents the position of s_i [23]. Using the above distance metric, the set of all points q which are closer to s_i than s_j , is characterized by

$$\Pi_i^{\text{MWVD}} = \{q \in \mathcal{F} \mid \frac{d(q, p_i)}{r_i} \leq \frac{d(q, p_j)}{r_j}\}. \quad (2)$$

2) *Additively Weighted Voronoi Diagram (AWVD)*: An additively weighted distance is defined by subtracting a positive weight from the distance between points, e.g., $d_{\text{AW}}(q, s_i) := d(q, p_i) - r_i$ [15]. With this metric, additively weighted Voronoi diagram (AWVD) divides the field \mathcal{F} such that

$$\Pi_i^{\text{AWVD}} = \{q \in \mathcal{F} \mid d(q, p_i) - r_i \leq d(q, p_j) - r_j\}. \quad (3)$$

3) *Power Diagram (PD)*: In power diagram, the power of q with respect to a circle of radius r_i centered at p_i is defined as $d_p(q, s_i) = d^2(q, p_i) - r_i^2$, and the subregion corresponding to s_i is characterized by [24]:

$$\Pi_i^{\text{PD}} = \{q \in \mathcal{F} \mid d^2(q, p_i) - r_i^2 \leq d^2(q, p_j) - r_j^2\}. \quad (4)$$

Remark 1. From (2)–(4), it is easy to see that when $r_i = r_j$ for all $i, j \in \mathcal{N}_S$, the above weighted Voronoi diagrams reduce to the Voronoi diagram in (1).

A fundamental property of the above weighted diagrams is that if a point in Π_i is not covered by s_i it cannot be covered

by any other sensor implying that the coverage holes can be found based on a *single-cell-based* search. This is because

Proposition 1. *In all of the above weighted Voronoi diagrams, for any $q \in \Pi_i$, $d(q, p_i) > r_i \Rightarrow d(q, p_j) > r_j$.*

Proposition 1 is valid only if the perfect locations of each sensor and its neighbors are known; otherwise, the diagrams defined by (1)–(4) cannot guarantee a single-cell-based coverage hole detection.

Localization, i.e., determining the location of sensors, is one of the central problems of real-world WSNs and is required for the completion of many basic tasks, including packet routing and coverage detection. In practice, the locations of sensors are not perfectly known, as they are either estimated by using a localization technique or received via a noisy channel. But, if the users are not able to obtain the accurate location information, related applications, e.g., coverage hole detection, cannot be accomplished. This motivates the investigation of reliable and simple coverage analysis methods.

III. GUARANTEED VORONOI-BASED DIAGRAMS

In this section, by modifying the existing Voronoi diagrams, we propose new diagrams that guarantee a single-cell-based coverage analysis. We study a WSN in which the locations of sensors are estimated at different nodes and thus are subject to errors. Our objective is to find n cells, each corresponding to one of the sensors, such that it is guaranteed that if a point inside a cell is not sensed by its corresponding sensor, no other sensor can sense it either.

A. Regions' Characteristics

Suppose that measurement errors are upper bounded and these bounds are known at each sensor. More precisely, let p_i (the exact location of s_i) be within a disk of radius ϵ_{ij} , $\epsilon_{ij} \geq 0$, centered at p_{ij} , where p_{ij} is the measured location of s_i at sensor j . Hence, $d(q, p_{ij}) - \epsilon_{ij} \leq d(q, p_i) \leq d(q, p_{ij}) + \epsilon_{ij}$; i.e., ϵ_{ij} denotes the maximum error in the estimation of the location of s_i at s_j . Similarly, suppose s_i measures its own location (p_{ii}) with some errors bounded by ϵ_{ii} .

Since p_{ij} is the estimated position of s_i at s_j , then the exact location of s_i is somewhere within a disk of radius ϵ_{ij} centered at p_{ij} . Hence, in general, for an arbitrary distance function f , the set of all points $q \in \mathcal{F}$ surely closer to s_i than to s_j is characterized by

$$\max_{q_1 \in C(p_{ii}, \epsilon_{ii})} f(q, q_1, r_i) \leq \min_{q_2 \in C(p_{ji}, \epsilon_{ji})} f(q, q_2, r_j), \quad (5)$$

where $C(a, r)$ denotes a circle of radius r centered at a .

For example, for the additively weighted distance where $f(q, q_1, r_i) \triangleq d(q, q_1) - r_i$, (5) reduces to

$$\max_{q_1 \in C(p_{ii}, \epsilon_{ii})} d(q, q_1) - r_i \leq \min_{q_2 \in C(p_{ji}, \epsilon_{ji})} d(q, q_2) - r_j. \quad (6)$$

Next, since the maximum possible distance between q and a point in $C(p_{ii}, \epsilon_{ii})$ is $d(q, p_{ii}) + \epsilon_{ii}$ and the minimum distance between q and a point in $C(p_{ji}, \epsilon_{ji})$ is $d(q, p_{ji}) - \epsilon_{ji}$, we design the cell borders based on these worst case scenario. That is, we use $d(q, p_{ii}) + \epsilon_{ii}$ and $d(q, p_{ji}) - \epsilon_{ji}$, rather than

$d(q, p_i)$ and $d(q, p_j)$ when finding the cell borders. Then, for any $q \in \mathcal{F}$, the region characterized by (6) can be shown by

$$d(q, p_{ii}) + \epsilon_{ii} - r_i \leq d(q, p_{ji}) - \epsilon_{ji} - r_j. \quad (7)$$

This means that, at the worst case where sensor s_i is shifted ϵ_{ii} away from point q in its region and the neighboring sensors have moved toward q by ϵ_{ij} , s_i is still closer to q than any s_j .

Using such a modified diagram, a single-cell-based search is guaranteed, even with estimation error. Then, the mathematical characterization of the i th region for different diagrams can be modified as below:

1) Guaranteed MWVD (GMWVD):

$$\Pi_i^{\text{GMWVD}} = \{q \in \mathcal{F} \mid \frac{d(q, p_{ii}) + \epsilon_{ii}}{r_i} \leq \frac{d(q, p_{ij}) - \epsilon_{ij}}{r_j}\}. \quad (8)$$

A special case of this approach, in which $\epsilon_{ii} = 0$ has recently been studied in [19].

2) Guaranteed AWVD (GAWVD):

$$\Pi_i^{\text{GAWVD}} = \{q \in \mathcal{F} \mid d(q, p_i) - r_i + \epsilon_{ii} \leq d(q, p_j) - r_j - \epsilon_{ij}\}. \quad (9)$$

3) Guaranteed Power Diagram (GPD):

$$\Pi_i^{\text{GPD}} = \{q \in \mathcal{F} \mid [d(q, p_{ii}) + \epsilon_{ii}]^2 - r_i^2 \leq [d(q, p_{ij}) - \epsilon_{ij}]^2 - r_j^2\}. \quad (10)$$

Remark 2. One special case of the above diagrams is the case with $\epsilon_{ii} = 0$, or $p_{ii} = p_i$. This is practically important, e.g., in GPS-equipped sensors [18] and can be studied per se, too.

