

Chiu-Kuo Liang<sup>1\*</sup>, Cheng-Yen Chung<sup>1</sup>, and Chuan-Feng Li<sup>1</sup>

<sup>1</sup> Department of Computer Science and Information Engineering, Chung Hua University Hsinchu 300, Taiwan, ROC {ckliang, m10002012, m9902018}@chu.edu.tw

Received 7 July 2015; Revised 27 December 2016; Accepted 28 December 2016

Abstract. A directional sensor network is composed of many directional sensor nodes. Unlike conventional omni-directional sensors that always have an omni-angle of sensing range, directional sensors may have a limited angle of sensing range due to technical constraints or cost considerations. Area coverage is still an essential issue in a directional sensor network. In this paper, we study the area coverage problem in directional sensor networks with mobile sensors, which can move to the coverage holes to get better coverage ratio. In order to increase the coverage ratio, each sensor will move from overlapped regions to coverage holes after random deployment. The movements of sensors are adjusted round-by-round so that the coverage ratio is gradually improved. To guide the moving direction of each sensor, we introduce a virtual forces based coverage algorithm for directional sensor networks. Our proposed scheme consists of four different forces caused by neighboring sensors and uncovered regions in the field. By applying the resultant force from these four different forces to directional sensors, coverage holes are eliminated quickly. Simulation results show the effectiveness of our scheme in term of the coverage improvement.

Keywords: area coverage, directional sensors, mobile sensors

# 1 Introduction

In recent years, wireless sensor networks have received a lot of attention due to their wide applications in military and civilian operations, such as fire detection, vehicle traffic monitoring, and battlefield surveillance [1-2]. Most of the past work is always based on the assumption of omni-directional sensors that has an omni-angle of sensing range. However, there are many kinds of directional sensors, such as video sensors, ultrasonic sensors and infrared sensors [3]. The omni-directional sensor node has a circular disk of sensing range. The directional sensor node has smaller sensing area (sector-like area) and sensing angle than the omni-directional one. Compared to isotropic sensors, the coverage region of a directional sensor is determined by its location and orientation. This can be illustrated by the example as shown in Fig. 1.

In wireless sensor networks, area coverage is a fundamental problem. Therefore, sensor nodes must be deployed appropriately to reach an adequate coverage level for the successful completion of the issued sensing tasks [4]. However, in many potential working environments, such as remote harsh fields, disaster areas, and toxic urban regions, sensor deployment cannot be performed manually. To scatter sensors by aircraft may result in the situation that the actual landing positions cannot be controlled. Consequently, the coverage may be inferior to the application requirements no matter how many sensors are dropped. In such case, it is necessary to make use of mobile sensors, which can move to the correct positions to provide the required coverage.

<sup>\*</sup> Corresponding Author



Fig. 1. An example of five directional sensors deployed to cover the target region

Most previous research efforts on deploying mobile sensors are based on the omni-directional sensor networks. For example, Howard, Mataril and Sukhatme [5] present a distributed, potential-field-based approach to solve the coverage problem. In their approach, sensor nodes are treated as virtual particles that are subject to force, these forces repel the neighboring sensor nodes from each other and from obstacles. Finally, sensor nodes will spread from dense to sparse area. The concept of potential-field was first proposed in the research of mobile robotic route plan and obstacle avoidance by Khatib [6]. Wang, Gao and Porta present a set of Voronoi diagram-based schemes to maximize sensing coverage [7]. After discovering a coverage hole locally, the schemes calculate new position for each sensor to move at next round. They use the Voronoi diagram to discover the coverage holes and design three movement-assisted sensor deployment schemes: VEC (VECtor-based), VOR (VORonoi-based), and Minimax. Lee, Kim, Han and Park design two movement-assisted schemes: Centroid-based and Dual-Centroid-based [8]. Based on the Voronoi diagram and centroid (geometric center), the proposed schemes can be used to improve the sensing coverage. Liang, He and Tsai proposed the movement schemes in directional sensor networks [10]. They described each directional sensor as a circumscribed circle and an inscribed circle. Then, by applying the movement schemes proposed by [7] to the circumscribed and inscribed circles, their proposed movement schemes can achieve better coverage performance.

