

## Research Article

# Distributed Voronoi-Based Self-Redeployment for Coverage Enhancement in a Mobile Directional Sensor Network

Tien-Wen Sung and Chu-Sing Yang

*Department of Electrical Engineering, Institute of Computer and Communication Engineering, National Cheng Kung University, No.1, University Road, Tainan City 701, Taiwan*

Correspondence should be addressed to Tien-Wen Sung; [tienwen.sung@gmail.com](mailto:tienwen.sung@gmail.com)

Received 28 June 2013; Revised 16 September 2013; Accepted 17 September 2013

Academic Editor: Shengming Jiang

Copyright © 2013 T.-W. Sung and C.-S. Yang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A directional sensor network is different from conventional wireless sensor networks. It uses directional sensors instead of omnidirectional ones in the network for different applications, and the effective sensing range is characterized by directionality and size-specific sensing angle. Therefore, conditions of directional sensor networks are dissimilar to those of generic wireless sensor networks for researches, especially on the sensing coverage. This study proposed a distributed approach to enhance the overall field coverage by utilizing mobile and direction-rotatable sensors in a directional sensor network. The algorithm makes sensors self-redeploy to the new location and new direction without global information by utilizing the features of geometrical Voronoi cells. Simulations were used to evaluate and prove the effectiveness of the proposed algorithm. The results show that the approach contributes to significant field coverage improvement in directional sensor networks.

## 1. Introduction

In a wireless sensor network (WSN) [1], the sensing coverage is always one of the key factors to ensure that sensing tasks will be well performed. The coverage ratio becomes a fundamental index of the measurement for WSN quality of service (QoS) [2]. Many studies related to the subject of WSN coverage have been successively proposed [3]. In recent years, directional sensor networks (DSNs) have drawn the attention of researchers. Differing from the conventional WSNs which use omnidirectional scalar sensors such as temperature or humidity sensors, a DSN is likely to be equipped with directional sensors such as image or video sensors. There are different problems and conditions regarding the coverage issues should be considered in the researches for DSNs due to the limited effective sensing range characterized by directionality and size-specific sensing angle [4, 5]. In the sensing field of a DSN, the sensing coverage depends on not only the location but also on the sensing direction and sensing angle of each sensor [6].

Although there are some studies concerning coverage of DSN [7], the problems and solutions are different. For

example, several of the studies focused on the target coverage and several focused on the overall field coverage; several aimed at solving  $k$ -coverage problem and several aimed at solving prioritized region coverage. For another example, several studies considered the obstacles in a sensing field and several considered an obstacle-free sensing environment. This paper proposes a distributed self-redeployed algorithm to deal with the enhancement of overall coverage ratio in the sensing field of a DSN which consists of mobile and rotatable directional sensors. The characteristics of geometrical Voronoi diagram were utilized to determine the target positions and sensing directions of sensors and reduce the decision-making complexity. Using Voronoi diagram in this study was motivated by the following advantages: (1) Voronoi diagram exactly can be generated from a set of sensors and each sensor is associated with only one Voronoi cell; (2) the cell-based structure can help design a localized/distributed algorithm without global information; (3) each sensor can only check its own cell to examine the coverage hole; (4) both Voronoi vertices and edges can help make the decision of sensor position and sensing direction; (5) for a mobile sensor, the moving distance can be confined to the cell;

and (6) the construction of a Voronoi cell is independent of sensing radii and angles; this helps a distributed algorithm be applicable to those sensors which have varied radii and angles. More detailed introduction of Voronoi diagram is described in Section 3.2.

The rest of the paper is organized as follows. Section 2 briefs the previous works related to DSN coverage. Section 3 describes the preconditions and assumptions. In Section 4, we provide the readers with our proposed scheme. Section 5 evaluates and analyzes the efficiency of the scheme. Finally, we conclude this paper in Section 6.

## 2. Related Works

There are four major categories of research subjects in WSNs [8]: (1) sensing coverage, (2) network connectivity, (3) network longevity, and (4) data fidelity. These subjects are also the fundamental requirements in DSN, but the existing WSN solutions or results are not completely suitable for DSNs, particularly the coverage issue. This is mainly due to that the directional sensors are characterized by working direction, angle of view (AOV), and line of sight (LOS) [9], and new solutions for the dissimilar problems are required [10].

In regard to DSN coverage problems, the related studies can be divided into two categories [11]: (1) known targets coverage and (2) region coverage. The former determines a subset of the sensors to attain coverage on the specific targets or positions, and the latter makes the sensors attain a full or certain ratio of sensing coverage in the task region. Concerning known-targets coverage, Chow et al. [12] proposed an algorithm to select multiple directional sensors for providing 360° complete monitoring of a target while one sensor can only cover a part of the target. Wang et al. [13] aimed at prioritized targets and used the genetic algorithm to obtain a minimum subset of directional sensors for covering all of the targets. Hsu et al. [14] proposed an algorithm to solve the problem of which the number of covered targets is maximized whereas the rotated angle of sensors is minimized. And concerning region coverage, Jing and Jian-Chao [15] used the coverage overlap between the adjacent sensors as the quantity of electric charge and then used Coulomb's inverse-square law to change the sensor direction for reducing the overlapped coverage. Liang et al. [16] aimed at DSNs with mobile sensors, so that sensors could move to appropriate positions and obtain a good total coverage. Guvensan and Yavuz [17] used a hybrid solution of mixing stationary sensors, motile sensors, and mobile sensors in a DSN to increase the sensing field coverage. Tezcan and Wang [18] considered the condition of obstacles within the sensing field and proposed an algorithm utilizing rotatable sensors to reduce the influence of obstacles on the coverage. Huang et al. [19] focused on the multimedia image sensor networks and proposed a virtual potential field-based method with the considerations of sensor direction and movement for the coverage enhancement. The virtual force causes the adjustment of angular magnitude to be a trouble in coverage problem; therefore, linear-relation-based algorithm (LRBA) and mechanism-based approximate

algorithm (MBAA) was proposed. These algorithms need the process of pairing between two adjacent nodes and a defect exists in the algorithms. That is, some nodes could not find their respective paired partner. These isolated sensors cannot perform the algorithms for the coverage contribution. Ma and Liu [20] analyzed deployment strategies and proposed a group-based strategy, namely, grouping scheduling protocol (GSP) for satisfying given coverage probability requirement in a directional sensor network. It needs repairing processes if certain grouped sensors are incommunicable to the sink. This needs the deployment of more sensors; therefore, the average coverage ratio of the grouped sensors could be decreased.

Voronoi-based method has been utilized in WSNs, but it has not drawn much attention of researchers on the DSN coverage problems. Li et al. [21] proposed the Voronoi-based distributed approximation (VDA) algorithm to make sensors cover the Voronoi edges as more as possible. The study approximately considers that if most Voronoi edges are covered, then most area will be covered; however, this is not definite and may cause more coverage overlap. This paper proposes a new distributed approach to determine location and direction movements of the mobile and rotational sensors for obtaining significant coverage contribution in the DSN. In addition, the geometrical features of Voronoi cells with its vertices, edges, and included angles were utilized in the proposed approach to assist in the determination of sensor self-redeployment.

## 3. Preliminaries

*3.1. Assumptions.* There are several related studies [16, 21, 22] assume that all the directional sensors have the same sensing radius and sensing angle range, which means that the sensors are homogeneous. The algorithm proposed in this paper does not have this limitation. It is applicable to sensors with different sensing radii and angles. But the general assumptions are listed as follows:

- (1) Each sensor is well aware of its coordinate by utilizing a certain localization technology [23].
- (2) Each sensor has enough communication range or multihop transmission capacity to transmit information to neighbor sensors.
- (3) Sensors are rotatable; they can do a clockwise or counterclockwise rotation to change the working direction.
- (4) Sensors are mobile; they can move within the sensing field. This assumption is not unrealistic in the real world [24, 25].

*3.2. Voronoi Diagram.* The Voronoi diagram is a computational geometry data structure with special characteristics [26], which is applicable to be utilized in the proposed algorithm to divide the sensing field into cells. The sensing field will be divided into Voronoi cells according to the initial positions of the deployed sensors, as shown in Figure 1. Given a set of  $n$  sensors  $s_1, s_2, \dots, s_n$  in the sensing field, the one and only one (unique) Voronoi diagram can be constructed

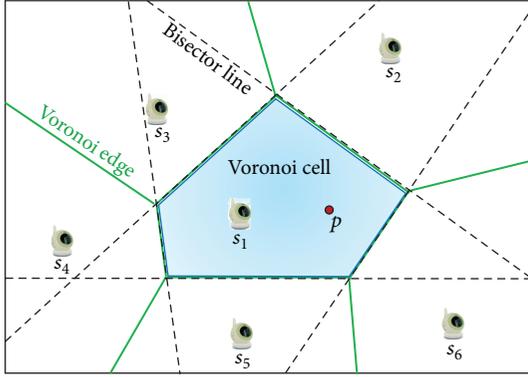


FIGURE 1: Construction of the Voronoi cells.

by drawing the perpendicular bisector of line segment of each sensor pair. Those bisector line segments form the boundaries of Voronoi cells and are called Voronoi edges. The endpoints of these edges are called Voronoi vertices. Finally, the sensing field is divided into  $n$  Voronoi cells, which meets the following two major properties.

- (1) Each sensor  $s_i$  lies in exactly one cell.
- (2) If a point  $p$  lies in the same cell as  $s_i$ , then the Euclidian distance from  $p$  to  $s_i$  will be shorter than the one from  $p$  to  $s_j$ , where  $s_j$  is any other sensor.

