An Effective Coverage Enhancing Algorithm in Directional Sensor Networks

Chiu-Kuo Liang, Yin-Chung Hsu, and Chih-Hung Tsai
Department of Computer Science and Information Engineering
Chung Hua University
Hsinchu, Taiwan, R.O.C.
e-mail: {ckliang, m09402045, e09902009}@chu.edu.tw

***Abstract*—Directional sensor network is composed of many directional sensor nodes. Unlike conventional 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 problem is still an essential issue in directional sensor networks. In this paper, we study the area coverage problem in directional sensor networks. The problem is to maximize the area coverage of a randomly deployed directional sensor network. Each directional sensor can through rotating orientation to get better coverage in an interested region. We, therefore, propose a greedy algorithm to enhance the area coverage. Simulation results show that our proposed algorithm outperforms the previous proposed method in term of the coverage area.**

Keywords-directional sensors; coverage; greedy algorithms.

I. INTRODUCTION

In recent years, wireless sensor networks (WSNs) have received a lot of attention due to their wide applications in military and civilian operations, such as environmental monitoring, battlefield surveillance, and habitat monitoring [1], [2]. Therefore, many research topics such as area coverage, routing, and network security [3] about WSNs gain widespread attention. However, area coverage is a fundamental problem in WSNs since it reflects how well the environment is monitored, and serves as a basis for applications such as habitat monitoring and target detection [3], [4]. 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 [5], ultrasonic sensors [6] and infrared sensors [2]. 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.

For obtaining the better performance in directional sensor networks, directional sensors (e.g. cameras) may be able to rotate around a fixed axis to enhance its coverage in sensing radius [7], [8]. Therefore, the coverage region of a directional sensor is determined by both its location and its direction of sensing radius. Those sensors that can rotate their sensing directions are called the rotatable sensors. We define the working direction of a sensor as the direction in which it is currently pointing at. We also call the sensing range of a sensor's working direction as its coverage region. The coverage region of different sensors may be overlapped with other sensors after they are randomly deployed. Thus, we need to schedule sensors to face to certain directions to maximize the covered area of the whole network.

In this paper, our goal is to maximize the area coverage of a randomly deployed directional sensor network. The problem of working direction scheduling to cover maximal regions, called Maximum Directional Area Coverage (MDAC) problem, has been proved to be *NP*-complete [9]. A greedy solution has also been provided through scheduling working directions of sensors. We propose two different algorithms for MDAC problem with rotatable sensors. Simulation results show that both of our proposed algorithms outperform than the previous proposed *Face-away (FA)* algorithm [7].

The remainder of this paper is organized as follows: Related work is discussed in Section II. In Section III, the problem statement and sensing model are proposed. In Section IV, we propose two greedy algorithms for solving the problem. Section V describes the setting of our experiments and the performance metrics. In Section VI, we present experimental results obtained from different perspectives on the number of sensors, the sensing radius and the sensing angle, respectively. Section VII summarizes our findings.

II. RELATED WORK

Recently, in directional sensor networks, the coverage problem has been received a lot of attention from many researchers, not only in area coverage but in target coverage as well. The difference between area coverage [10], [11] and target coverage [12], [13] is in the measurement of the coverage performance. In the area coverage problem, we are focused on the coverage performance on the covered region while in the target coverage problem; the coverage performance on the number of covered targets is discussed. In this paper, we pay our attention to the area coverage problem. Therefore, in the following, we only discuss the recent works related to area coverage problem.

Ma and Liu [14] discuss that the number of directional sensors can be deployed to achieve coverage rate p in a distributed directional sensor network (equation (1)). Directional sensors are randomly and uniformly scattered within a given area. Here, R is the sensing radius, S is the given area, and α is the offset angle of the field of view. To be clear, $\alpha R^2/S$ indicates that a directional sensor can