In this paper, we study the problem of area coverage by directional mobile sensor under the random deployment strategy. We develop a solution that maximizes the sensing coverage while minimizing the computation time in term of rounds. Based on the virtual forces of directional sensors, we design a moving algorithm, called the Virtual Force based Moving (VFM) algorithm, to guide each sensor's movement. Simulation results show that our distributed algorithm is effective in terms of coverage rate improvement.

The rest of this paper is organized as follows. In Section 2, we introduce some preliminaries. In Section 3, we state the problem formally and make some assumptions regarding the problem. We present our scheme in Section 4. Section 5 shows some simulation results. Finally, we conclude this paper in Section 6.

### 2 Preliminaries

Unlike isotropic sensors, directional sensors have their own sensing model. In this section, we describe the directional sensing model and some preliminaries that will be used throughout this paper.

#### 2.1 Directional Sensing Model

Compared to an omni-directional sensor, which has a disc-shaped sensing range, a directional sensor has a smaller sector-like sensing area and smaller sensing angle, as illustrated in Fig. 2. As shown in Fig. 2, the sensing region (also called the sensing sector) of a directional sensor is a sector denoted by 4-tuple (S,  $\vec{V}$ ,  $\alpha$ , R), where S is the location of the sensor,  $\vec{V}$  is the center line of sight of the field of view,  $\alpha$  is the sensing angle, and R is the sensing radius; further, in the figure,  $\beta$  is defined as its direction angle value relative to the horizontal.



Fig. 2. The directional sensing model

#### 2.2 Auxiliary Point and Centroid Circle

In order to describe the sensing sector of a directional sensor, we assume that there are three auxiliary points around the boundary. The auxiliary points are the endpoints of the boundary of the sensing sector. For each sensor  $s_i$ , we use  $A_{ik}$ , where  $1 \le k \le 3$ , to denote its corresponding auxiliary points as shown in Fig. 3. The centroid point of sensor  $s_i$ , denoted as  $C_i$ , is the centroid (geometric center) of the sector area of  $s_i$ .



Fig. 3. The virtual force points of a sensor

Furthermore, we define the centroid circle to be the circle with the center  $C_i$  and radius  $r_{cen}$  which is the distance from  $C_i$  to  $S_i$ , as shown in Fig. 4. The reason of using centroid circle instead of the circumscribed and inscribed circles is that we observe that the centroid circle is more accurate to approach the sector area of a directional sensor than both circumscribed and inscribed circles, as shown in Fig. 5. In other words, we use the centroid point as the center and  $r_{cen}$  as the radius to describe the sector area of a directional sensor.



Fig. 4. The centroid circle of a directional sensor



Fig. 5. Comparison of centroid, circumscribed, and inscribed circles

#### 2.3 Voronoi Polygon and Farthest Voronoi Point

Given a set of points  $S = \{s_1, s_2, \dots, s_n\}$ , the Voronoi diagram, a well-known data structure in computational geometry [9], partitions a plane into a set of convex polygons such that all points inside a polygon are closest to only one site. The construction effectively produces polygons with edges that are equidistant from neighboring sites.

We define the Voronoi polygon of  $s_i$  as  $G_i = (V_i, E_i)$ , where  $V_i$  is the set of Voronoi vertices of  $s_i$ , and  $E_i$  is the set of Voronoi edges. According to the property of Voronoi diagram, if an arbitrary point p in  $G_i$ , then  $d(p, s_i) < d(p, s_j)$ , where  $i, j = 1, 2, ..., n, i \neq j$ , and d(x, y) denotes the Euclidean distance between points x and y. Fig. 6 shows an example of Voronoi polygons.