B. Regions' Properties

It is now easy to see that Proposition 1 is valid for any of the above guaranteed regions. In addition, we have

Proposition 2. *The regions defined by (8)–(10) are disjoint.*

Proof: On the one hand, one can straightforwardly verify that $\Pi_i^{\text{GMWVD}} \subseteq \Pi_i^{\text{MWVD}}$, $\Pi_i^{\text{GAWVD}} \subseteq \Pi_i^{\text{MAVD}}$ and $\Pi_i^{\text{GPD}} \subseteq \Pi_i^{\text{PD}}$. For example, from (8) we obtain $\frac{d(q, p_i)}{r_i} \leq \frac{d(q, p_j)}{r_j} - \frac{\epsilon_{ii}}{r_i} + \frac{\epsilon_{ij}}{r_j}$ which implies $\frac{d(q, p_i)}{r_i} \leq \frac{d(q, p_j)}{r_j}$ in (2), and proves $\Pi_i^{\text{GMWVD}} \subseteq \Pi_i^{\text{MWVD}}$. On the other hand, since the regions corresponding to the MWVD, AWVD and PD, defined by (2)–(4), are mutually disjoint, their subsets are disjoint too. ■

It should be noted that the regions generated by (8)–(10) do not partition the field. That is, there are *neutral areas* which do not belong to any of the guaranteed regions. Mathematically, $\bigcup_{i=1}^n \Pi_i \neq \mathcal{F}$ for the GMWVD, GAWVD and GPD. However, the total neutral area is not comparable to the assigned area, if ϵ_{ij} is relatively small for any $i, j \in \mathcal{N}_S$. More importantly, the neutral areas are not necessarily coverage holes; they might be covered with one or more sensors. Examples of the above-mentioned guaranteed diagrams are shown in Fig. 1

IV. SHAPE OF THE REGIONS

A. With Perfect Location Information

To determine to which cell a given point belongs to, it is important to find the borders between cells in Voronoi-based diagrams. To find the locus of points which are on the cell

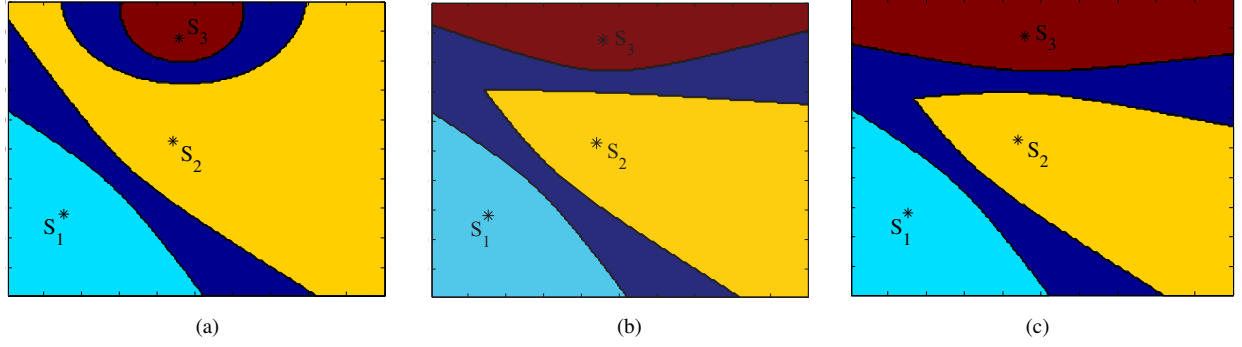


Fig. 1. Guaranteed diagrams for a field containing three sensors with different sensing radii and imperfect location information. The dark blue regions show the areas which are not assigned to any sensor, implying that a single-cell-based coverage hole detection is not applicable to them. (a) Guaranteed multiplicatively weighted Voronoi diagram (GMWVD). (b) Guaranteed additively weighted Voronoi diagram (GAWVD). (c) Guaranteed power diagram (GPD).

borders of different diagrams represented by (1)–(4), the “ \leq ” needs to be satisfied with “equality” in those regions. Then, for a VD in (1) we get $d(q, p_i) = d(q, p_j)$ which implies q to be on the *perpendicular bisector* of the segment connecting p_i to p_j ; i.e., the locus is a line. Next, on the borders of the MWVD we have $\frac{d(q, p_i)}{r_i} = \frac{d(q, p_j)}{r_j}$, or equivalently, $\frac{d(q, p_i)}{d(q, p_j)} = \frac{r_i}{r_j} \triangleq \alpha$, and we have

Proposition 3. *The border between s_i and s_j in the MWVD is an arc of an Apollonian circle of radius $R = \frac{\alpha d(p_i, p_j)}{(1-\alpha^2)}$.*

Proof: See Section VIII-A. ■

This is visualized in Fig. 2, where A and B represent the locations of s_i and s_j and $\alpha = r_i/r_j < 1$ is a constant. The locus of all points E such that $EA/EB = \alpha$ is a circle of radius $R = \frac{\alpha}{1-\alpha^2}d(A, B)$. In addition, the center of this circle is on the extension of line segment AB such that $OA = \frac{\alpha^2}{1-\alpha^2}d(A, B)$ and $OB = \frac{d(A, B)}{1-\alpha^2}$. These can be verified based on the coordinate of the O , which is $(\frac{x_A - \alpha^2 x_B}{1-\alpha^2}, \frac{y_A - \alpha^2 y_B}{1-\alpha^2})$ from the proof given in Section VIII-A.

Similarly, for the AWVD it can be seen that

Proposition 4. *The borders of regions in the AWVD are composed of arcs of different hyperbolas.*

Proof: From (3), the points satisfying the distance metric of the AWVD with equality result in $d(q, p_i) - d(q, p_j) = r_i - r_j \triangleq \beta$, where β is a constant. On the other hand, from analytical geometry we know that *hyperbola* is defined as the locus of points such that the absolute value of the difference of their distances from two fixed points is a constant. Hence, the border between s_i and s_j , located in A and B , respectively, in the AWVD is an arc of a hyperbola which intersects the segment AB in a point like T such that $AT = \frac{d(A, B) + \beta}{2}$ and $BT = \frac{d(A, B) - \beta}{2}$ (see Fig. 3). ■

Proposition 5. *The border between s_i and s_j in the PD is a line.*

Proof: Consider a constant α and two points A and B in a 2D plane. It can be easily shown that the locus of any point E such that $d^2(A, E) - d^2(B, E) = \alpha$ is a line perpendicular

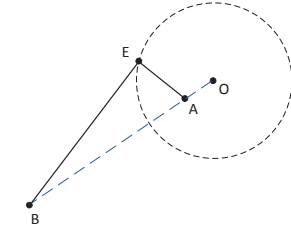


Fig. 2. Apollonian circle centered at O . A and B are the locations of s_i and s_j .