 ISBN: 978-1-61208-207-3

monitor given area that is within its sensing region. Therefore, after N directional sensors are deployed, the probability that covers a given area is represented in

$$p = 1 - (1 - \frac{\alpha R^2}{S})^N. \quad (1)$$

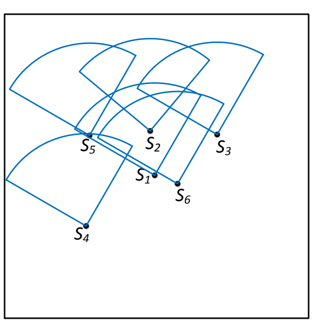
In other words, if the coverage rate of a given area is at least p, the number of deployed directional sensors should be represented in

$$N = \frac{\ln(1-p)}{\ln(S - \alpha R^2) - \ln S}. \quad (2)$$

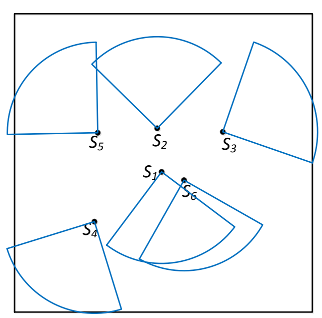
Kandoth and Chellappan [7] proposed a greedy solution called the *Face-Away (FA)* algorithm to achieve the maximal area coverage rate in the interested region. The *FA* algorithm works in a very simple manner. Each sensor calculates a new working direction that only needs the positions of neighboring sensors. The neighboring sensors of a directional sensor, say s, are those sensors located within the circular area centered at s with sensing radius R. In fact, every sensor should be recognizable from its surroundings when being viewed by its neighbors. Once a sensor is recognized, each sensor must center it in view and record the current working direction.

In Fig. 1(a), there are six sensors randomly deployed in this network, namely s_1, s_2, s_3, s_4, s_5 and s_6, each has an initial working directions. According to the *Face-away* algorithm, each sensor computes the position of its neighbors (the distance of R) by communicating with its neighbors. Each sensor will decide their working direction after rotating its angle to candidate point. The final result is shown in Fig. 1(b), where it can easily be seen that sensor s_1 are overlapped with s_6, and sensor s_2 is overlapped with s_3.

According to the above example, we can see that the *Face-away* (*FA*) algorithm cannot obtain better performance in term of the area coverage since there are still many overlapped area after scheduling by the *Face-away* algorithm, as shown in Fig. 1(b). In this paper, we propose a greedy algorithm to improve the performance of *Face-away* (*FA*) algorithm. The detailed procedure of our algorithms will be discussed in section IV.



(a) An initial deployment



(b) Final result of *Face-away*

Figure 1. An example of *FA* algorithm.

III. DIRECTIONAL SENSING MODEL

In this section, we describe the directional sensing model and notations for the Maximum Directional Area Coverage (MDAC) problem. In a directional sensor network, each directional sensor cannot sense the whole circular area. Therefore, from the concept of field of view in cameras, we can employ a 2-D model where the sensing area of a sensor is a sector denoted by 4-tuple (P, R, $\overrightarrow{W}$, α). Here P is the location of the sensor node, R is the sensing radius, $\overrightarrow{W}$ is the working direction and α is the angle of view. The common directional sensing capability for 2D spaces is illustrated in Fig. 2. The special case of this model, where $\alpha = 2\pi$ can be described as omni-sensing model.

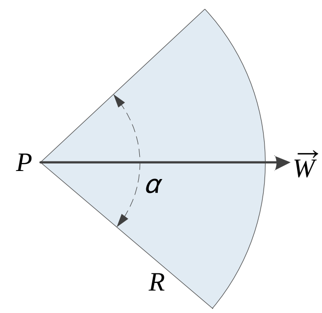


Figure 2. The directional sensing model.

We illustrate the characteristic of directional sensors:

1) Each directional sensor is homogeneous, such as: sensing angle, sensing radius, and communication radius.

2) Each directional sensor can sense only one limited angle of omni-direction.

3) Each directional sensor is fixed and can rotate arbitrary angle in sensing region.

4) The communication radius is twice than the sensing radius such that sensing neighbors can reliably communicate.

IV. THE PROPOSED GREEDY ALGORITHM

In this paper, we are going to improve the previous results in solving the Maximum Directional Area Coverage (MDAC) problem. The MDAC problem is shown to be *NP*-complete [9]. It is unlikely to solve the MDAC problem in polynomial time. Each directional sensor has an initial working direction and it has a lot of overlapped area in an interested region. Fortunately, we can rotate the sensing angle of sensors to avoid the overlapping among sensors which in a result can maximize the coverage area between directional sensors. However, since there is no global information available in a distributed environment, each directional sensor has to make its decision independently only based on its local information gathered from neighboring sensors. As we know that, although the distributed solution cannot be expected to achieve as maximal coverage as the centralized schemes, it is more computational scalable and does not incur high communication overhead as required by a centralized solution. Therefore, the localized solution is more practical and valuable. In this section, we present a distributed greedy algorithm for the MDAC problem.