**Fig. 6.** The Voronoi polygon  $G_0$ 

In this paper, we treat the centroid points of directional sensors as a set of points in the plane. Then, we observe that sensor  $s_i$  will cover the area of its corresponding Voronoi polygon  $G_i$ , as shown in Fig. 7. However, there exist some holes in the Voronoi polygon which cannot be covered by its corresponding sensor. In such case, we would like to move the sensor to the direction of coverage holes. To do so, for each Voronoi polygon  $G_i$ , we define the Farthest Voronoi Point, denoted as  $FVP_i$ , to be the Voronoi vertices in the polygon with the farthest distance to the corresponding centroid point  $C_i$  of sensor  $s_i$ . In other words, sensor  $s_i$  will move its direction to cover  $FVP_i$  such that the coverage hole inside  $G_i$  will be eliminated.



Fig. 7. The Farthest Voronoi Point

## 3 Problem Statement

Problem: Randomly deploying N mobile directional sensors with sensing range  $R_S$ , communication range  $R_C$  and sensing angle  $\alpha$  in a given target sensing region, we are asked to maximize the sensor coverage while minimizing the required total moving distances.

To address the above problem, we need to make the following assumptions in which some of them are similar to those in [7]:

- All directional sensors have the same sensing range  $(R_s)$ , communication range  $(R_c)$  and sensing angle  $\alpha$ , where  $0 < \alpha \le \pi$ . Directional sensors within  $R_c$  of a sensor are called the sensor's neighboring nodes. The communication range is equal to twice the sensing radius.
- · Directional sensors can move to arbitrary positions via GPS, but their sensing directions are not

rotatable.

- Each directional sensor knows its location information, which includes sensing region, sensing radius, sensing angle, and field of view. Each sensor can also obtain the location information of its neighboring sensors.
- We do not consider the energy consumption of the directional sensors, but we view the distance moved as proportional to energy consumption—i.e., the more distance moved, the more energy consumed.

# 4 Proposed Moving Schemes

To maximize the area coverage, we present a virtual force based movement algorithm (VFM) for directional mobile sensors. The basic ides of our virtual force based moving scheme consists of two different forces, namely the *push force* and the *pull force*. The push force is motivated by the attributes of electromagnetic particles: When two electromagnetic particles are too close to each other, an expelling force pushes them apart. Contrary to the push force, the pull force will pull sensors to cover their local coverage holes. Our proposed movement scheme is based on the multiple virtual forces occurred on each sensor. We observe that, for each sensor, there exist many forces incurred by other sensors since it is possible to have different relationships with other sensors. For example, it may have one or two auxiliary points overlapped with other sensors or no overlap at all. Even in the case with no overlapping, the distance between two directional sensor into four different categories of forces for a sensor to consider. The four different categories of forces consist of the Centroid Push Auxiliary point Force (CPAF), the Voronoi point Push Centroid Force (VPCF), and the Neighbor Repulsive Force (NRF). The final overall force on sensors is the vector summation of the above four different virtual forces and will be used as the direction for sensors to move.

## 4.1 Centroid Push Auxiliary point Force (CPAF)

The Centroid Push Auxiliary point Force (CPAF) is a push force that occurred when the auxiliary points of a sensor are covered by another sensor. As shown in Fig. 8, the auxiliary point  $A_{i3}$  of sensor  $s_i$  is located inside the area of sensor  $s_j$ . Thus, based on the idea of virtual force, we observe that the centroid point of  $s_j$  will have an expelling force for pushing the auxiliary point  $A_{i3}$  away from its sensing area. For simplicity, we denote the force as a vector. It should be noticed that all forces occurred in a sensor are finally acting on the sensor node.



Fig. 8. Illustration of the Centroid Push Auxiliary point Force

We use  $\overrightarrow{CPAF_{ij}}$  to denote the push force exerted by  $s_j$  on  $s_i$  that  $s_j$  will use this force to push  $s_i$  away from  $s_j$ . The push force can be calculated by following equation:

$$\overrightarrow{CPAF_{ij}} = \sum_{k=1}^{3} |r_{cen} - \overrightarrow{C_j A_{ik}}|| \times \frac{\overrightarrow{C_j A_{ik}}}{|\overrightarrow{C_j A_{ik}}|}$$
(1)

#### 4.2 Centroid Push Centroid Force (CPCF)

The Centroid Push Centroid Force (CPCF) is occurred when the centroid points of different sensors are close enough to each other. As shown in Fig. 9, the distance between centroid points of  $s_i$  and  $s_j$  is less than their sensing range. Therefore, sensors  $s_i$  and  $s_j$  may have some sensing area which is overlapped with each other. If they are overlapped then, according to the virtual force idea, sensor  $s_i$  will incurred a force to push away sensor  $s_j$  from  $s_i$  and vice versa. It should be noticed that the strength of this push force is proportional to the size of the overlapped area between  $s_i$  and  $s_j$ .