Regarding the construction of a Voronoi Diagram, there are several algorithms to generate the diagram from a set of points, such as Bowyer-Watson algorithm with an  $O(n^2)$  time complexity and the best known Fortune algorithm [27] with an  $O(n \log n)$  time complexity. They can be used to construct a global Voronoi diagram. In this study, we do not need to construct a global Voronoi diagram. Our proposed algorithm is a distributed method and each sensor only needs to construct one cell of the Voronoi diagram. The Voronoi diagram of a set of points (sensors) is unique (one and only one) [28], which consists of Voronoi cells. In our study, each distributed sensor broadcasts its coordinates as local information to the neighbors. A sensor constructs a bisector line between itself and the neighbor whenever it receives the coordinates of the neighbor. And finally, the sensor will be in a convex polygon, in which the polygon consists of certain line segments of those bisector lines. And the polygon is called a Voronoi cell associated with the sensor. No other sensors will appear in the cell. Any edge of the cell is a bisector line between the associated sensor and its one neighbor, and this edge is also an edge of the cell of that neighbor. It should be noted that a Voronoi diagram consists of the Voronoi cells associated with a given set of points is unique, which had been proven in [28].

**3.3. Directional Sensing Model.** Figure 2 shows the direction-rotatable sensing model for the directional sensors in the proposed algorithm of this paper. The notations and parameters are listed in Table 1.

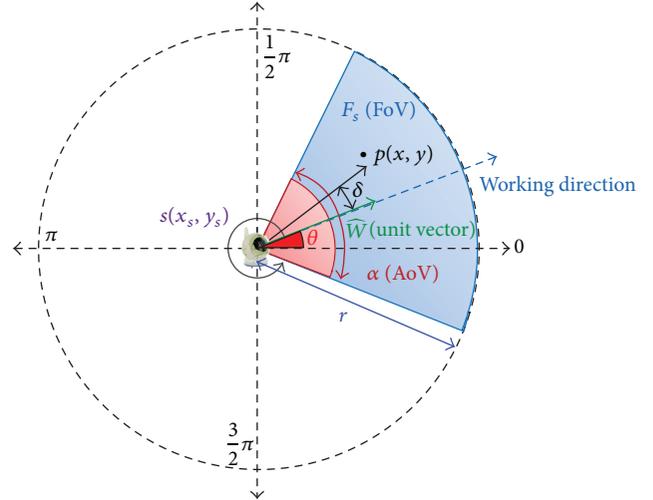


FIGURE 2: Sensing model for direction-rotatable sensors.

TABLE 1: Notations and parameters for the directional sensing model.

Notation/parameter	Description
$s(x_s, y_s)$	The directional sensor with its coordinate
$F_s$	The effective sensing field of $s$ ; the region covered by $s$ ; field of view (FoV)
$\alpha$	The angular size of sensing range of $s$ ; angle of view (AoV)
$r$	The sensing radius of $s$
$\theta$	The working direction of $s$ , defined as the angle value relative to the positive $x$ -axis of the sensing direction vector; $\theta$ is between 0 and $\pm\pi$ .
$\hat{w}$	The unit vector of the working direction
$p(x, y)$	A point with its coordinate covered by $s$
$\delta$	The included angle between $\vec{sp}$ and $\hat{w}$

The effective sensing field  $F_s$  is in sector shape. A point  $p$  is said to be covered by the sensor  $s$  if and only if the following two conditions are satisfied.

- (1) The Euclidean distance between  $p(x, y)$  and  $s(x_s, y_s)$  is less than or equal to the sensing radius ( $r$ ) of  $s$ . Consider

$$\sqrt{(x - x_s)^2 + (y - y_s)^2} \leq r. \quad (1)$$

- (2) The included angle ( $\delta$ ) between  $\vec{sp}$  and the working direction unit vector ( $\hat{w}$ ) is less than or equal to the half of the AoV ( $\alpha$ ). Consider

$$\vec{sp} \cdot \hat{w} = \|\vec{sp}\| \|\hat{w}\| \cos \delta \quad (2)$$

$$\begin{aligned} &\Rightarrow (x - x_s) \cos \theta + (y - y_s) \sin \theta \\ &\geq \sqrt{(x - x_s)^2 + (y - y_s)^2} \cos \left( \frac{\alpha}{2} \right). \end{aligned} \quad (3)$$

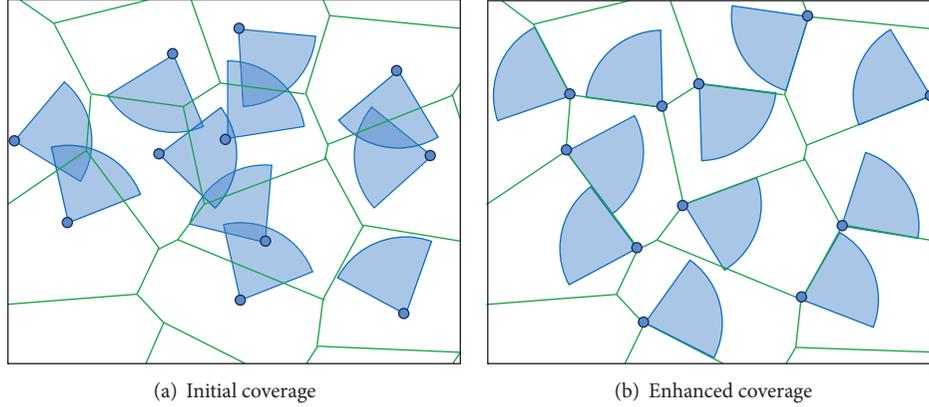


FIGURE 3: A schematic example of results before and after execution of the algorithm.

In brief, the directional sensing model shown in Figure 2 illustrates that a point  $p(x, y)$  is covered by the sensor  $s$ ; that is, to say,  $(x, y) \in F_s$ , if and only if (1) and (3) are satisfied with the given parameters  $r, \theta, \alpha$ , and sensor coordinate  $(x_s, y_s)$ .

#### 4. Distributed Voronoi-Based Self-Redeployment Algorithm

**4.1. Problem Description.** In a sensing field on which directional sensors were numerous and randomly deployed, it is difficult to obtain the optimal field coverage since partial regions are not covered and some regions are overlapped. The chief defect in using centralized algorithms to solve this problem is that the multihop transmissions over the network for collecting global information will be heavy on resources. On the contrary, a distributed algorithm does not need the global information, but only local information used for solutions. In addition, a distributed algorithm is more likely to make a real-time response or quick adjustment in changes of environment. Therefore, this paper proposes a distributed Voronoi-based self-redeployment algorithm, “DVSA” for short, to improve the field coverage ratio for a mobile and rotatable directional sensor network. And, it can be applied to sensors with different sensing radii and AoVs.

Each sensor constructs its own Voronoi cell according to the position information of their neighbor sensors after the initial deployment. By evaluating the suitability of each vertex with the parameters of cell structure and sensing model, one will be selected as the new location of the sensor (associated with the cell), and a new working direction also will be decided. The suitability evaluation aims to reduce the probability of coverage overlap across the cells. Figure 3 shows a schematic example of coverage comparison between the results before and after execution of the algorithm.

**4.2. Target Location of Sensor Movement.** Figure 4 shows an initially deployed directional sensor  $s$  and its associated Voronoi cell  $C_s$ . The sensor coordinate  $(x_s, y_s)$  will be known after initial sensor deployment and the sensing parameters  $\alpha$  and  $r$  are also given. In addition, the coordinates of all vertices of the cell will be obtained after the construction of local

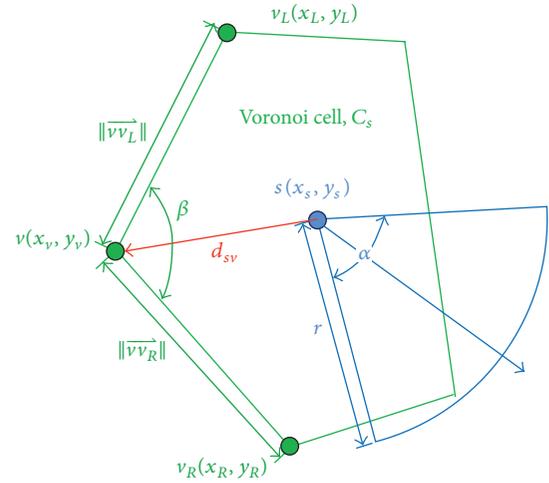


FIGURE 4: Included angle of a Voronoi vertex and related parameters.