Although we all know that a sensor can rotate its sensing direction to increase the coverage, we still do not know which direction is the best for a sensor to rotate. Therefore, the main idea of our proposed algorithm is to determine the most possible direction of a sensor to rotate. The following is our strategy for finding the rotating direction. We know that if there is some overlapped area in the whole circular area of two directional sensors, then these two directional sensors may have chances to overlap with each other in their sensing range after rotation. We also know that if some portion of arc on the circumference of a sensor is covered by another sensor, the whole circular area of these two sensors overlap. Therefore, we try to find the possible new direction for a directional sensor to rotate so that the possible overlapped area with its neighboring sensors is minimized. To do so, we evaluate each portion of arc on the circumference of a sensor associated with a weight indicating the degree of likelihood of that portion of arc that may be overlapped with neighboring sensors. Thus, the more weight of that portion of arc can be, the higher possibility of that portion of arc may be overlapped with other directional sensors. In our strategy, a sensor will rotate its direction to the portion of arc with least weight for reducing the overlapped area with other sensors. We describe our idea in detail in the following sections.

Let s_i and s_j be the two directional sensors, R be the sensing radius, and $d(s_i, s_j)$ be the distance between them. Then, we define the *degree of closeness* between sensors s_i and s_j, denoted as C_{ij}, as in the following equation:

$$C_{ij} = \frac{2R - d(s_i, s_j)}{2R}, if\ d(s_i, s_j) \leq 2R$$
$$= 0, \text{otherwise.}$$

Note that, the degree of closeness of two sensors indicates the degree of overlapping, which will influence the size of overlapped area. Obviously, the range of C_{ij} is [0, 1] and as the value of C_{ij} increased, the overlapped area between sensors s_i and s_j also increased.

For convenience, we also define $Arc(P, R, \vec{V}, \alpha)$ to indicate a portion of arc on the circumference of a sensor. Here P is the location of a sensor, R is the sensing radius, $\vec{V}$ is the direction and α is the angle of view. Fig. 3 shows the meaning of $Arc(P, R, \vec{V}, \alpha)$.

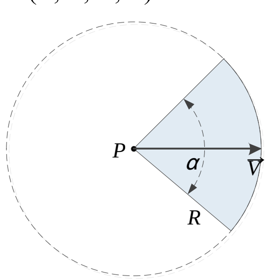


Figure 3. $Arc(P, R, \vec{V}, \alpha)$.

In order to determine which direction for a sensor s_i to rotate to achieve minimum overlapped area with other sensors, we evaluate the weights of the points on the arc of the circle of sensor s_i which can indicate the possibilities of s_i to overlap with another sensor, say s_j, if s_i rotates its working direction to face the new direction of the arc. The weights can be evaluated accordingly based on the following different overlapping situations: (a) $d(s_i, s_j) \geq 2R$, (b) $\sqrt{2}R \leq d(s_i, s_j) < 2R$, (c) $R \leq d(s_i, s_j) < \sqrt{2}R$, and (d) $d(s_i, s_j) < R$. Fig. 4 shows the different situations. In Fig. 4(a), since there is no overlapped area, the weights of all points on the circle of sensor s_i are zero. In Fig. 4(b), the points on the arc from intersection point x to point y along with the clockwise direction will be evaluated. In Fig. 4(c), the points on the arc from the point u to point v along with the clockwise direction will be evaluated, where u and v are the intersection points of the circle of s_i and the tangent lines from s_i to the circle of s_j. Finally, in Fig. 4(d), all the points on the circle of s_i will be weighted since there exist some overlapped area between sensors s_i and the circle of sensor s_j regardless the rotation of sensor s_i.

