Scientific
Research

Voronoi-Based Coverage Optimization for Directional Sensor Networks

Jing LI[1], Ruchuan WANG[1,2,3], Haiping HUANG[1], Lijuan SUN[1,3]

[1]*College of Computer, Nanjing University of Posts and Telecommunications, Nanjing, China*
[2]*State Key Laboratory of Information Security, Graduate School of Chinese Academy of Sciences, Beijing, China*
[3]*Institute of Computer Technology, Nanjing University of Posts and Telecommunications, Nanjing, China*
Email: *leejnjupt@126.com, {wangrc, hhp}@njupt.edu.cn*
Received July 25, 2009; *revised August* 3, 2009; *accepted August* 4, 2009

Abstract

Sensing coverage is a fundamental problem in sensors networks. Different from traditional isotropic sensors with sensing disk, directional sensors may have a limited angle of sensing range due to special applications. In this paper, we study the coverage problem in directional sensor networks (DSNs) with the rotatable orientation for each sensor. We propose the optimal coverage in directional sensor networks (OCDSN) problem to cover maximal area while activating as few sensors as possible. Then we prove the OCDSN to be NP-complete and propose the Voronoi-based centralized approximation (VCA) algorithm and the Voronoi-based distributed approximation (VDA) algorithm of the solution to the OCDSN problem. Finally, extensive simulation is executed to demonstrate the performance of the proposed algorithms.

1. Introduction

In recent years, wireless sensor networks (WSNs) have attracted intense interests due to their wide applications in military and civilian operations, such as environmental monitoring, battlefield surveillance, and health care [1–3]. The conventional research of WSNs is always based on isotropic sensors which can equally detect the environment in each orientation. However, sensors may have a limited angle of sensing range due to special applications, which are denoted as directional sensors. There are many kinds of directional sensors, such as video sensors [4,5], ultrasonic sensors [6] and infrared sensors [2]. The most familiar directional sensors are the video sensors widely used in wireless multimedia sensor networks (WMSNs). Note that the directional characteristic we discuss in this paper is from the point of view of the sensing, but not from the communicating activity of sensors.

Sensing coverage is a fundamental issue in all sensor networks, which has been studied by many works. Due to the large variety of sensors and applications, coverage is subject to a wide range of interpretations. In general, coverage can be considered as the measure of quality of service (QoS) in a sensor network. For example, the famous Art Gallery Problem [7] deals with determining the number of observers necessary to cover an art gallery room such that every point is detected by at least one observer.

One approach to sensing coverage is the deployment of sensors, which is a critical issue in existing works. A well-planned deployment can help maximize the sensing coverage while activating as few sensors as possible. A. Ghosh [8] studies the problem of coverage holes under random deployment. An algorithm is proposed to achieve a tradeoff between the cost of deployment and the percentage of area covered. In [9] and [10], two algorithms are proposed to efficiently deploy sensors and to maximize the coverage.

Different from traditional sensors, the coverage of a directional sensor is determined by both its location and orientation. Ma *et al* [11] provide a directional sensor model where each sensor is fixed to one direction and analyzes the probability of full area coverage. In [12,13], Tao presents two coverage-enhancing algorithms to minimize the overlapping sensing area of directional sensors, according to the characteristic of adjustable orientations of directional sensors. In [14–16], some algorithms are proposed to cover maximal number of targets. Unfortunately, they limit orientations of a directional sensor, i.e. the sensor they assume cannot rotate its orientation continuously. The previous works assume commonly that after directional sensors are deployed randomly, they adjust orientations to achieve efficient coverage.

Given a directional sensor network, we are interested

in designing a deployment algorithm that is able to optimize coverage. S. Megerian *et al* [17] present the worst-case coverage in WSNs. They attempt quantifying the QoS by finding areas of lower observability from sensors and detecting *breach* regions, where breach is defined as the minimum Euclidean distance from any point on a path to any sensor. Due to find a path, where minimize the probability of detecting the target moves along this path, they define *maximal breach path*. Similarly, J. Adriaens *et al* [18] put forward an optimal polynomial time algorithm for computing the worst-case coverage in DSNs. Both of them present centralized methods using the Voronoi diagram to solve the worst-coverage problem. The use of Voronoi diagram, efficiently and without loss of optimality, transforms the continuous geometric problem into a discrete graph problem.

In this paper, we mainly address the problem of maximal coverage while activating as few sensors as possible, called optimal coverage in directional sensor networks (OCDSN) problem. For the solution to this problem, we will propose two optimizing coverage algorithms based on the Voronoi diagram. Both of two algorithms, called the Voronoi-based centralized approximation (VCA) algorithm and the Voronoi-based distributed approximation (VDA) algorithm, contribute to making edges of Voronoi cover as more as possible, which could keep the worst-case coverage. Further, it is approximately considered that if most paths are covered, most Voronoi polygons are covered, i.e. most of the given monitoring area is covered.