Voronoi cell. Let  $V$  be the set of all vertices of  $C_s$ . For each  $v \in V$ ,  $v_L$  and  $v_R$  means the left adjacent vertex and right adjacent vertex of  $v$ , and their coordinates  $(x_L, y_L)$  and  $(x_R, y_R)$  are known. According to the geometric definition of inner product of two vectors in linear algebra, the included angle of  $\vec{vv}_L$  and  $\vec{vv}_R$ , notated as  $\beta$ , can be derived from

$$\begin{aligned} \vec{vv}_L \cdot \vec{vv}_R &= \|\vec{vv}_L\| \|\vec{vv}_R\| \cos \beta \\ \beta &= \cos^{-1} \frac{\vec{vv}_L \cdot \vec{vv}_R}{\|\vec{vv}_L\| \|\vec{vv}_R\|} \\ &= \cos^{-1} \frac{(x_L - x_v)(x_R - x_v) + (y_L - y_v)(y_R - y_v)}{\sqrt{(x_L - x_v)^2 + (y_L - y_v)^2} \sqrt{(x_R - x_v)^2 + (y_R - y_v)^2}}. \end{aligned} \quad (4)$$

The desired choice of a target vertex for the sensor movement is the vertex that its corresponding included angle is larger than or equal to the AoV of the sensor and the lengths of its two edges are larger than or equal to the sensing radius of the sensor. If there is more than one choice in this case,

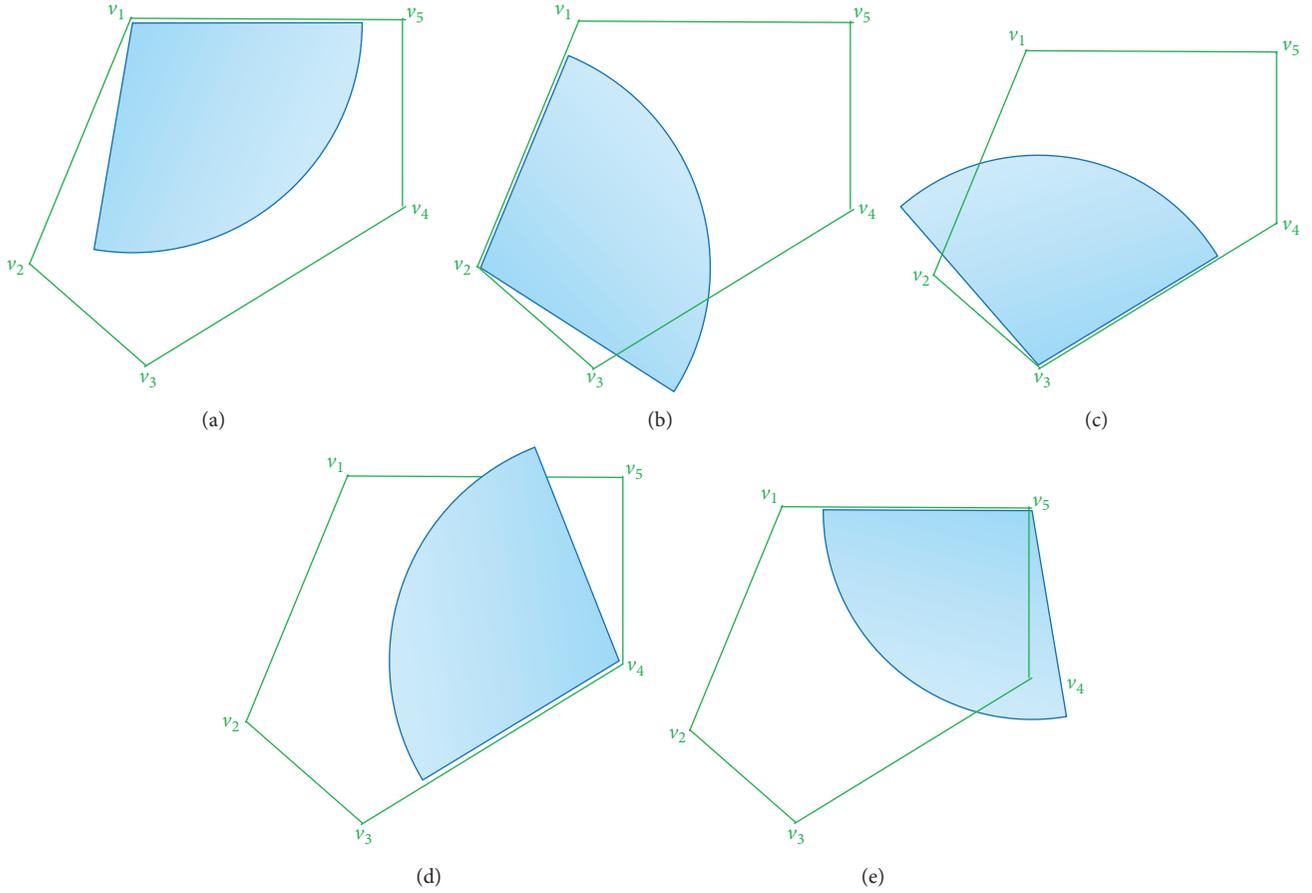


FIGURE 5: Results of choosing different vertices as the target location.

the vertex with the largest included angle is selected from the candidates, as shown as follows:

$$T = \{v \in V \mid \beta \geq \alpha, \|\overrightarrow{vv_L}\| \geq r, \|\overrightarrow{vv_R}\| \geq r\} \quad (5)$$

$$t = \arg \max_{v \in T} \beta,$$

where  $T$  is the set of target vertex candidates with the highest priority, and  $t$  is the selected target vertex. If  $T$  is empty, the second preferred choice of target vertex is the vertex that its included angle is larger than or equal to the AoV of the sensor, but only one edge is larger than or equal to the sensing radius  $r$ . Similarly, if there is more than one choice in this priority level, the vertex with largest included angle is selected from the new candidates. On the contrary, if there is neither vertex has an included angle  $\beta \geq \alpha$  nor an edge larger than or equal to  $r$ , the one with largest included angle is selected from these lowest-priority candidates. Figure 5 shows an example with the results of choosing different vertices as the target location. The vertex  $v_1$  will be selected as the new location of the sensor. The moving distance of the sensor for moving to the target vertex  $v$  is as (6), and this movement will certainly be within the limitation of cell boundary. Consider

$$d_{sv} = \sqrt{(x_s - x_v)^2 + (y_s - y_v)^2}. \quad (6)$$

In this phase, each sensor in the DSN selects a suitable vertex from its own local Voronoi cell for obtaining a higher *intracell* coverage ratio. This will be beneficial to avoid coverage overlap across the cells as much as possible and keep higher overall coverage ratio of the sensing field. The detail of the algorithm for this phase is shown as Algorithm 1.

**4.3. Rotation of Working Direction.** The target location of sensor movement is selected under the criteria described in previous section. After the directional sensor has moved to the suitable vertex on its local Voronoi cell, the next is to decide a working direction that the direction should be controlled to make sensor contribute to full *intracell* coverage as much as possible. By this rule, the coverage overlaps across the cells will be decreased, and it benefits the overall filed coverage.

Figure 6 shows an example model to control the working direction. The sensor is at the position of vertex  $v$  with coordinate  $(x_v, y_v)$ . The working direction, notated as  $\theta$ , should be controlled in the range limited by the angle values  $\theta_1$  and  $\theta_2$ , and then the sensing coverage of the sensor will not go beyond the cell edges  $e_R$  or  $e_L$ .  $\theta_1$  is the most right limit of direction for reserving the *intracell* coverage, and  $\theta_2$  is the most left limit. Because the sensor is direction rotatable and the FoV of the sensor should not go beyond the cell edges of

```

get self-coordinate  $(x_s, y_s)$ 
broadcast  $(x_s, y_s)$  to neighbor sensors
receive coordinates from neighbors
construct local Voronoi cell  $C_s = (V, E)$ 
   $V = \{v_1, v_2, \dots, v_m\}$  is the set of vertices of the cell
   $E = \{e_1, e_2, \dots, e_n\}$  is the set of edges of the cell
 $T_2 = \phi; T_1 = \phi; T_0 = \phi;$ 
for each  $v(x_v, y_v) \in V$ 
{
  let  $v_L(x_L, y_L)$  is the left adjacent vertex of  $v$ 
  let  $v_R(x_R, y_R)$  is the right adjacent vertex of  $v$ 
   $\beta_v = \frac{(x_L - x_v)(x_R - x_v) + (y_L - y_v)(y_R - y_v)}{\sqrt{(x_L - x_v)^2 + (y_L - y_v)^2} \sqrt{(x_R - x_v)^2 + (y_R - y_v)^2}}$  is the included angle of  $\vec{v v_L}$  and  $\vec{v v_R}$ 
  if  $(\beta_v \geq \alpha)$ 
    if  $(\|\vec{v v_L}\| \geq r \text{ and } \|\vec{v v_R}\| \geq r)$ 
       $T_2 = T_2 \cup \{v\}$ 
    else
      if  $(\|\vec{v v_L}\| < r \text{ and } \|\vec{v v_R}\| < r)$ 
         $T_0 = T_0 \cup \{v\}$ 
      else
         $T_1 = T_1 \cup \{v\}$ 
      endif
    endif
  else
     $T_0 = T_0 \cup \{v\}$ 
  endif
}
if  $(T_2 \neq \phi)$ 
   $t = \arg \max_{v \in T_2} \beta_v$ 
else
  if  $(T_1 \neq \phi)$ 
     $t = \arg \max_{v \in T_1} \beta_v$ 
  else
     $t = \arg \max_{v \in T_0} \beta_v$ 
  endif
endif
endif

```

ALGORITHM 1: Target location selection for sensor movement.

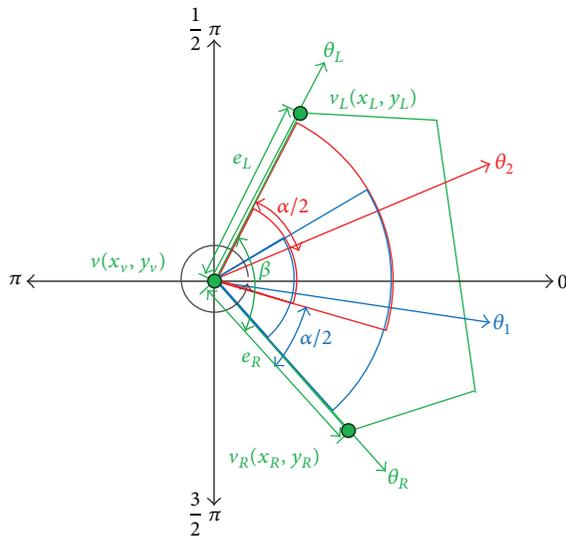


FIGURE 6: Rotation of working direction.