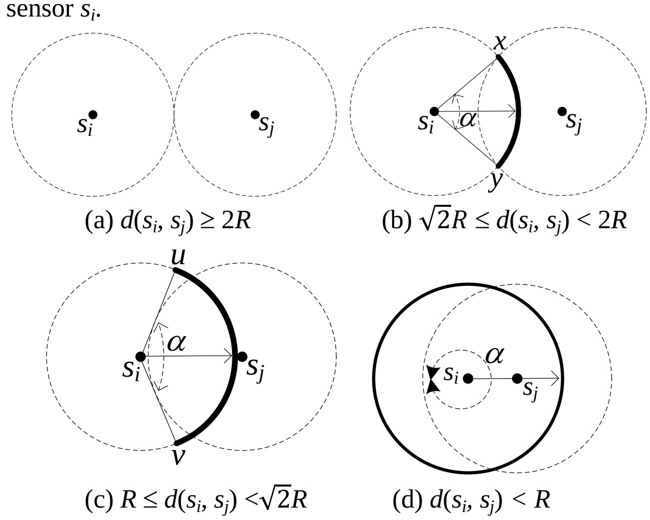


Figure 4. Four overlapping situations between s_i and s_j.

According to Fig. 4, we know that once two sensors, say s_i and s_j, are intersected, then part of the arc on the circles of s_i and s_j should be evaluated for the weight. The weight of a point on the arc can be evaluated as follows. Let $Arc(P, R, \vec{V}, \alpha)$ be an arc on the circle of a sensor , T be a point on the arc and β be the angle between $\overrightarrow{PT}$ and $\vec{V}$. Then the weight of T is evaluated according to the following equation:

$$W(T) = \frac{\alpha/2 - \beta}{\alpha/2} \cdot C_{ij}$$

where C_{ij} is the degree of closeness between s_i and s_j. Note that according to the above equation, the weights of the points on the arc $Arc(P, R, \vec{V}, \alpha)$ will be increasing from the starting point, say s, to the center line $\vec{V}$ and then decreasing to the ending point, say t, as shown in Fig. 5.

 ISBN: 978-1-61208-207-3

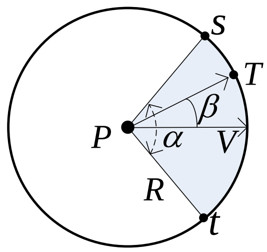


Figure 5. The weights of the points on an arc.

For clarity, we summarize the weight evaluation methodology in this study as follows:

1) $d(s_i, s_j) \geq 2R$. As shown in Fig. 4(a), the weights of all points on the arc of circles of s_i and s_j are zero.

2) $\sqrt{2}R \leq d(s_i, s_j) < 2R$. As shown in Fig. 4(b), the effective arc of sensor s_i is arc Arc(s_i, R, $\vec{V}$, α), where α is the angle between $\overrightarrow{s_i x}$ and $\overrightarrow{s_i y}$, and the weights of points on the arc are computed according to the weighting function.

3) $R \leq d(s_i, s_j) < \sqrt{2}R$. As shown in Fig. 4(c), the effective arc of sensor s_i is arc $Arc(s_i, R, \vec{V}, \alpha)$, where α is the angle between $\overrightarrow{s_i u}$ and $\overrightarrow{s_i v}$, and the weights of points on the arc are computed according to the weighting function.

4) $d(s_i, s_j) < R$. As shown in Fig. 4(d), the effective arc of sensor s_i is arc $Arc(s_i, R, \vec{V}, \alpha)$, where $\alpha = 2\pi$, and the weights of points on the arc are computed according to the weighting function.

Here, we describe our proposed greedy algorithm for MDAC problem. The proposed algorithm is called the *Maximal Overlapped-Area First* (MOAF) algorithm. It should be recalled that the basic idea of FA algorithm is to find the largest angle between adjacent directions and makes the bisector to that angle as the new working direction. However, the FA algorithm did not take the overlapped area between sensors into account. Therefore, the increasing coverage rate that can be obtained from the FA algorithm is limited. On the contrary, our proposed MOAF algorithm will consider the overlapped area as priority of each sensor. As we mentioned above, we evaluate the weight of points on the circle of each sensor to indicate the possibilities that could be overlapped with other sensors after rotation. In our algorithm, the total weight of the circle of a sensor is considered to be the priority for the sensor. Therefore, if the weight of a sensor is high, meaning that the sensor has many neighbors and the size of overlapped area is large, which as a result the sensor has higher priority to be scheduled for rotation.

Once a sensor has determined to rotate, the new direction can be obtained by finding the point, say A, on the circle with least weight value. Then, the working direction $\overrightarrow{W}$ of the sensor will be rotate to the direction of $\overrightarrow{PA}$.