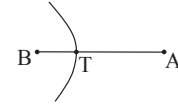


Fig. 3. A representation of the border between sensors in the AWVD, where s_i and s_j are located in A and B , respectively.

to segment AB which intersects it in H such that $AH = \frac{d^2(A, B) - \alpha}{2d(A, B)}$ and $BH = \frac{d^2(A, B) + \alpha}{2d(A, B)}$ (see Fig. 4). Hence, the border between s_i and s_j in the PD is a *line* perpendicular to the segment connecting s_i to s_j [25], [26]. ■

Table I summarizes the main points of the above propositions.

When location information is perfect, the power diagram is preferred to the other weighted Voronoi diagrams from the region construction point of view, because computing the cell boundaries and constructing the regions is simpler. This is particularly important in the sensor networks context as a less computationally complex algorithm implies a better energy efficiency. However, when estimation error comes in, this argument is not valid.

B. With Location Estimation Error

The main advantage of the GAWVD to the GMWVD and GPD is the fact that the borders of the region for each sensor can be determined analytically; in fact, similar to the AWVD diagram, these borders are hyperbolic arcs, except that the constant $\beta \triangleq r_i - r_j$ is replaced with $\beta - \epsilon_{ii} - \epsilon_{ji}$, as from

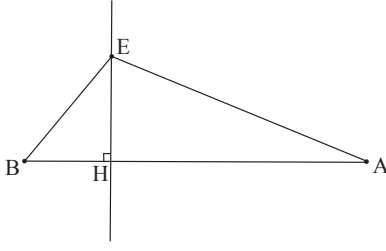


Fig. 4. A representation of the border between sensors in the PD. A and B are the locations of s_i and s_j , respectively.

TABLE I
TYPES OF VORONOI DIAGRAMS AND THEIR BORDERS

Diagram Type	Distance from s_i	Borders
VD	$\ x - p_i\ $	line
AWVD	$\ x - p_i\ - r_i$	hyperbolic arc
MWVD	$\frac{\ x - p_i\ }{r_i}$	circular arc
PD	$\ x - p_i\ ^2 - r_i^2$	line

(9) we obtain $d(q, p_{ii}) - d(q, p_{ij}) = r_i - r_j - \epsilon_{ii} - \epsilon_{ji}$. In contrast, the characteristics of the regions defined by the GMWVD and GPD are different from that of the MWVD and PD. Moreover, it is not easy to find the borders of regions in the GMWVD and GPD. For instance, (8) does not result in a constant $\frac{d(q, p_{ii})}{d(q, p_{ij})}$; hence, unlike (2), the locus is not an Apollonian circle. Likewise, the regions shape is not known for (10), while in GAWVD the regions have the same shape as AWVD.

There are many different algorithms for constructing various types of Voronoi diagrams, but in all of them it is required to find the boundaries of the Voronoi regions [26]. Since energy consumption is a major concern in any WSN, it is important to keep this process as simple as possible and the GAWVD seems to be a better option in this sense, as it will require less message exchange for region construction.

V. DEPLOYMENT PROTOCOLS

In this part different deployment protocols are developed for a network of mobile sensors with non-identical sensing ranges subject to inaccurate locations information. To this end, we use existing movement strategies, e.g., farthest point method [16], and apply them to the proposed guaranteed diagrams.

A. Deployment Algorithm Details

The proposed deployment algorithms are iterative, and in each iteration the following steps are carried out:

- i) All sensors broadcast their sensing radii and positions. Thus, based on the received information, every sensor constructs its own region, given a guaranteed diagram.
- ii) Every sensor detects coverage holes in its own region in a distributed manner (i.e., independently).
- iii) After discovering the coverage holes, by using a specific movement algorithm (which will be discussed in subsection V-B), the corresponding sensor calculates its new candidate location.

Algorithm 1 Deployment Algorithm for the FPGAW algorithm

```

 $k \leftarrow 0$ ,  $iterations \leftarrow 0$ ,  $\mathcal{D} \leftarrow 1$ 
while  $\mathcal{D} = 1$  and  $iterations < I_{max}$  do
  for  $i = 1$  to  $n$  do
    • construct  $\Pi_i$  according to (9) and set  $\Pi_i^k \leftarrow \Pi_i$ 
    • find  $\pi_i^k$  (the area of the  $i$ th region that is covered by  $s_i$ )
  end for
  •  $iterations \leftarrow iterations + 1$ 
  •  $k \leftarrow k + 1$ 
  •  $\mathcal{C} \leftarrow 0$ 
  for  $i = 1$  to  $n$  do
    • calculate a new location ( $p_i^k$ ) for  $s_i$ , based on the FPGAW movement strategy
    • evaluate  $\pi_i^k$  based on the new location for the current region ( $\Pi_i^{k-1}$ )
      if  $\pi_i^k > \pi_i^{k-1} + \delta$  then
        move  $s_i$  to  $p_i^k$ 
         $\mathcal{C} \leftarrow \mathcal{C} + 1$ 
      end if
  end for
  if  $\mathcal{C} = 0$  then
     $\mathcal{D} = 0$ 
  end if
end while

```

iv) Once the new location is calculated, the corresponding coverage area is evaluated (based on the previously-constructed region) and compared to the current coverage area. The sensor moves to the new location only if the resulting coverage area is greater than the present value; otherwise, it does not move in this iteration.

v) To have a termination criterion for the algorithm, a proper threshold δ is defined; the algorithm is terminated if no sensor can improve its coverage area by this threshold or a predefined number of iterations (I_{max}) has been completed. δ and I_{max} are chosen based on which of the coverage, energy consumption or convergence time is the main concern. For example, when the convergence time or energy consumption is the main concern the operator choose a relatively small I_{max} and relatively large δ in the beginning. On the other hand, when the coverage is the most important concern, a relatively large I_{max} and small threshold δ are chosen by the operator such that the covered area increases as much as possible.

The deployment algorithms are iterative and in each iteration it is tried to improve the total coverage, at least as much as a threshold $\delta > 0$. The algorithm is stopped if no improvement is possible or a certain number of iterations (I_{max}) has passed. Algorithm 1 briefly describes the FPGAW method. The deployment algorithm for the other algorithms (i.e., the FPGMW, MPGP, etc.) is exactly the same, except that the movement algorithm changes accordingly.