$e_R$  and  $e_L$ , the angle values  $\theta_R$  and  $\theta_L$  of the vectors  $\vec{v v_R}$  and  $\vec{v v_L}$  relative to the positive  $x$ -axis are calculated as the following (7), respectively. The values of  $\theta_R$  and  $\theta_L$  are between 0 and  $\pm\pi$ . Consider

$$\theta_R = \begin{cases} \sin^{-1} \frac{y_R - y_v}{\sqrt{(x_R - x_v)^2 + (y_R - y_v)^2}}, & x_R > x_v \\ \pi - \sin^{-1} \frac{y_R - y_v}{\sqrt{(x_R - x_v)^2 + (y_R - y_v)^2}}, & x_R < x_v, y_R > y_v \\ -\pi - \sin^{-1} \frac{y_R - y_v}{\sqrt{(x_R - x_v)^2 + (y_R - y_v)^2}}, & x_R < x_v, y_R < y_v \end{cases}$$

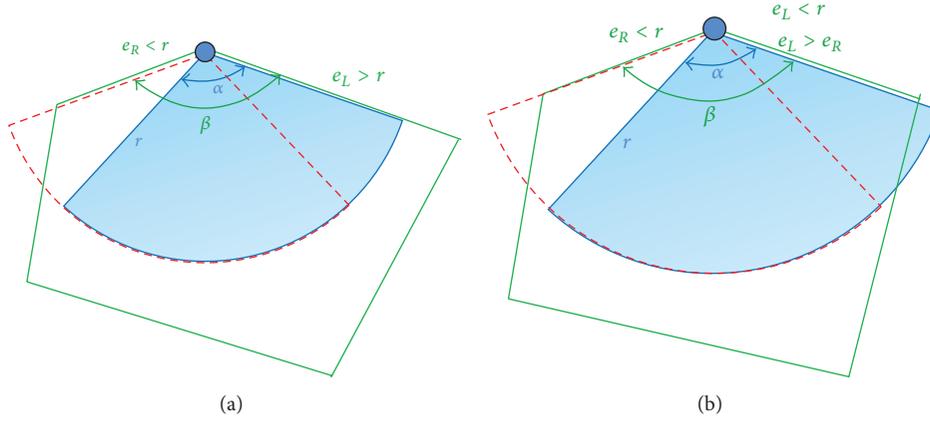


FIGURE 7: Two other cases of direction rotation.

$$\theta_L = \begin{cases} \sin^{-1} \frac{y_L - y_v}{\sqrt{(x_L - x_v)^2 + (y_L - y_v)^2}}, & x_L > x_v \\ \pi - \sin^{-1} \frac{y_L - y_v}{\sqrt{(x_L - x_v)^2 + (y_L - y_v)^2}}, & x_L < x_v, y_L > y_v \\ -\pi - \sin^{-1} \frac{y_L - y_v}{\sqrt{(x_L - x_v)^2 + (y_L - y_v)^2}}, & x_L < x_v, y_L < y_v. \end{cases} \quad (7)$$

If the working direction rotates right toward the  $\theta_R$  and the right sideline of the sector is lying on the cell edge  $e_R$ , then the angle value of working direction at the moment is  $\theta_1$ . Similarly,  $\theta_2$  is the angle value of working direction while the sensor rotates left toward the  $\theta_L$  and the left sideline of the sector is lying on the cell edge  $e_L$ . Because the AoV of the sensor is  $\alpha$ ,  $\theta_1$  and  $\theta_2$  can be calculated as (8). The values of  $\theta_1$  and  $\theta_2$  are between 0 and  $\pm\pi$ . Consider

$$\theta_1 = \begin{cases} \theta_R + \frac{\alpha}{2}, & \theta_R + \frac{\alpha}{2} \leq \pi \\ \theta_R + \frac{\alpha}{2} - 2\pi, & \text{otherwise} \end{cases} \quad (8)$$

$$\theta_2 = \begin{cases} \theta_L - \frac{\alpha}{2}, & \theta_L - \frac{\alpha}{2} \geq -\pi \\ \theta_L - \frac{\alpha}{2} + 2\pi, & \text{otherwise.} \end{cases}$$

Figure 7 shows other two different cases in which  $e_L$  is longer than  $e_R$ . However in Figure 7(a), sensing radius  $r$  is longer than  $e_R$  and shorter than  $e_L$ ; in Figure 7(b), sensing radius is longer than both  $e_R$  and  $e_L$ . In both of the two cases, making the left sideline of the sector lie on  $e_L$  and making the right sideline lie on  $e_R$  will obtain different *intracell* coverages. In the phase of working direction rotation, the proposed algorithm makes one of the sidelines of the sector lie on

the longer one of  $e_L$  and  $e_R$ ; for example, the  $e_L$  in Figure 7. This is mostly beneficial to reserve larger *intracell* coverage. Therefore, if  $\theta_0$  is the original working direction after the sensor was initially deployed and moved to the target vertex, the rotation range  $\varphi$  of the sensor is shown as (9). The sensor will make a clockwise or counterclockwise rotation with a range size of  $\varphi$ . Consider

$$\varphi = \begin{cases} \begin{cases} |\theta_0 - \theta_1|, & |\theta_0 - \theta_1| \leq \pi \\ 2\pi - |\theta_0 - \theta_1|, & |\theta_0 - \theta_1| > \pi \end{cases}, & e_R \geq e_L \\ \begin{cases} |\theta_0 - \theta_2|, & |\theta_0 - \theta_2| \leq \pi \\ 2\pi - |\theta_0 - \theta_2|, & |\theta_0 - \theta_2| > \pi \end{cases}, & e_L > e_R. \end{cases} \quad (9)$$

The detail of the algorithm for this phase is shown as Algorithm 2.

**4.4. Summary of the Procedures.** This subsection summarizes the procedures of the proposed algorithm. As shown in Figure 8, there are seven major procedures in the proposed distributed self-redeployment algorithm. The procedures are performed in each sensor with only few local information (the coordinates). Global information is not required. After the procedures are finished, overall coverage of sensing field is improved and then the sensors begin to do their sensing works.

Regarding the time complexity, the best known Fortune algorithm [27] can generate a Voronoi diagram from a given set of points in  $O(n \log n)$  time. However, in our proposed DVSA algorithm, each distributed sensor only constructs a local Voronoi cell by itself after the initial deployment according to the coordinates of its neighbor sensors. Therefore, the time complexity is  $O(n)$  for the initialization phase. In the decision phase, each sensor selects one of the cell vertices as the target position of movement. The time complexity is also  $O(n)$ . Finally, the sensor decides the working direction according to the two edges connected at the target vertex. The time complexity is  $O(1)$ . Therefore, the proposed DVSA algorithm has an overall time complexity of  $O(n)$ . This is better than the centralized algorithm of  $O(n^2 \log n + n^3)$  and the distributed algorithm of  $O(n \log n)$  proposed in [21].

```

 $v(x_v, y_v)$  is the coordinate of new sensor position
let  $v_R(x_R, y_R)$  is the right adjacent vertex of  $v$ 
let  $v_L(x_L, y_L)$  is the left adjacent vertex of  $v$ 
let  $\theta_0$  is the initial direction of the sensor
function get_edge_angle_value( $x, y$ )
{

$$\theta' = \begin{cases} \sin^{-1} \frac{y - y_v}{\sqrt{(x - x_v)^2 + (y - y_v)^2}}, & x > x_v \\ \pi - \sin^{-1} \frac{y - y_v}{\sqrt{(x - x_v)^2 + (y - y_v)^2}}, & x < x_v \text{ and } y > y_v \\ -\pi - \sin^{-1} \frac{y - y_v}{\sqrt{(x - x_v)^2 + (y - y_v)^2}}, & x < x_v \text{ and } y < y_v \end{cases}$$

return  $\theta'$ 
}
function get_rotation_range( $\theta_i$ )
{

$$\theta' = \begin{cases} |\theta_0 - \theta_i|, & |\theta_0 - \theta_i| \leq \pi \\ 2\pi - |\theta_0 - \theta_i|, & |\theta_0 - \theta_i| > \pi \end{cases}$$

return  $\theta'$ 
}
 $\theta_R = \text{get\_edge\_angle\_value}(x_R, y_R)$ 
 $\theta_L = \text{get\_edge\_angle\_value}(x_L, y_L)$ 
 $\theta_1 = \begin{cases} \theta_R + \alpha/2, & \theta_R + \alpha/2 \leq \pi \\ \theta_R + \alpha/2 - 2\pi, & \text{otherwise} \end{cases}$ 
 $\theta_2 = \begin{cases} \theta_L - \alpha/2, & \theta_L - \alpha/2 \geq -\pi \\ \theta_L - \alpha/2 + 2\pi, & \text{otherwise} \end{cases}$ 
 $e_R = \|\vec{vv}_R\|$ 
 $e_L = \|\vec{vv}_L\|$ 
if ( $e_R \geq e_L$ )
 $\varphi = \text{get\_rotation\_range}(\theta_1)$ 
rotate the working direction from  $\theta_0$  to  $\theta_1$  with rotation range  $\varphi$ 
else
 $\varphi = \text{get\_rotation\_range}(\theta_2)$ 
rotate the working direction from  $\theta_0$  to  $\theta_2$  with rotation range  $\varphi$ 
endif

```

ALGORITHM 2: Working direction rotation for directional sensor.

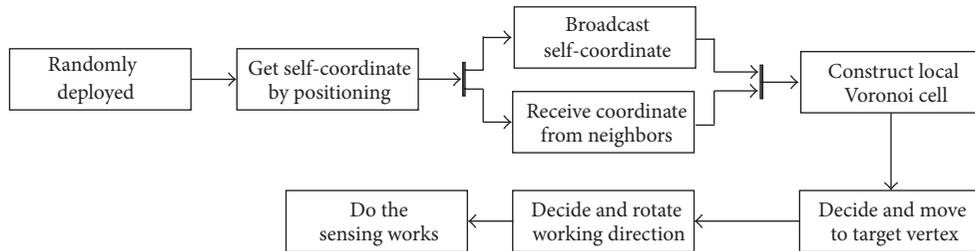


FIGURE 8: Procedures of the proposed self-redeployment algorithm.