The pseudo-code of the *Maximal Overlapped-Area First* (MOAF) algorithm is shown as follows:

Algorithm: Maximal Overlapped-Area First algorithm

I. Initialization Phase (only performed once)

send a coverage message containing sensor ID and location of sensor s_i
calculate the weights of all points on the arc of circle that is overlapped with its neighbors after waiting for a period of time to collect the coverage messages from sensing neighbors
determine the priority value P_i and broadcast the value
collect the priority values from all of its neighboring sensors and go to the **Decision Phase**.

II. Decision Phase

while *true* **do**
    find the highest priority values, denoted as P_{max}, among neighboring sensors
    if $P_i > P_{max}$ **then**
        find the point, say A, on the circle with least weight value
        rotate its working direction to point A, set its priority value to 0 and send a scheduled message containing ID and priority value to its sensing neighbors
        Exit the while loop
    else
        if $P_i = P_{max}$ **then**
            wait for a random duration or a scheduled message sent by a sensor, say s_j, is received
            if no scheduled message received **then**
                find the point, say A, on the circle with least weight value
                rotate its working direction to point A, set its priority value to 0 and send a scheduled message containing ID and priority value to its sensing neighbors.
                Exit the while loop
            end if
        else
            wait until a scheduled message sent by a sensor, say s_j, is received
        end if
        set the status of s_j as "scheduled" and update its priority P_i according to its remaining "unscheduled" neighboring sensors
        send P_i to its "unscheduled" neighboring sensors
        collect the priority values from all of its "unscheduled" neighboring sensors
    end if
end while

V. SIMULATION RESULTS

This section describes the parameters and performance effects of different perspectives on our proposed algorithms.

We conducted our experiments on a computer with 3.0 GHz CPU and 4GB memory. All experiments are done in C# on .NET platform. Our simulation network consists of 50 to 200 directional sensor nodes placed randomly within a 500 *m* x 500 *m* area. Every experiment was repeated 100 times and the recorded data was averaged over those runs. Table 1 lists the values of the common parameters used in all the experiments.

TABLE I. EXPERIMENTAL PARAMETER

Parameters	Description
Network Size	500 × 500 (m^2)
Sensing Radius	30*m*, 35*m*, …, 60*m*
Sensing Angle	60°,80°, …, 180°
Number of Sensors	50, 75, …, 200

The main goal of our simulation is focused on the evaluation of the performance of our proposed algorithms in term of the coverage rate. The coverage rate p is defined as the ratio of the total covered area by all sensors over the network size. We evaluate the effects of our algorithm on three different perspectives. First, we examine the effect that the number of sensors N makes to the improvement of coverage rate p. Second, we evaluate the effect that the sensing radius improves the coverage rate p. Third, we examine the effect that the offset angle makes to the improvement of coverage rate p.

Following are our simulation results that demonstrate the effects of our coverage-enhancing algorithms.

VI. SIMULATION RESULTS

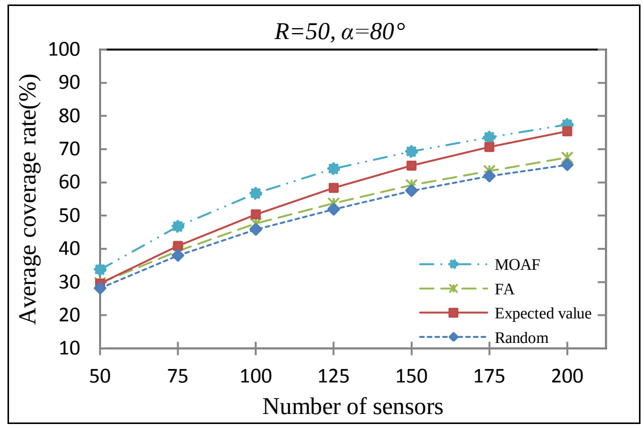
We evaluate the performance of the *Maximal Overlapped-Area First* (MOAF) algorithm, the *Random approach* (Random) in which each sensor select its sensing direction randomly and the *Face-Away* (FA) algorithm. Moreover, we compare the simulation results with the theoretic solution, denoted as *Expected Value*, which are obtained by (1).