Fig. 9. Illustration of the Centroid Push Centroid Force

We use  $\overline{CPCF_{ij}}$  to denote the push force exerted by  $s_j$  on  $s_i$  that  $s_j$  will use this force to push  $s_i$  away from  $s_j$ . The push force can be calculated by following equation:

$$\overline{CPCF_{ij}} = \frac{2r_{cen} - |\overline{C_jC_i}|}{2} \times \frac{OA_{ij}}{SA} \times \frac{\overline{C_jC_i}}{|\overline{C_iC_i}|}$$
(2)

where  $OA_{ij}$  denotes the size of the overlapped area between  $s_i$  and  $s_j$ , and SA denotes the size of the entire sensing area.

#### 4.3 Voronoi point Pull Centroid Force (VPCF)

The Voronoi point Pull Centroid Force (VPCF) is occurred when the Voronoi polygon  $G_i$  of sensor  $s_i$  cannot be fully covered by  $s_i$ , as shown in Fig. 10. In other words, there exists some coverage holes inside  $G_i$ . Therefore, we introduce an attraction force occurred at the farthest Voronoi point of  $G_i$  to pull sensor  $s_i$  to get closer to it. By doing so, the coverage hole can be eliminated. We use  $\overline{VPCF_{ij}}$  to denote the pull force generated by  $FVP_i$  on  $s_i$  that  $s_i$  will use this force to move to cover  $FVP_i$ . The pull force can be calculated by following equation:

$$\overrightarrow{VPCF_{ij}} = |r_{cen} - |\overrightarrow{C_jFVP_i}|| \times \frac{\overrightarrow{C_jFVP_i}}{|\overrightarrow{C_jFVP_i}|}$$
(3)



Fig. 10. Illustration of the Voronoi point Pull Cenrtoid Force

#### 4.4 Neighbor Repulsive Force (NRF)

In order to have better coverage, sensors need to move from high density region to sparse region. Therefore, we introduce a repulsive force that occurred by the neighboring sensors of each sensor, as shown in Fig. 11. The action of this force is to push a sensor away from its neighboring sensors. Thus, in a high density region, a sensor which is located around the boundary of this region will be pushed forward to a sparse region. By doing so, the coverage rate can be increased. We use  $NRF_{ij}$  to denote such force and it can be calculated as in the following equation:

$$\overline{VPCF_{ij}} = |r_{cen} - |\overline{C_j FVP_i}|| \times \frac{\overline{C_j FVP_i}}{|\overline{C_j FVP_i}|}$$
(4)



Fig. 11. Illustration of the neighbor repulsive force

#### 4.5 The Movement Scheme

In this subsection, we propose a movement scheme that based on the virtual force idea to improve the coverage in directional sensor networks. Our proposed movement scheme consists of three phases, namely the Discovery phase, the Move-forward phase and the Move-back phase. The purpose of discovery phase is to collect all the necessary information from neighboring sensors of each sensor. Then, each sensor will enter into the Move-forward phase which is to move each sensor to the new location based on final resultant force it has received. The Move-forward phase is executed on a round-by-round basis. In order to save energy, we set a upper bound of the number of rounds, namely *Max<sub>round</sub>*. The Move-forward phase will be stopped after executing *Max<sub>round</sub>* rounds. Then, each sensor will enter into the Move-back phase if its sensing area is out of the target area.

The purpose of Move-back phase is to prevent the sensor's sensing region from being outside the target region. If the auxiliary point of a sensor is out of the target region, the sensor should move to the new position until its auxiliary point is located on the boundary of the target region, as shown in Fig. 12. The Move-back procedure can easily be obtained by doing horizontal and vertical movement. If the auxiliary point is above or below the horizontal boundary of the target region, the sensor will move back vertically. Similarly, if the auxiliary point is beyond the vertical boundary of the target region, the sensor will move back horizontally.