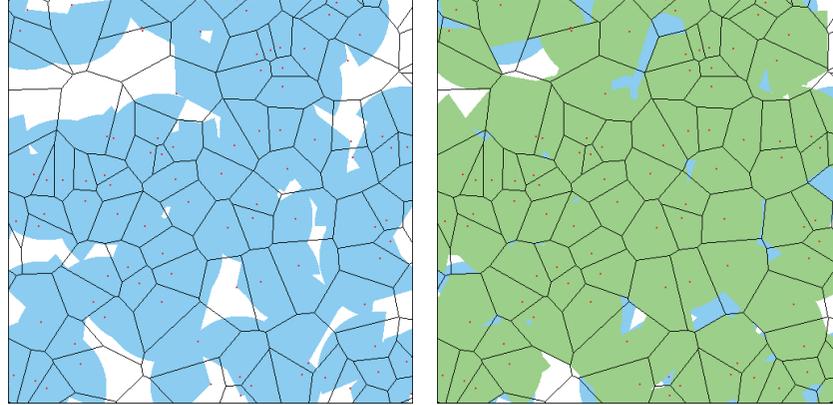
## 5. Performance Evaluation

This study evaluated the performance of the proposed distributed Voronoi-based self-redeployment algorithm (DVSA) by simulations. Field coverage ratio is the problem concerned in the algorithm and it is the key point of the

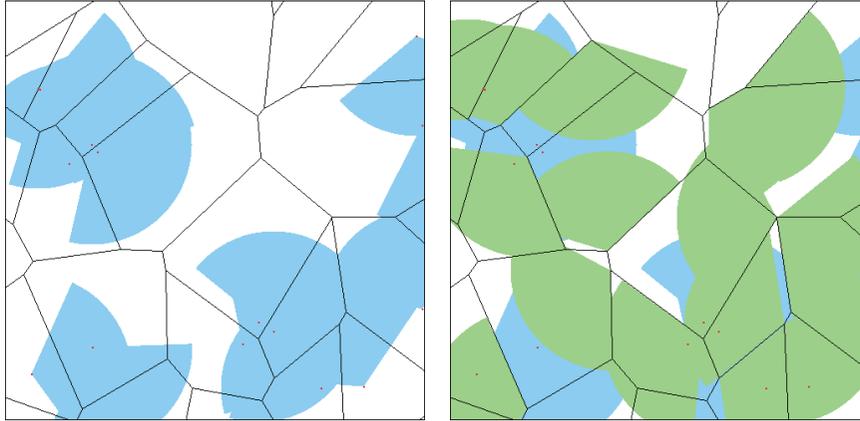
performance evaluation. In the simulations, mobile and direction-rotatable sensors are randomly deployed in a sensing field at the initial phase. The parameter of sensing field size is fixed at 500 m × 500 m. The other key parameters of the simulations comprise (1) number of sensors, (2) AoV of the sensor, and (3) sensing radius of the sensor.

TABLE 2: Notations and parameters for the directional sensing model.

Parameters	Notations	Default value	Variational values
Size of sensing field	$A$	500 m $\times$ 500 m	None
Number of sensors	$n$	150	50, 70, 90, 110, 130, 150
Sensing radius	$r$	60 m	10 m, 20 m, 30 m, 40 m, 50 m, 60 m
Angle of view	$\alpha$	180° ( $\pi$ )	30°, 60°, 90°, 120°, 150°, 180°



(a) Case 1: Initial versus DVSA



(b) Case 2: Initial versus DVSA

FIGURE 9: Pictures captured from the simulation program.

There are variations in values of the parameters in the simulations. As shown in Table 2, the number of sensors is 50~150 with an interval of 20, the sensing radius is 10 m~60 m with an interval of 10 m, and the angle of view (AoV) is 30°~180° with an interval of 30°. Each case in the simulations was repeated 100 times, and the average was taken as the result data. The simulation results are described in the following subsections. And Figure 9 illustrates a result with pictures captured from the simulation program.

Sections 5.1, 5.2, and 5.3 show the coverage performance of the proposed DVSA and the relationship among the number of sensors, sensing radius, and AoV. Then Section 5.4 shows the comparison with the other algorithms so as to prove that the proposed DVSA can improve the sensing field coverage well. Finally, Section 5.5 shows the performance result of sensors with different radii and AoVs; in other

words, each sensor has a random radius and a random AoV different from the ones of the other sensors.

#### 5.1. Coverage Ratio with Various Sensing Radii and AoVs.

Figure 10 shows the results of simulation by changing the sensing radius ( $r$ ) and the AoV ( $\alpha$ ), while fixing the number of sensors of  $n = 150$ . It illustrates the relationship among the coverage ratio, the sensing radius, and the AoV. For each AoV curve, the coverage ratio is increased with the increase of sensing radius. In addition, the larger the AoV is, the larger the slope is. In other words, this indicates that the larger the AoV is, the larger the increment of coverage ratio is. For instance, there is a coverage ratio increment of 43.72% from 1.52% (at  $r = 10$  m) to 45.24% (at  $r = 60$  m) on the curve of  $\alpha = 60^\circ$ , but an increment of 89.17% from 8.22% (at  $r = 10$  m) to 97.39% (at  $r = 60$  m) on the curve of  $\alpha = 180^\circ$ . On

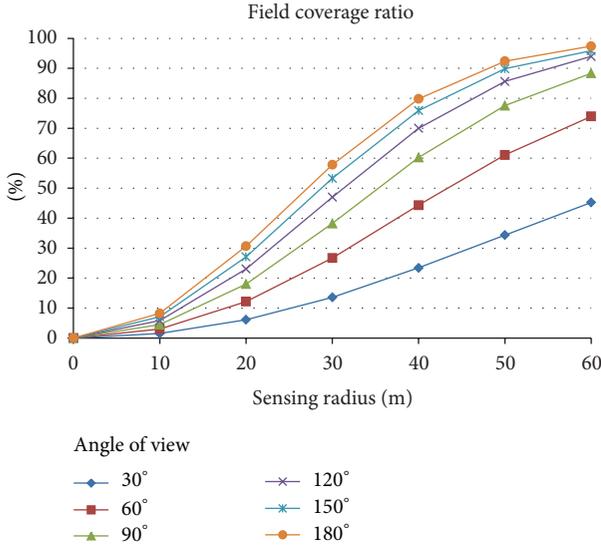


FIGURE 10: Coverage ratio (%) with fixed number of sensors ( $n = 150$ ).

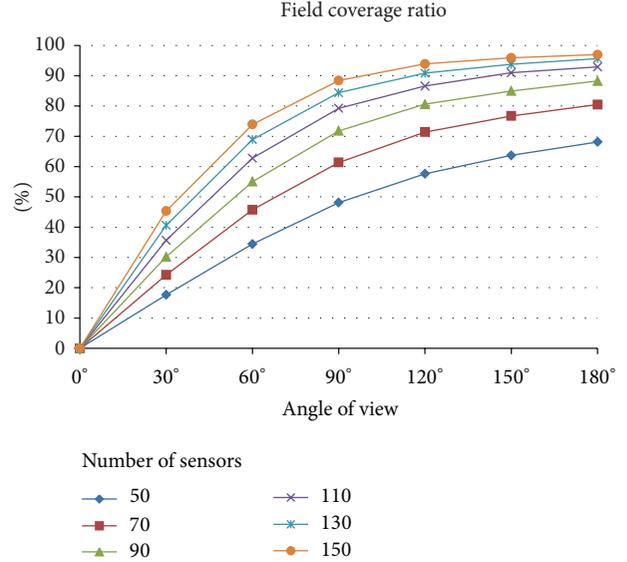


FIGURE 12: Coverage ratio (%) with fixed sensing radius ( $r = 60$  m).

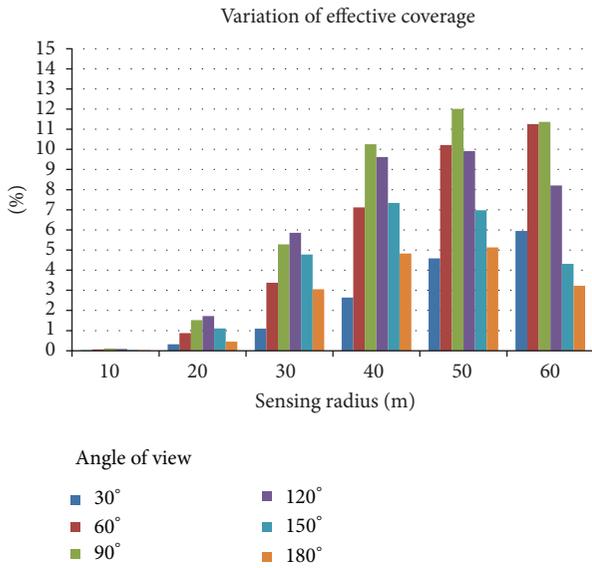


FIGURE 11: Variation of effective coverage (%) with fixed number of sensors ( $n = 150$ ).

the other hand, the figure shows that the larger the AoV is, the smaller the interval between two adjacent AoV curves is. This indicates that the increment of coverage contribution caused by the increment of AoV becomes smaller because the coverage overlap becomes larger.