A. Coverage rate vs. Number of sensors

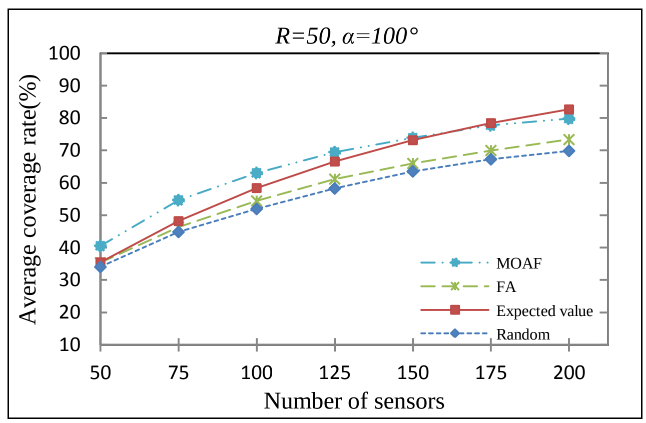
This experiment evaluates the effect that the number of sensors N makes to the performance of coverage rate p of Random approach, FA algorithm, and the MOAF algorithm, respectively. The sensing radius R is set to 50m. We first set the sensing angle α to 80°, and then repeated the experiment with sensing angle equals to 100°. The results are shown in Figure 6.

In these graphs, we can see that our proposed MOAF algorithm outperforms Face-away (FA) algorithm and Random approach. For example, when the number of sensors is 200, the sensing radius is 50m and the offset angle is 40°, the coverage rates of Random approach, Face-away (FA) algorithm, and MOAF algorithm are 65.31%, 67.48%, , and 77.42% respectively. Thus, our proposed MOAF algorithm performs 9.94% better than FA algorithm. This is because that our MOAF algorithm can achieve the less overlapping area and the order of sensors chosen to rotate will influence the performance of coverage rate. Therefore, our proposed MOAF algorithm can get the most improvement on coverage rate among all algorithms.

Furthermore, in these graphs, we observe some similar behaviors. We can see from comparing Fig. 6(a) and Fig. 6(b), as the sensing angle α increases (α increases from 80° to 100° in this experiment), the coverage rates of all algorithms increase. This is obvious since the larger the offset angle is, the more area can be covered. Similarly, as the number of sensor nodes N increases, the average coverage rate p also rises. However, once the value of N exceeds a certain value (≥ 150 in this experiment), the increasing coverage rate becomes flat rising. This is because, when the sensing radius and offset angle are fixed, the greater the network density is, the smaller the possibility of uncovered area becomes.



(a)



(b)

Figure 6. Coverage rate – Different sensing angles. (a) α = 80° (b) α = 100°

B. Coverage rate vs. Sensing radius

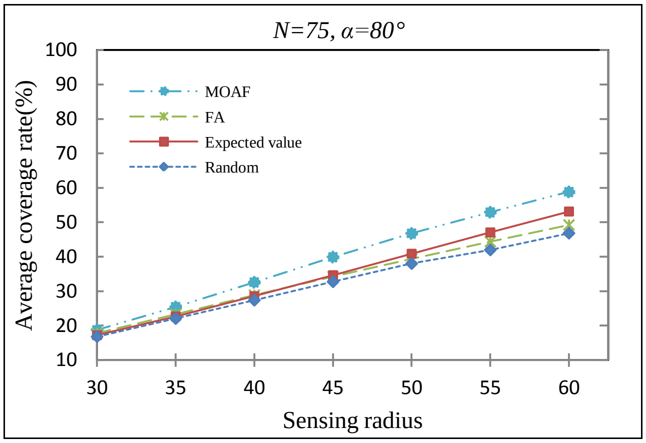
This experiment examines the effect that sensing radius R makes to the performance of coverage rate p of Random approach, FA algorithm, and the MOAF algorithm, respectively. The sensing offset angle is set to 80°. We first set the number of sensors to 75, and then repeated the experiment with the number of sensors equals to 150. The results are shown in Figure 7.