Fig. 12. Illustration of the Move-back phase

Fig. 13 shows an example of our Move-back procedure. In Fig. 13(a), the auxiliary points are located out of the target region. Since there are one auxiliary point above the upper boundary and one auxiliary point out of the left boundary of the target region respectively, the sensor will move back horizontally and vertically into the target region, as shown in Fig. 13(b). The Move-back procedure will be executed until its auxiliary points are located on the boundary of the target region.



Fig. 13. Illustration of Move-back procedure

The pseudo-code of our proposed Virtual Force based Movement (VFM) algorithm is shown in Fig. 14.

Virtual Force based Movement (VFM) algorithm
Notations :
$\overrightarrow{CPAF_{ij}}, \overrightarrow{CPCF_{ij}}, \overrightarrow{VPCF_{ij}}, \overrightarrow{NRF_{ij}}, OA_{ij}$ : defined above
$N_i$ : the neighbors of sensor $s_i$
$\vec{V}_i$ : moving vector of $s_i$
<i>Max<sub>round</sub></i> : pre-defined maximum number of round
(1) Upon entering <i>Discovery phase</i> :
(1.1) set <i>timer</i> to be <i>discovery_interval</i>
enter Moving phase upon timeout
(1.2) broadcast <i>hello</i> after a random time slot
(2) Upon entering <i>Moving phase:</i>
(2.1) set <i>timer</i> to be <i>moving_interval</i> ,
enter Discovery phase upon timeout
(2.2) Call Move()
(2.3) entering Move-back phase
$(2.4) Max_{round} = Max_{round} - 1$
(2.5) Done when $Max_{round} = 0$
(3) Upon receiving a <i>hello</i> message from sensor $s_j$ :
(3.1) Update $N_i$ , $G_i$ , and $OA_{ij}$ for each $s_j$ in $N_i$
(4) Move()
$(4.1) \ \overrightarrow{V_i} = 0$
(4.2) for each $s_j$ in $N_i$
Compute $\overrightarrow{CPAF_{ij}}$ , $\overrightarrow{CPCF_{ij}}$ , $\overrightarrow{VPCF_{ij}}$ , and $\overrightarrow{NRF_{ij}}$

 $\vec{V}_i = \vec{CPAF}_{ij} + \vec{CPCF}_{ij} + \vec{VPCF}_{ij} + \vec{NRF}_{ij}$ (4.3) do movement adjustment (5) Upon entering Move-back phase: (5.1) do move-back scheme



## 5 Simulation Results

In this section, we evaluate the performance of our proposed Virtual Force based Movement algorithm, denoted as VFM, and the random deployment approach, denoted as Random, in which all sensors are randomly deployed initially. Moreover, we compare the simulation results with the theoretically optimal solution, denoted as Optimal. The theoretically optimal solution is obtained by computing the total area that can be covered by all sensors in which no two sensors are overlapped. We evaluate our proposed scheme from two aspects: coverage and moving distance. Each simulation is executed 20 times with results averaged. Experimental environment is shown in Table 1.

#### Table 1. Experimental parameters

Parameters	Values
Network size	$500 \times 500 \ m^2$
Sensing radius $(R_S)$	60 <i>m</i>
Communication radius $(R_C)$	120 <i>m</i>
Sensing angle ( $\alpha$ )	90°
No. of directional sensors	90
No. of rounds	12

Fig. 15 shows the experimental result for evaluating the effect that the number of sensors N makes to the performance of coverage ratio of Random approach and the VFM algorithm, respectively. In these graphs, we can see that our proposed VFM algorithm can achieve better coverage ratio as the number of sensors increased. For example, when the number of sensors is 90, the coverage ratio of Random approach and VFM algorithm are 59% and 79%, respectively. Thus, our proposed VFM algorithm performs 20% better than Random approach.