Figure 11 shows the variation of effective coverage, which indicates the increments of coverage ratio relative to the initial ratio after deployment. It can be found that when  $r = 10$  m (one-fiftieth of the side length of the sensing field), there is a very small (almost zero) increment of coverage. This may mean the situation of almost no coverage overlap among sensors. In the figure, it also can be found that the peak value of increment occurs at  $r = 50$  m and  $\alpha = 90^\circ$ , which

means that these two parameter values can help obtain a best performance under the fixed number of sensors ( $n = 150$ ). Furthermore, for the  $\alpha = 30^\circ$  and  $60^\circ$ , the larger the sensing radius is, the larger the increment of coverage ratio is; but for  $\alpha = 120^\circ$ ,  $150^\circ$ , and  $180^\circ$ , the increments at  $r = 60$  m become smaller than the ones at  $r = 50$  m and at  $r = 40$  m. This means that the sensing radius larger than 60 m with the AoV larger than  $120^\circ$  will not keep raising the increment of coverage ratio. In other words, although the increment of coverage will become smaller under the cases, it will still be larger than the increment of overlap and the proposed algorithm will still obtain an improved coverage.

5.2. Coverage Ratio with Various AoVs and Numbers of Sensors.

Figure 12 shows the results of simulation by changing the AoV ( $\alpha$ ) and the number of sensors ( $n$ ), while fixing the sensing radius of  $r = 60$  m. It illustrates the relationship among the coverage ratio, the AoV, and the number of sensors. For each curve, the coverage ratio is increased with the increase of AoV. All of the curves have a larger slope at the beginning of the curve and a smaller slope at the end of the curve. This means that any curve does not keep the increment of coverage ratio equivalent every time the AoV is increased. For instance, there is a coverage ratio increment of 16.77% from 17.67% (at  $\alpha = 30^\circ$ ) to 34.44% (at  $\alpha = 60^\circ$ ) on the curve of  $n = 50$ , but only an increment of 4.42% from 63.71% (at  $\alpha = 150^\circ$ ) to 68.13% (at  $\alpha = 180^\circ$ ) on the same curve. The larger the AoV is, the smaller the coverage contribution is smaller. Also, the figure shows that the larger the number of sensors is, the smaller the interval between two adjacent curves is. This indicates that the increment of coverage contribution caused by the increment of number of sensors becomes smaller because the coverage overlap becomes larger.

Figure 13 shows the variation of effective coverage of fixed sensing radius ( $r = 60$  m) as AoV varies from  $30^\circ$  to  $180^\circ$

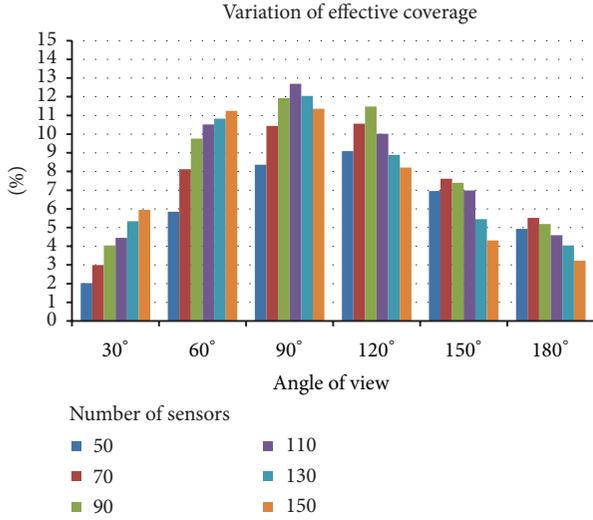


FIGURE 13: Variation of effective coverage (%) with fixed sensing radius ( $r = 60$  m).

and number of sensors varies from 50 to 150. For all of these cases, the proposed algorithm obtained positive increments of the coverage relative to the initial values significantly. The peak value of increment occurs at  $\alpha = 90^\circ$  and  $n = 110$ , which means that these two parameter values can help obtain a best performance under the fixed sensing radius ( $r = 60$  m). And in the figure, it can be found that the larger increments occur around  $\alpha = 60^\circ \sim 120^\circ$ . The AoV larger than  $120^\circ$  cannot bring more increment. This should be caused by the increase of coverage overlap. Furthermore, it also can be found that at  $\alpha = 30^\circ$  (smallest AoV), the bar of  $n = 150$  is the highest; on the contrary, at  $\alpha = 180^\circ$  (the largest AoV), the bar of  $n = 150$  is the lowest. Although the largest  $n$  with largest  $\alpha$  brings considerably large coverage overlap, it still have positive increment of coverage ratio due to the larger coverage contribution.

**5.3. Coverage Ratio with Various Numbers of Sensors and Sensing Radii.** Figure 14 shows the results of simulation by changing the number of sensors ( $n$ ) and the sensing radius ( $r$ ), while fixing the AoV of  $\alpha = 180^\circ$ . It illustrates the relationship among the coverage ratio, the number of sensors, and the sensing radius. The coverage ratio is low while the sensing radius is the shortest (at  $r = 10$  m); however, the increment of coverage ratio is significant if the sensing radius is increased. But the slopes of the curves seem smaller than the ones in the figures illustrated in previous subsections. The figure looks like that the increase of sensing radius brings more significant coverage ratio increment than the increase of number of sensors.

Figure 15 shows the variation of effective coverage of fixed AoV ( $\alpha = 180^\circ$ ) as number of sensors varies from 50 to 150 and sensing radius varies from 10 m to 60 m. It can be found that in certain cases even if the sensing radius is small and the number of sensors is not many, the increment of coverage relative to the initial value after deployment is a negative value

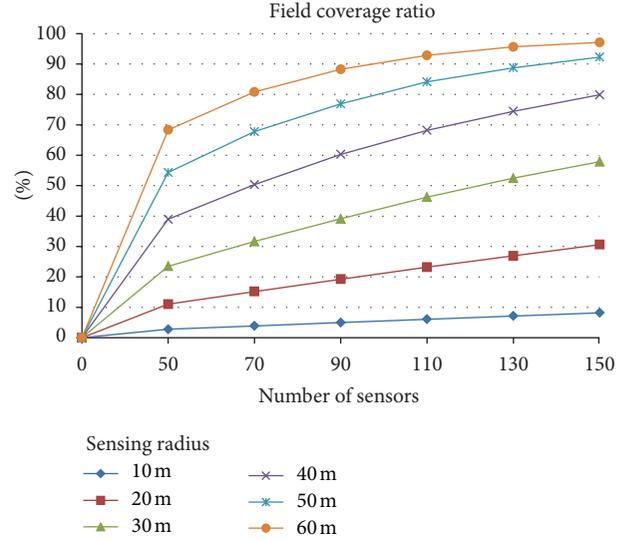


FIGURE 14: Coverage ratio (%) with fixed AoV ( $\alpha = 180^\circ$ ).

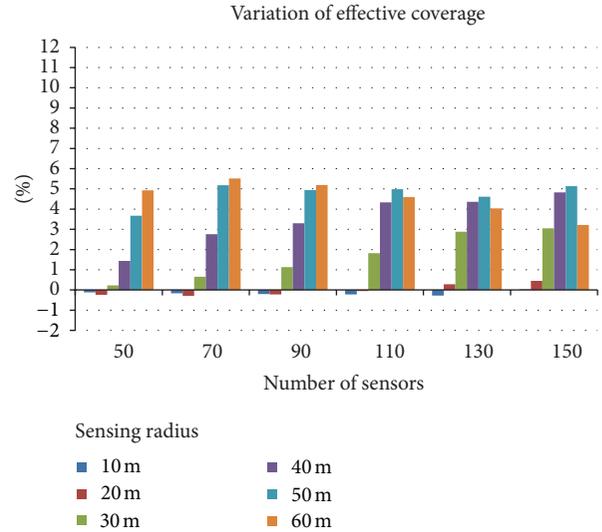


FIGURE 15: Variation of effective coverage (%) with fixed AoV ( $\alpha = 180^\circ$ ).

(around  $-0.1\% \sim -0.3\%$ ). This should be caused by the large AoV ( $\alpha = 180^\circ$ ), and the coverage overlap is a little larger than the coverage contribution in these cases. In addition, the difference of the increment between different numbers of sensors with same sensing radius is averagely not significant. This may mean that the increment of coverage is near the increase of overlap in these cases.

**5.4. Comparison with Different Algorithms.** The following show the results of comparing our proposed DVSA with the algorithms of VDA proposed in [21] and mentioned in Section 2. The RND (random) method has also been compared, which the RND means the initial value after the random deployment of sensors (with random position and random direction). Firstly, Figure 16 is the result by changing

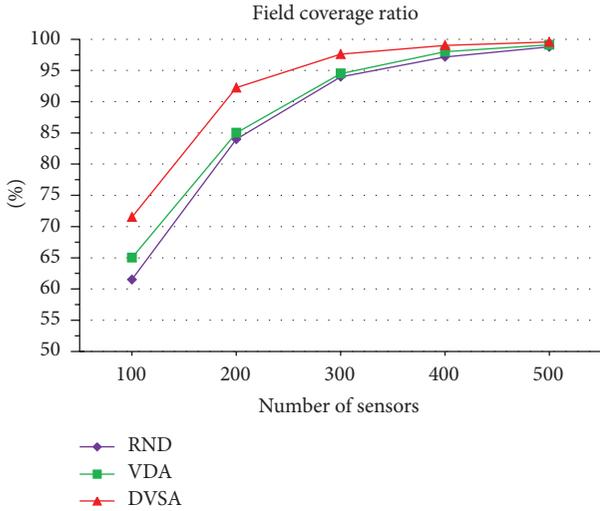


FIGURE 16: DVSA Compared with RND and VDA ( $n = 100 \sim 500$ ,  $r = 50$  m,  $\alpha = 120^\circ$ ).