In these graphs, we can see that our proposed MOAF algorithm outperforms FA algorithm and Random approach. For example, when the number of sensors is 150, the sensing

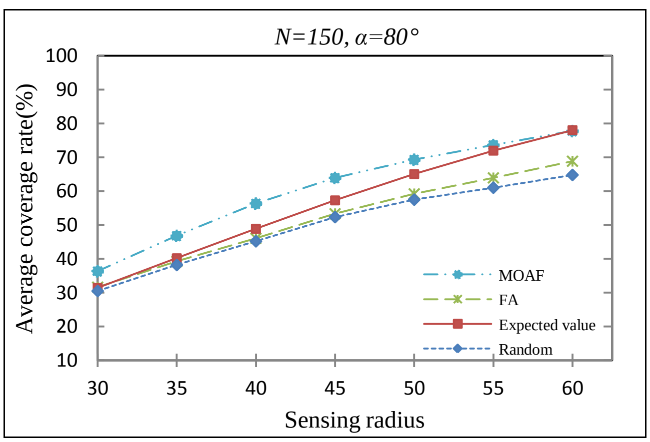
 ISBN: 978-1-61208-207-3

radius is 50*m* and the sensing angle is 80°, the coverage rates of Random approach, Face-away (FA) algorithm, and MOAF algorithm are 57.50%, 59.22%, and 69.29% respectively. Thus, our proposed MOAF algorithm performs 10.07% better than FA algorithm. This is because that our proposed MOAF algorithm can achieve less overlapping area and higher coverage rate. We also note that, as the sensing radius increases, the coverage rates of all algorithms rise. This is obvious since the greater the sensing radius is, the more sensing area can be obtained.

Furthermore, we can see from comparing Fig. 7(a) and Fig. 7(b), as the number of sensors N increases (N increases from 75 to 150 in this experiment), the coverage rates of all algorithms increase. This is also obvious to be seen that, when the sensing radius and offset angle are fixed, the greater the network density is, the smaller the possibility of uncovered sensing area becomes.



(a)



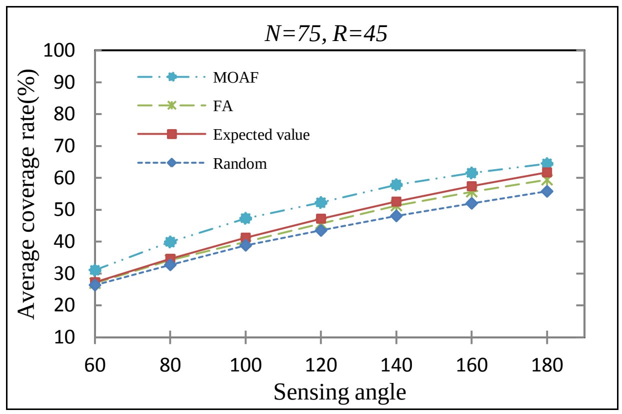
(b)

Figure 7. Coverage rate of sensing radius – Different number of sensors. (a) $N = 75$ (b) $N = 150$

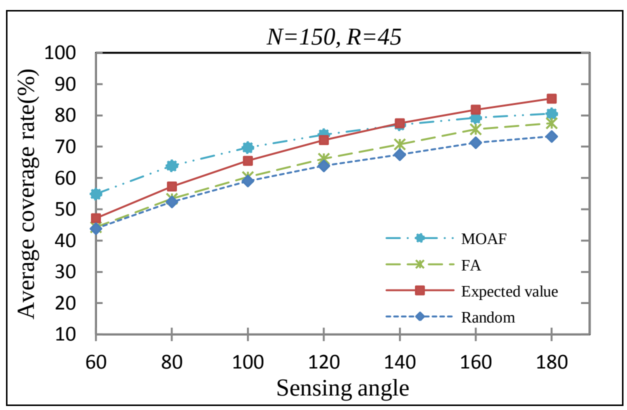
C. Coverage rate vs. Network size

This experiment evaluates the effect that sensing angle α makes to the performance of coverage rate p of Random approach, FA algorithm, and the MOAF algorithm, respectively. The sensing radius is set to 45. We first set the number of sensors to 75, and then repeated the experiment with the number of sensors equals to 150. The results are shown in Figure 8.