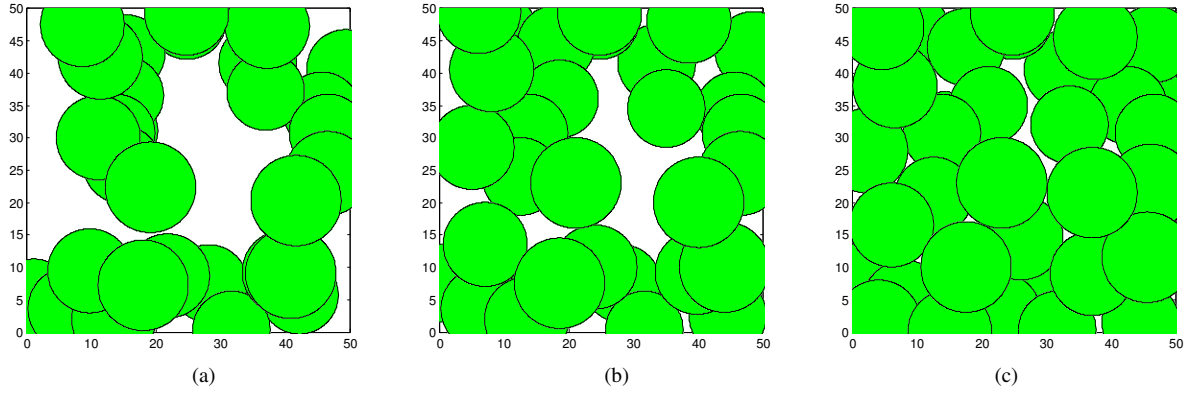


Fig. 5. Snapshots for the positions of 36 sensors with three different sensing radii, as given in Table II, at a $50m \times 50m$ field under MPGP algorithm. (a) the initial coverage (b) coverage after one iteration (c) coverage after five iterations

B. Movement Strategies

In this subsection, we describe the movement strategies applied for sensor deployment in this paper. Remember that a movement strategy is required in step iii of the above iterative algorithm to find a new candidate point (position) for each sensor. We use the following movement strategies in our work.

1) *Farthest Point (FP) Strategy*: The main idea behind this strategy is to move every sensor to the farthest point in its region such that the area of coverage hole is decreased. If the sensor s_i detects a coverage hole in its corresponding region, it calculates the farthest point in that region, and moves toward it until this point is covered.

2) *Minmax Point (MP) Strategy*: There are certain network topologies and sensor configurations for which the FP strategy is not very effective. Examples of such regions are shown in [27], in which usually the angle is narrow, and thus if the sensor moves there the local coverage (i.e., coverage in the cell) decreases. Under such a situation, the algorithm forces the sensor to remain in its previous location while it was still possible to improve the local coverage if the sensor was properly placed in the region.

The minmax point (MP) strategy is proposed in the sequel to address this shortcoming of the FP strategy. The main idea behind the MP strategy is that to achieve maximum coverage, no sensor should be too far from any point in its corresponding region. The MP strategy considers the candidate position for each sensor as a location inside the corresponding region whose distance from the farthest point of the region is minimum. This may yield a better coverage compared to the one obtained by using the FP technique.

The above movement strategies can be used in combination with the diagrams we introduced in Section III. Specifically, the following deployment algorithms are possible if the FP strategy is used for movement (step iii of the above iterative algorithm).

- Farthest point based GMWVD (FPGMW)
- Farthest point based GAWVD (FPGA)
- Farthest point based GPVD (FPGP)

Similarly, one can use the MP movement strategy in each of the diagrams to get MPGMW, MPGA and MPGP deployment algorithms. We compare the performance of these

TABLE II
THE NUMBER OF SENSORS WITH DIFFERENT RADII FOR EACH n .

	$r = 6m$	$r = 6.5m$	$r = 7m$
$n = 18$	10	6	2
$n = 27$	15	9	3
$n = 36$	20	12	4
$n = 45$	25	15	5

algorithms for various number of sensors in the next section.

Remark 3. It is worth mentioning that the computational complexity for calculating the new location of the i -th sensor in the farthest point and minmax point strategies is of order $O(m_i)$ and $O(m_i^4)$, respectively, where m_i is the number of boundary curves of the i -th region [28], [16]. Since typically a guaranteed Voronoi-based region does not have "too many" boundary curves, the computational complexity of the proposed strategies is usually not very high.

VI. SIMULATION RESULTS

We evaluate the performance of the proposed algorithms by performing simulations for a different number of sensors in a $50m \times 50m$ field. The results presented in this section are obtained by using 20 random initial arrangements for the sensors. The coverage improvement threshold δ is set to be $0.1m^2$, meaning that if the local coverage (i.e., coverage in a cell) improvement by all sensors is less than $0.1m^2$, the algorithm stops. There are three types of sensors with sensing radii equal to 6m, 6.5m and 7m. The value of ϵ_{ii} is set to be 0, i.e., we assume each sensor knows its exact location, whereas $\epsilon_{ij} = 0.1m$ for all nodes. We run simulations for four different number of sensors, namely, $n = 18, 27, 36, 45$. Table II shows the number of sensors from each type for different n . For each setting, we carry out simulation for different algorithms which are the FPGAW, FPGMW, FPGP, MPGA, MPGMW, and MPGP. Note that, the parameters used in this section are within the ranges used in the literature [10], [29]–[32], and they are also consistent with sensor prototypes such as Smart Dust (UC Berkeley), CTOS dust, and Wins (Rockwell) [33].

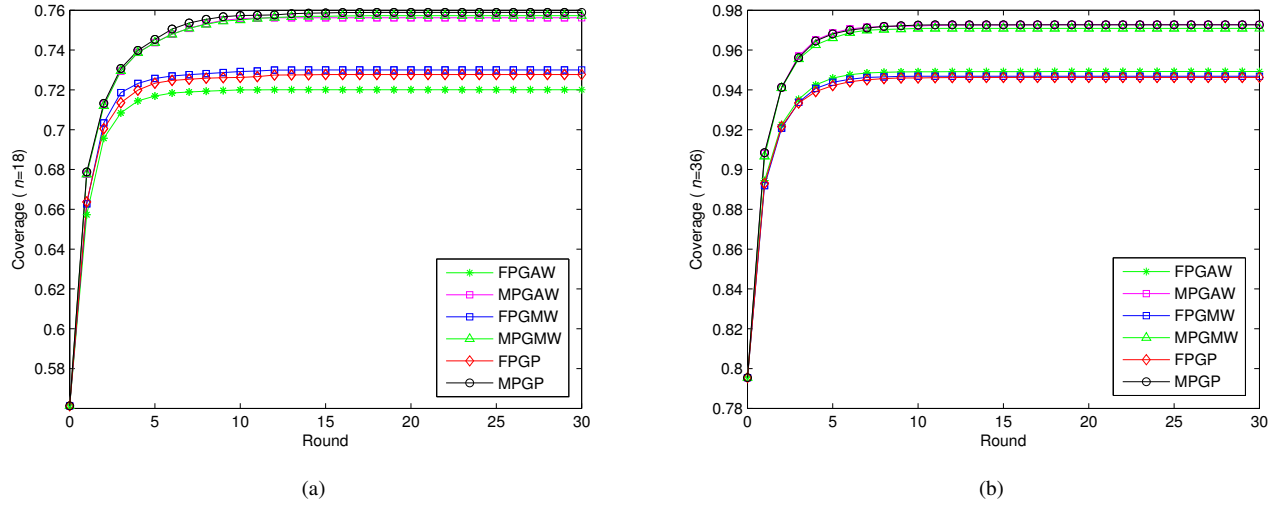


Fig. 6. Network coverage using different algorithms at various iterations: (a) for 18 sensors (b) for 36 sensors.