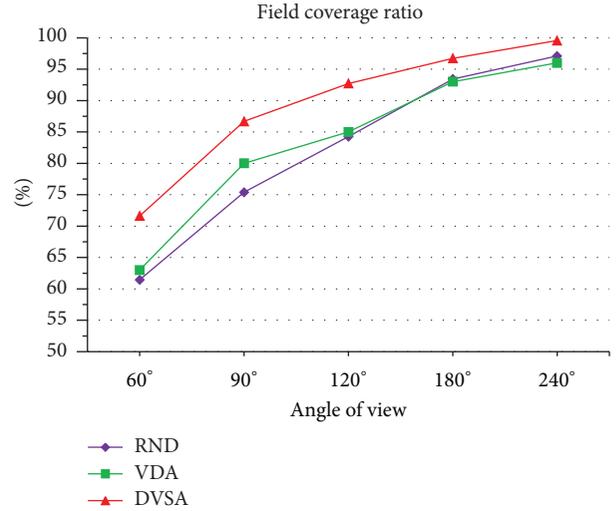


FIGURE 18: DVSA Compared with RND and VDA ( $n = 200$ ,  $r = 50$  m,  $\alpha = 60^\circ \sim 240^\circ$ ).

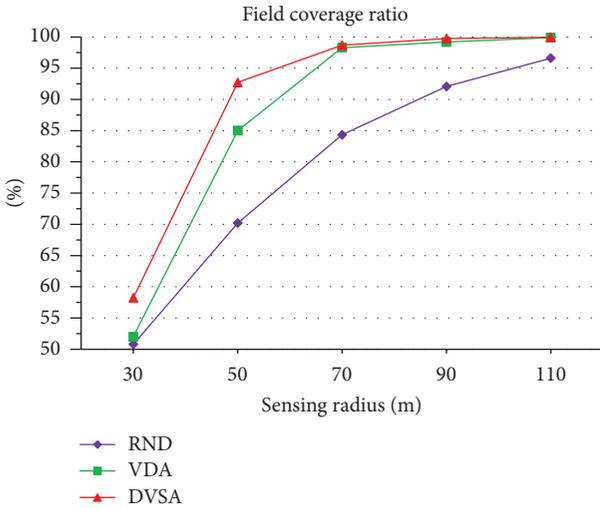


FIGURE 17: DVSA Compared with RND and VDA ( $n = 200$ ,  $r = 30$  m~110 m,  $\alpha = 120^\circ$ ).

the number of sensors from 100 to 500, and the sensing radius and AoV are fixed at 50 m and  $120^\circ$ , respectively. The DVSA curve shows a significant improvement of coverage performance, which is better than both VDA and RND. At the value of  $n = 100$ , the improvement is the largest, but the larger the  $n$  is, the lesser the improvement is. In particular at  $n = 500$ , DVSA, VDA and RND almost have the same performance; more sensors cannot bring more coverage contribution. Figure 17 shows the result of setting parameters as  $n = 200$ ,  $r = 30$  m~110 m, and  $\alpha = 120^\circ$ . The DVSA performs better than VDA at smaller number of sensors and performs almost equally at larger number of sensors. They converge on the point of  $r$  that is larger than 70 m while the performance RND is still significantly lower. Figure 18 shows the result of setting parameters as  $n = 200$ ,  $r = 50$  m, and  $\alpha = 60^\circ \sim 240^\circ$ . The DVSA still performs significantly better than

VDA. The VDA performs almost the same as RND except at the point of  $\alpha = 90^\circ$ . From these results, the proposed DVSA is proved that it performs very well under the situations of varied numbers of sensors, sensing radii, and angles of view.

Figure 19 shows the coverage ratio of the proposed DVSA algorithm compared with the result presented in the related study [19]. The simulation parameters: the size of sensing field is 500 m  $\times$  500 m; the number of sensors is 100; the sensing radius is 60 m; and the AoV is  $60^\circ$ , which are the same with the ones in that study. In the Figure 19, it shows that the coverage ratio of the sensing field can be well improved by the proposed DVSA algorithm while being compared with the related algorithms.

Figure 20 shows the result of the proposed DVSA algorithm compared with the GSP algorithm presented in the related study [20]. This simulation uses a large number of sensors of  $n = 1000$ . The sensing radius is fixed at  $r = 20$  m in Figure 20(a), and the AoV is fixed at  $\alpha = 120^\circ$  in Figure 20(b). The results also show that the DVSA obtained better coverage ratios than the GSP. The DVSA algorithm can be performed well.

In summary, the proposed DVSA method utilizes the vertices, edges, and included angles in each Voronoi cell to precisely compute the most suitable location and working direction for each sensor according to the algorithms described in Sections 4.2 and 4.3. On the contrary, the VDA approximately considers that if most Voronoi edges are covered, then most area will be covered; this is not definite and may cause more coverage overlap. In LRBA and MBAA algorithms, some nodes could not find their respective paired partner and cannot perform the algorithms for coverage contribution. And in the GSP algorithm, it needs the deployment of more sensors for repairing processes if certain grouped sensors are incommunicable to the sink, and the average coverage ratio of the grouped sensors will be decreased. Accordingly, the above performance results show

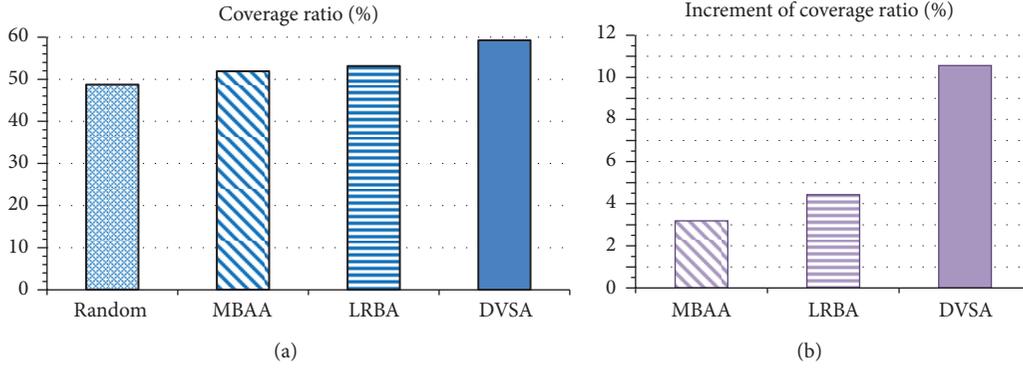


FIGURE 19: Simulation result compared with different algorithms ( $n = 100, r = 60 \text{ m}, \alpha = 60^\circ$ ).

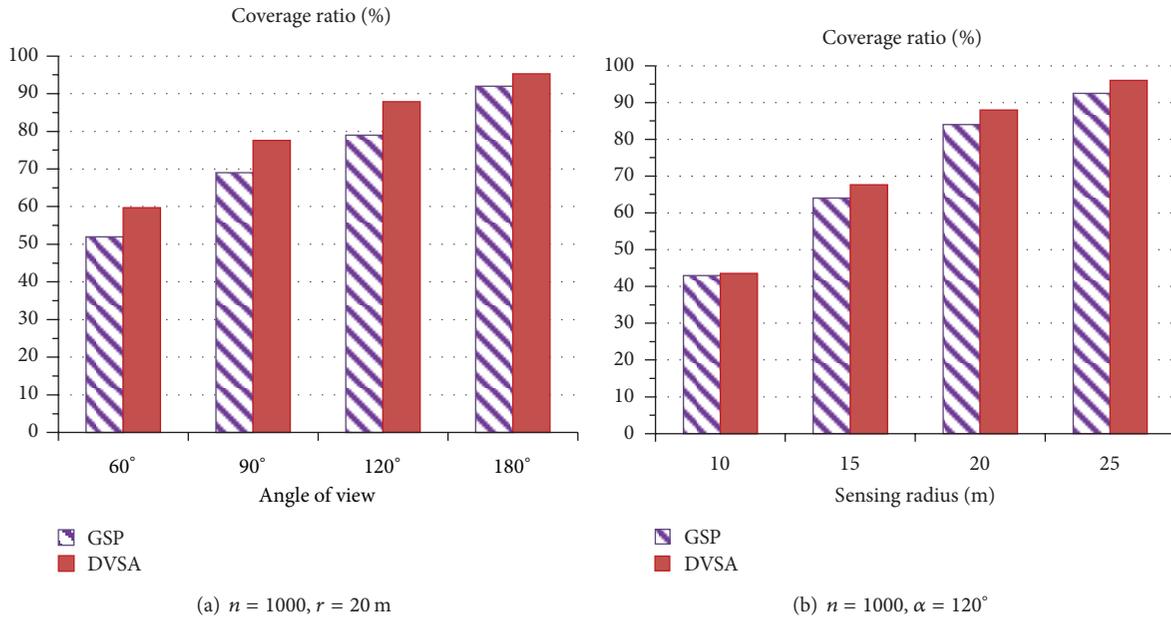


FIGURE 20: Simulation result compared with the GSP algorithm.

that our proposed DVSA performs better than the algorithms of VDA, LRBA, MBAA, and GSP.

**5.5. Performance of Sensors with Different Sensing Radii and AoVs.** In this subsection, the result of coverage performance of sensors with different radii and different AoVs is shown. In other words, the sensing radius and AoV of a sensor are randomly generated and different from the ones of the other sensors. Figure 21 illustrates this scenario with the pictures captured from the screenshot of the simulation program.

The parameter of the number of sensors varies from 30 to 200. The sensing radius of a sensor is randomly generated with a range between  $r_{\min} = 10 \text{ m}$  and  $r_{\max} = 100 \text{ m}$ , and the AoV is randomly generated with a range between  $\alpha_{\min} = 30^\circ$  and  $\alpha_{\max} = 150^\circ$ . Figure 22 shows the performance of DVSA compared with RND (random deployment). DVSA performs very well with an improved coverage ratio, no matter what the number of sensors is. And the increment of coverage ratio is kept around 5%. This result proves that our proposed DVSA

is applicable to sensors with different sensing radii and angles (we mentioned this in the beginning of Section 3.1).