Fig. 16 shows the experimental result for evaluating the effect that the number of sensors N makes to the performance of moving distance of our VFM algorithm. In these graphs, we can see that the moving distance of our proposed VFM algorithm is proportionally increased to the number of sensors. This is reasonable as when the number of sensors increased, each sensor will move to the coverage hole so that the total moving distance will be increased as well.



Fig. 15. Coverage ratio vs. Number of sensors



Fig. 16. Moving distance vs. number of sensors

Fig. 17 and Fig. 18 show the results for evaluating the effect that the number of rounds makes to the performance of coverage ratio of our VFM algorithm. In this experiment, we evaluate the effects of two different numbers of sensors, 60 and 90 sensors. In these graphs, we can see that the coverage ratio can be greatly improved in the first few rounds by our proposed VFM algorithm. The coverage ratio can be improved by 10% by our algorithm after first round of execution. The initial coverage ratios of random deployment are around 45% and 60% of 60 and 90 sensors, respectively. The coverage ratios are both stable after execution of 5 rounds, which result in about 15% and 20% improvement respectively. Thus, our proposed VFM algorithm can improve the coverage ratio effectively.



Fig. 17. Coverage ratio vs. number of rounds – 60 sensors



Fig. 18. Coverage ratio vs. number of rounds – 90 sensors

Fig. 19 shows the result for evaluating the effect that the number of sensors N makes to the performance of coverage incremental ratio of our VFM algorithm. In these graphs, we can see that the as the number of sensors increased, the coverage incremental ratio is also raised. However, as the number of sensors exceeds 50, the coverage incremental ratio is stable around 25%. This is because that as the number of sensors approaches 50, the target area is fulfilled with sensors so that the coverage ratio cannot be improved any further.



Fig. 19. Coverage incremental ratio vs. number of sensors

## 6 Concluding Remarks

In this paper, we investigated the area coverage problem in directional sensor networks with mobile sensors. We proposed a virtual force based movement scheme to increase the coverage ratio. Our proposed scheme consists of four different forces introduced by neighboring sensors and coverage holes. By applying the resultant force on sensors, each sensor can move to sparsely area as a result the coverage ratio can be increased. Simulation results show that our proposed scheme can achieve better coverage ratio effectively. Specifically, our proposed scheme can increase the coverage ratio for 20% after 5 rounds of execution.

# References

- K. Chakrabarty, S. Iyengar, H. Qi, E. Cho, Grid coverage for surveillance and target location in distributed sensor networks, IEEE Trans. on Computers 51(12)(2002) 1448-1453.
- [2] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, A survey on sensor networks, in: Proc. ACM Trans. on Multimedia Computing, Communications and Applications, 2002.
- [3] J. Djugash, S. Singh, G. Kantor, W. Zhang, Range-only slam for robots operating cooperatively with sensor networks, in: Proc. IEEE International Conference on Robotics and Automation, 2006.
- [4] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, K.K. Saluja, Sensor deployment strategy for target detection, in: Proc. First ACM Int'l Workshop Wireless Sensor Networks and Applications, 2002.
- [5] A. Howard, M.J. Mataril, G.S. Sukhatme, Mobile sensor network deployment using potential field: a distributed scalable solution to the area coverage problem, in: Proc. the 6th Int'l Symp. on Distributed Autonomous Robotics Systems (DARS 2002), 2002.
- [6] O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, International Journal of Robotics Research 5(1)(1986) 90-98.
- [7] G. Wang, G. Cao, T.L. Porta, Movement-assisted Sensor Deployment, IEEE Trans. Mobile Computing 5(6)(2006) 640-652.
- [8] H. Lee, Y. Kim, Y. Han, C.Y. Park, Centroid-based movement assisted sensor deployment schemes in wireless sensor networks, in: Proc. IEEE 70th Vehicular Technology Conference, 2009.
- [9] F. Aurenhammer, Voronoi diagrams: a survey of a fundamental geometric data structure, ACM Computing Surveys 23(1991) 345-405.
- [10] C.K. Liang, M.C. He, C.H. Tsai, Movement assisted sensor deployment in directional sensor networks, in: Proc. the 6th international conference on Mobile Ad-hoc and Sensor Networks, 2010.