A. Coverage

Total sensing coverage of the field, after a certain number of iterations, is the first parameter we are interested in. Fig. 6 shows the coverage factor of the sensor network (defined as the ratio of the covered area to the total area) for each algorithm at various iterations when 18 and 36 sensors are deployed. It can be seen that the coverage is sharply increasing for the first few iterations and after several iterations the coverage remains almost constant. For example, the coverage goes from less than 80% up to more than 96% just in five iterations, for all MP-based algorithms in Fig. 6(b). The coverage plots for $n = 27, 45$ are not included due to the space limit and similarity of their behavior to the cases in Fig. 6. Obviously, the sensing coverage depends on the number of sensors and it increases when n goes up. To visualize this, in Fig. 7 we plot the initial and final coverage factor versus the number of sensors for each of the proposed deployment algorithms.

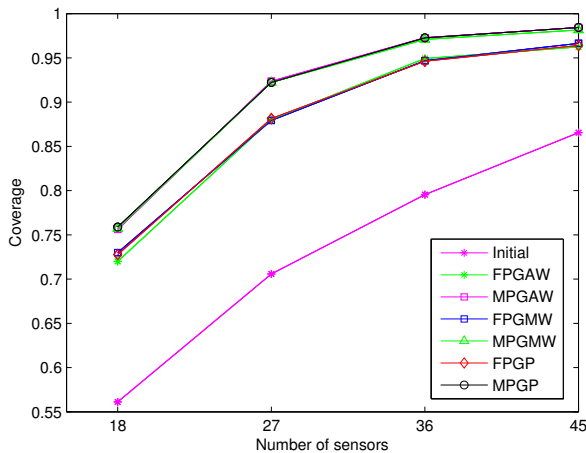


Fig. 7. Initial and final coverage versus the number of sensors for different algorithms. The FP- and MP-based algorithms make two separate clusters.

Overall, the coverage provided by the MP-based algorithms is larger than that of the FP-based algorithms. This is, however, achieved at the expense of a larger number of movements, more travel distances, and longer time (more iterations), as represented in Fig. 8(a) to Fig. 8(c), respectively. Figure 8(a) indicates that the average moving distance per sensor, required to cover the field, decreases as the number of sensors increases. This makes sense as the higher the number of sensors, the better the initial coverage and thus fewer movements are needed to improve the coverage. It is worth mentioning that finding the minimum number of sensors to reach a 100% coverage for a network of nonidentical mobile sensors is an open problem even when the exact locations of sensors are known and there is no measurement error.

B. Energy Consumption

The three parameters evaluated in Fig. 8 are key indicators of energy consumption in sensor deployment [27]. According to [34], [35], each mobile sensor node consumes 8.268J energy to travel one meter non-stop. In addition, each starting/braking consumes energy which varies in different systems [10].² Based on the above examples, we consider two cases: in the first case the required energy for a sensor to restart is equal to the amount of energy it consumes to travel 1m, and in the second case it is equal to energy consumption for 4m movement [10], [33]. Tables III and IV represent the energy consumption corresponding to each algorithm at those two scenarios.

In general, the difference in the performance of the MP-based algorithms is not too much compare to the differ-

²Apart from the movement and restarting, there is also another cause for energy consumption. This is message passing required to figure out the neighboring sensors and make the movement decision. We neglect this as it is very small compared to the energy require for movements [10].

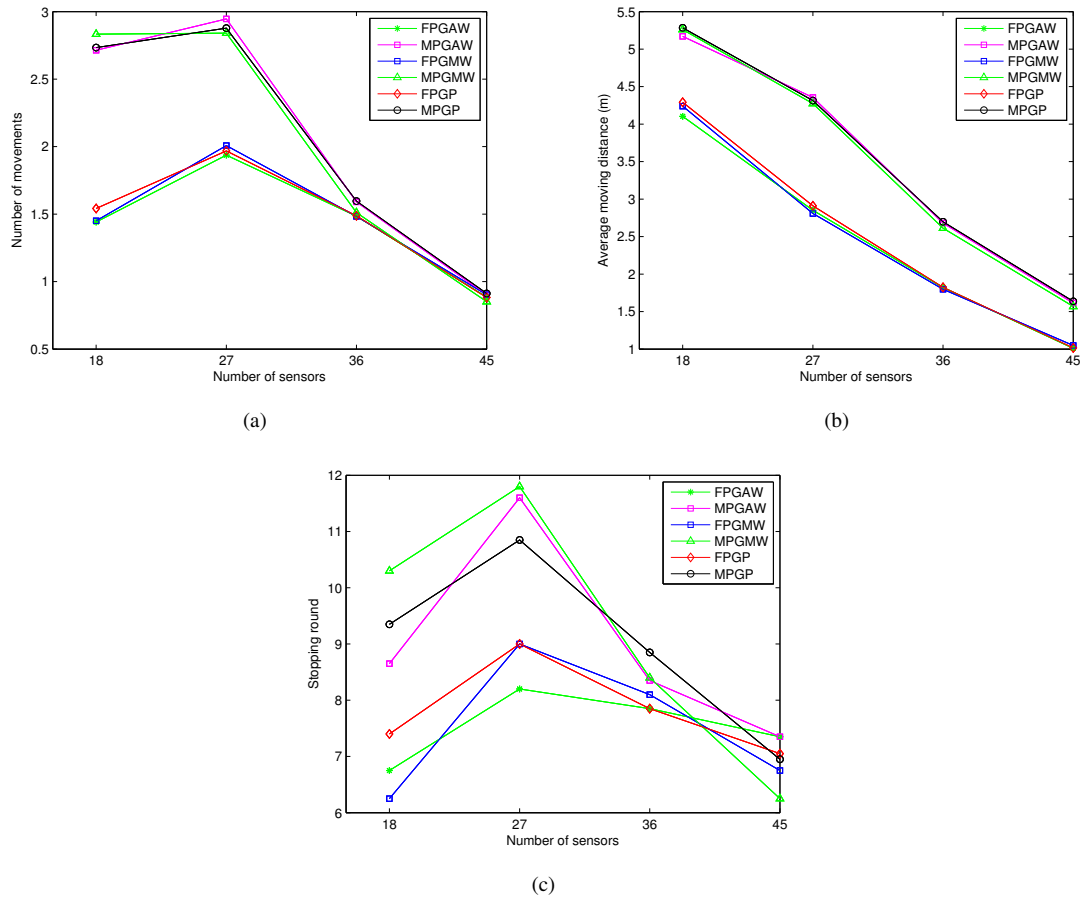


Fig. 8. Movement related graphs for the proposed algorithms with different number of sensors: (a) number of movements, (b) average travel distance per sensor, and (c) average number of iterations, required to reach the termination condition.

TABLE III
THE ENERGY CONSUMPTION IN JOULE IN THE FIRST CASE.

Algorithm	$n = 18$	$n = 27$	$n = 36$	$n = 45$
MPGAW	65.1927	60.3690	35.3093	20.7924
MPGMW	66.9240	58.8187	34.0968	20.0077
MPGP	66.2859	59.4573	35.4912	21.0696
FPGAW	45.8470	39.5863	27.3415	15.6598
FPGMW	45.0402	39.8145	27.1246	16.1306
FPGP	48.2261	40.3440	27.3778	15.6896

TABLE IV
THE ENERGY CONSUMPTION IN JOULE IN THE SECOND CASE.