## 6. Conclusion and Future Work

This study utilized the geometrical features of Voronoi diagram and the advantages of a distributed algorithm to propose the distributed Voronoi-based self-redeployment algorithm (DVSA), aiming to improve the overall field coverage of directional sensor networks effectively. The performance of the proposed algorithm and comparison with the different algorithm are also presented in the paper. The simulations prove that the DVSA method can improve the sensing coverage performance well.

Our future work will focus on combining the algorithm with an energy consumption model to give consideration to both coverage and lifetime performance in mobile and direction-rotatable directional sensor networks.

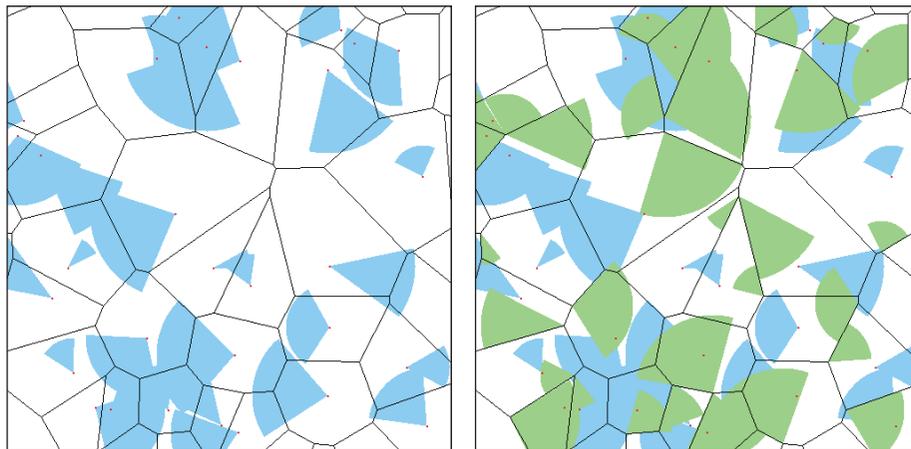


FIGURE 21: Pictures captured from the simulation program (initial versus DVSA).

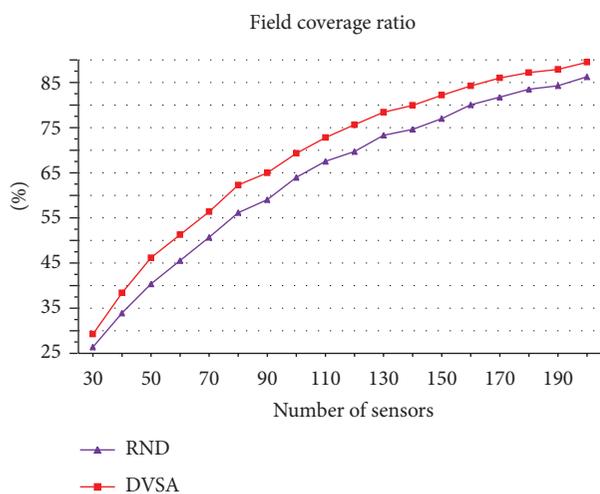


FIGURE 22: Performance ( $n = 30 \sim 200$ ;  $r_{\min} = 10$  m,  $r_{\max} = 100$  m;  $\alpha_{\min} = 30^\circ$ ,  $\alpha_{\max} = 150^\circ$ ).

## Acknowledgments

This work is supported by National Science Council of Taiwan under Grant no. NSC 102-2219-E-006-001 and the Research Center for Energy Technology of National Cheng Kung University under Grant no. DI102-23015.

## References

- [1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [2] C. Zhu, C. Zheng, L. Shu, and G. Han, "A survey on coverage and connectivity issues in wireless sensor networks," *Journal of Network and Computer Applications*, vol. 35, no. 2, pp. 619–632, 2012.
- [3] G. J. Fan and S. Y. Jin, "Coverage problem in wireless sensor network: a survey," *Journal of Networks*, vol. 5, no. 9, pp. 1033–1040, 2010.
- [4] Y. Charfi, N. Wakamiya, and M. Murata, "Challenging issues in visual sensor networks," *IEEE Wireless Communications*, vol. 16, no. 2, pp. 44–49, 2009.
- [5] S. Soro and W. Heinzelman, "A survey of visual sensor networks," *Advances in Multimedia*, vol. 2009, Article ID 640386, 22 pages, 2009.
- [6] J. Ai and A. A. Abouzeid, "Coverage by directional sensors in randomly deployed wireless sensor networks," *Journal of Combinatorial Optimization*, vol. 11, no. 1, pp. 21–41, 2006.
- [7] M. G. Guvensan and A. G. Yavuz, "On coverage issues in directional sensor networks: a survey," *Ad Hoc Networks*, vol. 9, no. 7, pp. 1238–1255, 2011.
- [8] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: a survey," *Ad Hoc Networks*, vol. 6, no. 4, pp. 621–655, 2008.
- [9] M. A. Guvensan and A. G. Yavuz, "A new coverage improvement algorithm based on motility capability of directional sensor nodes," in *Ad-Hoc, Mobile, and Wireless Networks*, vol. 6811 of *Lecture Notes in Computer Science*, pp. 206–219, Springer, Berlin, Germany, 2011.
- [10] H. Ma and Y. Liu, "On coverage problems of directional sensor networks," in *Mobile Ad-Hoc and Sensor Networks*, vol. 3794 of *Lecture Notes in Computer Science*, pp. 721–731, Springer, Berlin, Germany, 2005.
- [11] C. Pham, A. Makhoul, and R. Saadi, "Risk-based adaptive scheduling in randomly deployed video sensor networks for critical surveillance applications," *Journal of Network and Computer Applications*, vol. 34, no. 2, pp. 783–795, 2011.
- [12] K.-Y. Chow, K.-S. Lui, and E. Y. Lam, "Wireless sensor networks scheduling for full angle coverage," *Multidimensional Systems and Signal Processing*, vol. 20, no. 2, pp. 101–119, 2009.
- [13] J. Wang, C. Niu, and R. Shen, "Priority-based target coverage in directional sensor networks using a genetic algorithm," *Computers and Mathematics with Applications*, vol. 57, no. 11–12, pp. 1915–1922, 2009.
- [14] Y.-C. Hsu, Y.-T. Chen, and C.-K. Liang, "Distributed coverage-enhancing algorithms in directional sensor networks with rotatable sensors," in *Distributed Computing and Networking*, vol. 7129 of *Lecture Notes in Computer Science*, pp. 201–213, Springer, Berlin, Germany, 2012.
- [15] Z. Jing and Z. Jian-Chao, "A virtual centripetal force-based coverage-enhancing algorithm for wireless multimedia sensor

- networks,” *IEEE Sensors Journal*, vol. 10, no. 8, pp. 1328–1334, 2010.
- [16] C.-K. Liang, M.-C. He, and C.-H. Tsai, “Movement assisted sensor deployment in directional sensor networks,” in *Proceedings of the 6th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN '10)*, pp. 226–230, Hangzhou, China, December 2010.
- [17] M. A. Guvensan and A. G. Yavuz, “A hybrid solution for coverage enhancement in directional sensor networks,” in *Proceedings of the 7th International Conference on Wireless and Mobile Communications*, pp. 134–138, IARIA, Luxembourg City, Luxembourg, 2011.
- [18] N. Tezcan and W. Wang, “Self-orienting wireless multimedia sensor networks for occlusion-free viewpoints,” *Computer Networks*, vol. 52, no. 13, pp. 2558–2567, 2008.
- [19] H. Huang, L. Sun, R. Wang, and J. Li, “A novel coverage enhancement algorithm for image sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 370935, 11 pages, 2012.
- [20] H. Ma and Y. Liu, “Some problems of directional sensor networks,” *International Journal of Sensor Networks*, vol. 2, no. 1-2, pp. 44–52, 2007.
- [21] J. Li, R. Wang, H. Huanh, and L. Sun, “Voronoi-based coverage optimization for directional sensor networks,” *Wireless Sensor Network*, vol. 1, no. 5, pp. 417–424, 2009.
- [22] W. Cheng, S. Li, X. Liao, S. Changxiang, and H. Chen, “Maximal coverage scheduling in randomly deployed directional sensor networks,” in *Proceedings of International Conference on Parallel Processing Workshops (ICPPW '07)*, Xian, China, September 2007.
- [23] Z. Li, R. Li, Y. Wei, and T. Pei, “Survey of localization techniques in wireless sensor networks,” *Information Technology Journal*, vol. 9, no. 8, pp. 1754–1757, 2010.
- [24] G. T. Sibley, M. H. Rahimi, and G. S. Sukhatme, “Robomote: a tiny mobile robot platform for large-scale ad-hoc sensor networks,” in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1143–1148, Washington, DC, USA, May 2002.
- [25] G. Wang, M. J. Irwin, P. Berman, H. Fu, and T. La Porta, “Optimizing sensor movement planning for energy efficiency,” in *Proceedings of the International Symposium on Low Power Electronics and Design*, pp. 215–220, San Diego, Calif, USA, August 2005.
- [26] F. Aurenhammer, “Voronoi diagrams—a survey of a fundamental geometric data structure,” *ACM Computing Surveys*, vol. 23, no. 3, pp. 345–405, 1991.
- [27] S. Fortune, “A sweepline algorithm for Voronoi diagrams,” *Algorithmica*, vol. 2, no. 1-4, pp. 153–174, 1987.
- [28] A. Okabe, B. Boots, and K. Sugihara, *Spatial Tessellations, Concepts and Applications of Voronoi Diagrams*, John Wiley & Sons, 2nd edition, 2000.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