In these graphs, we can see that our proposed MOAF algorithm outperforms Face-away (FA) algorithm and Random approach. For example, when the number of sensors is 150, the sensing radius is 45*m* and the sensing angle is 60°, the coverage rates of Random approach, Face-away (FA) algorithm, and MOAF algorithm are 63.86%, 66.15%, and 73.86% respectively. Thus, our proposed MOAF algorithm performs 7.71% better than FA algorithm. This is because that our proposed MOAF algorithm can achieve less overlapping area and higher coverage rate. We also note that, as the sensing angle α increases, the coverage rates of all algorithms rise. However, once the value of α exceeds a certain value (≥ 120° in this experiment), the increasing coverage rate becomes flat rising. This is because, when the network density and sensing radius are fixed, the larger the sensing angle is, the smaller the possibility of uncovered area becomes.



(a)



(b)

Figure 8. Coverage rate of sensing angle – Different number of sensors. (a) $N = 75$ (b) $N = 150$

VII. CONCLUSION AND FUTURE WORK

In this paper, we investigated the Maximum Directional Area Coverage (MDAC) problem in which we are asked to maximize the area coverage by scheduling the sensing direction or rotating the working direction of each sensor. We propose a greedy algorithm, called the Maximal Overlapped-Area First (MOAF) approach, which is based on the size of overlapped area between directional sensors. Simulation results show that our proposed algorithms both outperform the previous algorithm in terms of coverage rate on different number of sensors, sensing radius and sensing angle. In the future, we will pay our attention to find the solutions for minimizing the energy consumption while maximizing the coverage rate.

REFERENCES

[1] M. Li and Y. Liu, "Underground Structure Monitoring with Wireless Sensor Networks," Proceedings of ACM/IEEE IPSN, Massachusetts, USA, pp. 69-78, 2007.

[2] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler, "An analysis of a large scale habitat monitoring application," in ACM Conference on Embedded Networked Sensor Systems (SenSys), pp. 214-226, 2004.

[3] M. Amac Guvensan, and A. Gokhan Yavuz, "On coverage issues in directional sensor networks: A survey," Ad Hoc Networks, vol. 9, no. 7, pp. 1238-1255, 2011.

[4] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Coverage problems in wireless ad-hoc sensor networks", in IEEE INFOCOM, Anchorage, USA, pp. 1380-1387, 2001.

[5] H. D. Ma and D. Tao, "Multimedia sensor network and its research progresses," Journal of Software, vol. 17, no.9, pp. 2013-2028, 2006.

[6] J. Djugash, S. Singh, G. Kantor, and W. Zhang, "Range-only slam for robots operating cooperatively with sensor networks," Proceedings of IEEE International Conference on Robotics and Automation, pp. 2078-2084, 2006.

[7] C. Kandoth and S. Chellappan, "Angular Mobility Assisted Coverage in Directional Sensor Networks," *Proceedings of International Conference on Network-Based Information Systems* (NBIS 2009), pp. 376-379, 2009.

[8] D. Tao, H. D. Ma, and L. Liu, "Coverage-Enhancing Algorithm for Directional Sensor Networks," Proceedings of 2nd International Conference, Mobile Ad-hoc and Sensor Network, Hong Kong, China, pp. 256–267, 2006.

[9] W. Cheng, S. Li, X. Liao, C. Chen, and H. Chen , "Maximal Coverage Scheduling in Randomly Deployed Directional Sensor Networks" in International Conference on Parallel Processing Workshops, ICPPW 2007, pp. 68, 2007.

[10] Y. Cai, W. Lou, and M. Li, "Target-Oriented Scheduling in Directional Sensor Networks," in IEEE INFOCOM, pp. 1550–1558, 2007.

[11] X. Han, X. Cao, E. L. Lloyd, and C.-C. Shen, "Deploying Directional Sensor Networks with Guaranteed Connectivity and Coverage," *Proceedings of 5th Annual 2008 IEEE conference on SECON*, pp. 153-160, 2008.

[12] Y. Cai, W. Lou, and M. Li, "Cover Set Problem in Directional Sensor Networks," *Proc. of IEEE International Conference on Future Generation Communication and Networking (FGCN '07)*, pp. 274-278, 2007.

[13] U. R. Chen, B. S. Chiou, J. M. Chen, and W. Lin, "An Adjustable Target Coverage Method in Directional Sensor Networks," in IEEE Asia-Pacific Services Computing Conference(APSCC 2008), pp. 174–180, 2008

[14] H. D. Ma and Y. Liu, "On coverage problems of directional sensor networks," in International Conference on Mobile Ad-hoc and Sensor Networks (MSN), pp. 721-731, 2005.

ISBN: 978-1-61208-207-3