Algorithm	$n = 18$	$n = 27$	$n = 36$	$n = 45$
MPGAW	132.5080	133.4489	74.7545	43.1435
MPGMW	137.2020	129.2804	71.5784	41.1187
MPGP	134.0835	130.8377	75.0743	43.6963
FPGAW	81.6061	87.6326	64.1685	37.5976
FPGMW	83.0060	89.6063	63.9172	38.5369
FPGP	86.4656	89.1711	64.2738	37.5723

TABLE V
THE RATIO OF QUALITY AGAINST PRICE FOR THE FIRST SCENARIO.

Algorithm	$n = 18$	$n = 27$	$n = 36$	$n = 45$
MPGAW	1.6111	1.4166	1.9132	2.6302
MPGMW	1.5716	1.4518	1.9774	2.7256
MPGP	1.5901	1.4363	1.9031	2.5954
FPGAW	2.1814	2.0590	2.4109	3.4146
FPGMW	2.2510	2.0448	2.4241	3.3287
FPGP	2.0958	2.0230	2.4001	3.4135

TABLE VI
THE RATIO OF QUALITY AGAINST PRICE FOR THE SECOND SCENARIO.

Algorithm	$n = 18$	$n = 27$	$n = 36$	$n = 45$
MPGAW	0.7926	0.6408	0.9037	1.2676
MPGMW	0.7666	0.6605	0.9420	1.3263
MPGP	0.7861	0.6527	0.8997	1.2514
FPGAW	1.2255	0.9301	1.0272	1.4222
FPGMW	1.2214	0.9086	1.0287	1.3933
FPGP	1.1689	0.9153	1.0223	1.4254

ence between one MP-based algorithm and one FP-based algorithm, in terms of coverage, average moving distance, average number of movements to achieve a certain level of coverage or after a certain number of iterations, and the energy required for that. A similar pattern is observed for the FP-based algorithms. In other words, although the structures of

cells in the GAWVD, GMWVD and GPD are different, this does not have much effect on the performance of the sensor deployment algorithms. In contrast, the movement strategy, i.e. the FP or MP, is more determinant, as for a given diagram (e.g.,

the GPD), the performance of the deployment algorithms, in various senses, is noticeably different for the MP and FP, as it is evident from Figs. 6–8 and Tables III–IV.

By considering the coverage and energy consumption as quality and price of the algorithms, respectively, one can define the ratio of quality to price as a new parameter for comparing the performance of the proposed algorithms. More precisely, this parameter is defined as the (coverage factor \times area of the field) over (energy consumption of each sensor \times number of sensors). Tables V and VI present the above-mentioned parameter for each algorithm in both scenarios.

C. Convergence time

It is also instructive to look at the number of iterations required to terminate each algorithm. This is shown in Fig. 8(c) in which, except for $n = 18$, the number of iterations required to reach a steady state (i.e., to terminate the algorithm since there is not more improvement) decreases as the number of sensors goes up. This makes sense, as with higher number of sensors, initial coverage is high and it takes less time (iterations) to reach the steady state. In addition, when there are larger numbers of sensors in the field, the guaranteed Voronoi-based regions are smaller and hence the chance that each sensor covers its region is higher, which implies that the termination condition will be satisfied in a shorter period of time. However, when the number of sensors is very small to cover the field, the average number of movements and the number of iterations become smaller, possibly because initial distribution of sensors is such that they do not overlap much, and thus do not require many movement to get to the final (optimum) distribution. To better understand this, consider the extreme case where there is only one sensor in the field. Then, it is clear that if the sensing area of the sensor is completely inside the field, the coverage cannot be improved anymore and thus the sensor does not need any movement; further, the number of iterations in the deployment algorithm will be 0. If the sensing area of the sensor is partially out of the field, the system can reach its best performance just in one movement and at the first iteration. In light of Fig. 8(c), and Fig. 8(a), it can be concluded that 18 sensors are not enough to effectively cover the field.

D. Magnitude of Error Bound

Finally, we investigate the effect of the magnitude of error on the coverage performance. Table VII presents the coverage results for the FPGMW algorithm. Interestingly, the system performance can improve if the error bound is increased to a certain extent, depending on the number of sensors in the field. For example, from Table VII one can see that for $n = 18$ if ϵ_{ij} goes from 0 up to 0.75m, the coverage increases. For this specific case, the best result is achieved for $\epsilon_{ij} = 0.75m$ which is relatively high when compared with the sensors radii. When the number of sensors increases the best performance is attained for smaller error magnitudes, e.g., 0.1m for $n = 36, 45$. However, for any n the performance with some mild errors is better than that for the based on the exact information where $\epsilon_{ij} = 0$ and $\epsilon_{ii} = 0$. Similar behavior is seen for the

other algorithms. Specifically, Table IV in [36] reports such results for the FPGAW.

The key to better understand the effect of the magnitude of the error bounds in the above results, is the fact that assuming a non-zero error bound for some sensor's location (i.e., $\epsilon_{ij} \neq 0$) our algorithms do not partition the field, meaning that, there remain regions in the field that are not assigned to any of the sensors, as shown in dark blue in Fig. 1. This, however, does not imply that such regions will not be covered, necessarily. It means that, compared to the case where there was no error bound (i.e., $\epsilon_{ij} = 0$), the sensors will target a smaller region to cover, and thus they are more likely to cover this region, and satisfy the algorithm termination criterion, in fewer iterations. With this in mind, one can see that if the bounds are very big, the sensors' regions shrink largely so that they can easily cover this small region without covering the region corresponding to $\epsilon_{ij} = 0$ to a good extent. Under such circumstances, one can expect the proposed algorithms perform poorly, and the total coverage be less than the case with no error. Simulation results confirm this, as for any n the coverage corresponding to $\epsilon_{ij} = 1.5m$ is smaller than that of the first column ($\epsilon_{ij} = 0$), in Table VII. When the magnitude of the error bound is relatively small,³ the sensors have bigger regions and higher degrees of freedom to move, and can be placed in a better position such that maximize the local coverage. In the extreme case of $\epsilon_{ij} = 0$, although the regions are the largest (they partition the field), the sensors are more prone to cover the same area too. That is, even though the individual coverage can be more than the case with $\epsilon_{ij} \neq 0$, the overall coverage could be less due to the possible overlaps between the sensor's covered areas. For example, in the case of the farthest point based algorithms, if the farthest point of two neighboring sensors is the same point, they both move toward that point and this increases the chance that the area around that point be covered by both sensors. An error bound, however, creates a neutral region between the sensors (see Fig. 1), eliminating the common boundaries. So, even if two sensors move toward each other the area of the overlapped region would be smaller. Therefore, in general, we expect to have smaller overlapped regions throughout the field.

In view of the above results and the intuition we provided, even when the locations of the sensors are exactly known at each node, it makes sense to assume some bounded errors for them and exploit the proposed guaranteed Voronoi-based diagrams to improve the system performance. The magnitude of the optimal bound depends on other parameters such as the density of sensors in the field and their sensing radii, and it decreases as the number of sensors goes up. Note that, although the system performance improves by considering a certain amount of error, this improvement is at the cost of more complexity in computation of the sensors' regions.

VII. CONCLUSIONS

We have developed three Voronoi-based diagrams to increase the sensing coverage of mobile sensor networks with

³This comparison is with respect to the area assigned to the sensors. So for a given field, when n increases the regions assigned to the sensors becomes smaller, on average, and the error bounds should be smaller too.

TABLE VII
THE FINAL COVERAGE PERCENTAGE FOR DIFFERENT MEASUREMENT ERRORS BASED ON THE FPGMW ALGORITHM WITH $\epsilon_{ii} = 0.1m$.

	$\epsilon_{ij} = 0$	$\epsilon_{ij} = 0.1m$	$\epsilon_{ij} = 0.25m$	$\epsilon_{ij} = 0.5m$	$\epsilon_{ij} = 0.75m$	$\epsilon_{ij} = 1m$	$\epsilon_{ij} = 1.5m$
$n = 18$	72.61%	72.12%	72.03%	73.13%	73.30%	73.06%	72.14%
$n = 27$	89.30%	89.79%	90.02%	89.11%	88.42%	87.40%	85.88%
$n = 36$	97.16%	97.26%	96.85%	96.10%	95.03%	94.07%	92.37%
$n = 45$	97.84%	97.92%	97.91%	97.3%	96.82%	96.23%	94.36%

non-identical sensing radii, in the presence of location estimation errors. In all of these algorithms, the search for coverage holes is performed independently in each cell, to keep the complexity of the algorithms low. We have then used two movement strategies, namely, the farthest point and minmax point, within the cells developed by the proposed diagrams to relocate the sensors in order to increase the networks sensing coverage. Numerical results show the effectiveness of the proposed diagrams and deployment algorithms in increasing the network's coverage in the presence of localization error. By assuming some error bounds for the location information, the boundaries of the sensors regions are separated to reduce the area of overlapping coverage regions, and thus to increase the total coverage when compared to the case where the exact location information is applied.

VIII. APPENDIX

A. Proof of Proposition 3

Proof: Let $\alpha \triangleq \frac{|d(q, p_i)|}{|d(q, p_j)|}$. First, suppose $\alpha \neq 1$. We can write

$$\begin{aligned} |d(q, p_i)| &= \alpha |d(q, p_j)| \\ \Leftrightarrow d^2(q, p_i) &= \alpha^2 d^2(q, p_j) \\ \Leftrightarrow (x - x_{p_i})^2 + (y - y_{p_i})^2 &= \alpha^2 [(x - x_{p_j})^2 + (y - y_{p_j})^2] \\ \Leftrightarrow \left(x - \frac{x_{p_i} - \alpha^2 x_{p_j}}{1 - \alpha^2}\right)^2 + \left(y - \frac{y_{p_i} - \alpha^2 y_{p_j}}{1 - \alpha^2}\right)^2 &= R^2, \end{aligned} \quad (11)$$

$R = \frac{\alpha}{1 - \alpha^2} \sqrt{(x_{p_i} - x_{p_j})^2 + (y_{p_i} - y_{p_j})^2} = \frac{\alpha}{1 - \alpha^2} d(p_i, p_j)$. In this proof, the first two steps are obvious by inspection. The last step follows simple, but a bit cumbersome, algebra. For $\alpha = 1$, which implies $r_i = r_j$, the MWVD simplifies to the VD and we know that the locus is a line (the perpendicular bisector of the segment connecting p_i to p_j). ■

REFERENCES

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.
- [2] Z. Zou, Y. Bao, H. Li, B. F. Spencer, and J. Ou, "Embedding compressive sensing-based data loss recovery algorithm into wireless smart sensors for structural health monitoring," *IEEE Sensors Journal*, vol. 15, no. 2, pp. 797–808, 2015.
- [3] C.-Y. Chang, C.-T. Chang, Y.-C. Chen, and H.-R. Chang, "Obstacle-resistant deployment algorithms for wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 6, pp. 2925–2941, 2009.
- [4] F. Ingelrest, G. Barrenetxea, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "Sensorscope: Application-specific sensor network for environmental monitoring," *ACM Transactions on Sensor Networks*, vol. 6, no. 2, pp. 1–32, 2010.
- [5] Z. Shen, Y. Chang, H. Jiang, Y. Wang, and Z. Yan, "A generic framework for optimal mobile sensor redeployment," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 8, pp. 4043–4057, 2010.
- [6] S. Susca, F. Bullo, and S. Martinez, "Monitoring environmental boundaries with a robotic sensor network," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 2, pp. 288–296, 2008.
- [7] X. Li, W. Shu, M. Li, H.-Y. Huang, P.-E. Luo, and M.-Y. Wu, "Performance evaluation of vehicle-based mobile sensor networks for traffic monitoring," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 1647–1653, 2009.
- [8] G. Isbitiren and O. B. Akan, "Three-dimensional underwater target tracking with acoustic sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 8, pp. 3897–3906, 2011.
- [9] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wireless communications*, vol. 11, no. 6, pp. 6–28, 2004.
- [10] G. Wang, G. Cao, and T. La Porta, "Movement-assisted sensor deployment," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 640–652, 2006.
- [11] S. Megerian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Worst and best-case coverage in sensor networks," *IEEE Transactions on Mobile Computing*, vol. 4, no. 1, pp. 84–92, 2005.
- [12] Q. Du, V. Faber, and M. Gunzburger, "Centroidal Voronoi tessellations: Applications and algorithms," *SIAM review*, vol. 41, no. 4, pp. 637–676, 1999.
- [13] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, pp. 1327–1332, 2002.
- [14] H. Mahboubi, J. Habibi, A. Aghdam, and K. Sayrafian-Pour, "Distributed deployment strategies for improved coverage in a network of mobile sensors with prioritized sensing field," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 451–461, 2013.
- [15] H. Mahboubi, K. Moezzi, A. G. Aghdam, and K. Sayrafian-Pour, "Distributed deployment algorithms for efficient coverage in a network of mobile sensors with nonidentical sensing capabilities," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 8, pp. 3998–4016, 2014.
- [16] H. Mahboubi, K. Moezzi, A. Aghdam, K. Sayrafian-Pour, and V. Marbukh, "Self-deployment algorithms for coverage problem in a network of mobile sensors with unidentical sensing range," in *Proc. Globecom*, pp. 1–6, 2010.
- [17] L. Hu and D. Evans, "Localization for mobile sensor networks," in *Proceedings of the 10th annual international conference on Mobile computing and networking*, pp. 45–57, ACM, 2004.
- [18] G. Han, H. Xu, T. Q. Duong, J. Jiang, and T. Hara, "Localization algorithms of wireless sensor networks: a survey," *Telecommunication Systems*, vol. 52, no. 4, pp. 2419–2436, 2013.
- [19] F. Sharifi, Y. Zhang, and A. Aghdam, "A distributed deployment strategy for multi-agent systems subject to health degradation and communication delays," *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1–4, pp. 623–633, 2014.
- [20] S. Ghiasi, A. Srivastava, X. Yang, and M. Sarrafzadeh, "Optimal energy aware clustering in sensor networks," *Sensors*, vol. 2, no. 7, pp. 258–269, 2002.
- [21] X. Wang, S. Han, Y. Wu, and X. Wang, "Coverage and energy consumption control in mobile heterogeneous wireless sensor networks," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 975–988, 2013.
- [22] E. Deza and M. M. Deza, *Encyclopedia of Distances*. Springer, 2009.
- [23] H. Mahboubi and A. G. Aghdam, "An energy-efficient strategy to improve coverage in a network of wireless mobile sensors with nonidentical sensing ranges," in *Proc. IEEE 77th Vehicular Technology Conference (VTC Spring)*, pp. 1–5, 2013.
- [24] H. Mahboubi, M. Vaezi, and F. Labeau, "Mobile sensors deployment subject to measurement error," in *Proc. IEEE 80th Vehicular Technology Conference (VTC Fall)*, pp. 1–5, 2014.
- [25] R. C. James and G. James, *The Mathematics Dictionary*. Springer, 1992.
- [26] O. Setter, M. Sharir, and D. Halperin, "Constructing two-dimensional Voronoi diagrams via divide-and-conquer of envelopes in space," *Transactions on Computational Science IX*, p. 1, 2010.

- [27] H. Mahboubi, K. Moezzi, A. G. Aghdam, K. Sayrafian-Pour, and V. Marbukh, "Distributed deployment algorithms for improved coverage in a network of wireless mobile sensors," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 163–174, 2014.
- [28] H. Mahboubi, K. Moezzi, A. Aghdam, K. Sayrafian-Pour, and V. Marbukh, "Distributed deployment algorithms for improved coverage in a network of wireless mobile sensors," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 163–174, 2014.
- [29] R. S. Chang and S. H. Wang, "Self-deployment by density control in sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 3, pp. 1745–1755, 2008.
- [30] A. Gallais, J. Carle, D. Simplot-Ryl, and I. Stojmenovic, "Localized sensor area coverage with low communication overhead," *IEEE Transactions on Mobile Computing*, vol. 7, no. 5, pp. 661–672, 2008.
- [31] C. H. Caicedo-Nez and M. Zefran, "Distributed task assignment in mobile sensor networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2485–2489, 2011.
- [32] K. Ma, Y. Zhang, and W. Trappe, "Managing the mobility of a mobile sensor network using network dynamics," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 1, pp. 106–120, 2008.
- [33] G. Wang, G. Cao, P. Berman, and T. F. La Porta, "Bidding protocols for deploying mobile sensors," *IEEE Transactions on Mobile Computing*, vol. 6, no. 5, pp. 563–576, 2007.
- [34] S. Yoon, O. Soysal, M. Demirbas, and C. Qiao, "Coordinated locomotion and monitoring using autonomous mobile sensor nodes," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 10, pp. 1742–1756, 2011.
- [35] M. Rahimi, H. Shah, G. S. Sukhatme, J. Heideman, and D. Estrin, "Studying the feasibility of energy harvesting in a mobile sensor network," in *Proc. IEEE International Conference on Robotics and Automation*, vol. 1, pp. 19–24, 2003.
- [36] H. Mahboubi, M. Vaezi, and F. Labeau, "Distributed deployment algorithms in a network of nonidentical mobile sensors subject to location estimation error," in *Proc. IEEE Sensors*, pp. 1–4, 2014.



Mojtaba Vaezi (S'09, M'14) is a postdoctoral research fellow in the Department of Electrical Engineering at Princeton University. He received the Ph.D. degree from McGill University in 2014, and B.Sc. and M.Sc. degrees from Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran, all in Electrical Engineering. Mojtaba has served as the president of McGill IEEE Student Branch during 2012–2013. Before joining McGill in 2009, Mojtaba was the Head of Mobile Radio Network Design and Optimization Group at Ericsson Iran. His research interests lie at the intersection of signal processing, communications and information theory. Mojtaba is a recipient of a number of academic, leadership, and research awards, including the McGill Engineering Doctoral Award (MEDA) for 2009–2012, IEEE Larry K. Wilson Regional Student Activities Award in 2013, and the Natural Sciences and Engineering Research Council of Canada (NSERC) Postdoctoral Fellowship in 2014.



Hamid Mahboubi (S'08, M'14, SM'15) is a recipient of the Governor General of Canada Academic Gold Medal in 2015. He is also a recipient of the Gold Medal in the 1999 National Math Olympiad in Iran. He was awarded an honorary admission to Sharif University of Technology, Tehran, Iran, where he received his B.Sc. degree in Electrical Engineering in 2003. He received his M.A.Sc. degree in Electrical and Computer Engineering from the University of Tehran, Tehran, Iran, in 2006. He received his Ph.D. degree in 2014 from Concordia University,

Montreal, Canada. He is currently a postdoctoral fellow in the Department of Electrical and Computer Engineering at McGill University, Montreal, Canada. Dr. Mahboubi is the recipient of Natural Sciences and Engineering Research Council of Canada (NSERC) Postdoctoral Fellowship, Concordia University Distinguished Doctoral Dissertation Prize in Engineering and Natural Science, Doctoral Prize in Engineering and Computer Science, Fonds qubcois de la recherche sur la nature et les technologies (FQRNT) Post-Doctoral Award, Bourse d'Etudes Hydro Quebec Scholarship, Power Corporation of Canada Graduate Fellowship, and Canadian National Award in Transportation.

Dr. Mahboubi has served as Chair of the Control Systems Chapter of the IEEE Montreal Section since January 2012. He is a member of Editorial Board of IEEE SigView (IEEE Signal Processing Society) and a senior member of the IEEE. He also was a member of the Technical Program Committee of 2015 IEEE International Conference on Wireless for Space and Extreme Environments. His research interests include mobile sensor networks, multi-agent systems, hybrid systems, networked control systems, smart grids, and optimization.



Fabrice Labeau is an Associate Professor with the Electrical and Computer Engineering Department, McGill University, where he holds the NSERC/Hydro-Quebec Industrial Research Chair in Interactive Information Infrastructure for the Power Grid. From 1996 to 2000, he was with the Communications and Remote Sensing Laboratory, Université Catholique de Louvain (UCL), Belgium. His research interests are in applications of signal processing, to healthcare, power grids, communications and signal compression. He has authored more than 150 papers in refereed journals and conference proceedings in these areas. He has held several administrative and management positions at McGill University, including Associate department Chair, Associate Dean, Interim Chair and Acting Dean. He currently is the Associate Dean for Faculty Affairs at the McGill Faculty of Engineering. He is currently the elected President of the IEEE Vehicular Technology Society (VTS). He was part of the organizing committee of ICASSP 2004 in Montreal and is/was Technical Program Committee co-chair for the IEEE Vehicular Technology Conference in the Fall of 2006 and 2012, and the IEEE International Conference on Image Processing 2015. He is a Senior Member of IEEE.