From the perspective of directional sensors, the effect of coverage has three parameters: the number of sensors, sensing range and the field of view (FOV). As shown in the following sections, we will discuss the impact of these parameters on the coverage of a directional sensor network.

The rest of this paper is organized as follows: We formally establish a mathematical model for representing the FOV in Section 2, and then define the OCDSN problem and prove that it is NP-complete in Section 3. Furthermore, we propose VCA and VDA of solution to the OCDSN in Section 4. Simulation results are presented to show the effectiveness of the proposed algorithms in Section 5. Finally, we conclude the paper in Section 6.

2. Preliminaries

2.1. Sensing model

A directional sensor has a finite angle of view. From the concept of FOV, its sensing region can be viewed as a sector in a two-dimensional plane denoted by 4-tuple (L, R_S, α, β) as shown in Figure 1. The sensing model of a directional sensor is defined as follows.
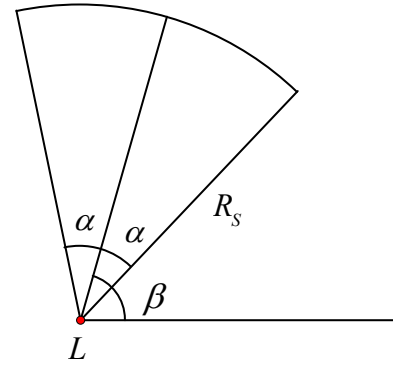

Figure 1. Sensing model of a directional sensor.

Definition 1 *The directional sensing model:* 4-tuple (L, R_S, α, β)

- L: the location of the sensor.
- R_s: sensing range of the sensor.
- α: the offset angle of FOV , which equals to the half of the vertex angle of sensing sector.
- β: the horizontal angle to the midline of sensing sector, $0 \leq \beta \leq 2\pi$. This parameter defines the orientation of the directional sensor.

2.2. Notations and Assumptions

We adopt the following notations throughout the paper.
- A: the given specified area to be covered.
- N: the number of sensors.
- S_i: the ith sensor, $1 \leq i \leq N$.
- S: the set of sensors. $S = \{s_1, s_2, ..., s_N\}$.
- $\varphi_{i,\beta}$: the coverage of S_i whose orientation is β, $1 \leq i \leq N$, $0 \leq \beta \leq 2\pi$.
- Φ: the set of the coverage of all sensor. $\Phi = \{\varphi_{i,\beta} \mid i = 1, 2, ... N, \ 0 \leq \beta \leq 2\pi\}$.
- $P \in \varphi_{i,\beta}$: the point P covered by S_i, $1 \leq i \leq N$, $0 \leq \beta \leq 2\pi$.

A point P is said to be covered by sensor S_i if and only if the following conditions are satisfied:

1) $dis(L_i, P) \leq R_S$, where $dis(L_i, P)$ is the Euclidean distance between the location L_i of sensor S_i and point P. Since S_i corresponds to L_i, throughout the rest of the paper, unless otherwise mentioned, $dis(s_i, P)$ means $dis(L_i, P)$.

2) The horizontal angle to $\overrightarrow{L_iP}$ is within [$\beta_i - \alpha$, $\beta_i + \alpha$].

An area A is covered by sensor S_i, if and only if for any point $P_j \in A (j = 1, 2, ..., \infty)$, P_j is covered by S_i, i.e. $P_j \in \varphi_{i,\beta}$.

For simplicity and computability, we make assumptions as below, however some of them can be relaxed.

- Directional sensors are homogeneous. Specifically, all sensors have the same omnidirectional communication range R_c and shape of sensing sectors (i.e. R_s and α).
- The communication range is at least twice as large as the sensing range, i.e. $R_c \geq 2R_s$.
- Every directional sensor knows its exact location information.
- Each direction of every sensor has a uniform sensing sector.

2.3. Geometry Notation

The Voronoi diagram is an important data structure in computational geometry, which is a fundamental construct defined by a discrete set of sites [19]. In a two-dimensional plane, the Voronoi diagram partitions the plane into a set of convex polygons such that all points inside a polygon are closest to only one site. This construction effectively produces polygons with edges that are equidistant from neighboring sites.

We define the Voronoi polygon of S_i as $VP(S_i)$. According to the property of Voronoi diagram, if an arbitrary point $P \in VP(s_i)$ then $dis(P, s_i) \leq dis(P, s_j)$, where $i, j = 1, 2, ..., N$ and $i \neq j$. Therefore, if a sensor cannot detect the expected phenomenon in its Voronoi polygon, no other sensor can detect it (assume that the directional sensor can adjust its orientation circularly). The Voronoi diagram generated by the set of sensors S is defined as $VD(S) = \bigcup_{s_i \in S} VP(s_i)$.

However, some of the lines at the boundaries of the Voronoi diagram extend to infinity. In this paper, the monitoring area with our supposal is finite, so we clip the Voronoi diagram within the boundary polygons. To do this, we introduce extra edges in the Voronoi diagram corresponding to the boundary edges and discard any line segments that lie outside the boundary of the field as shown in 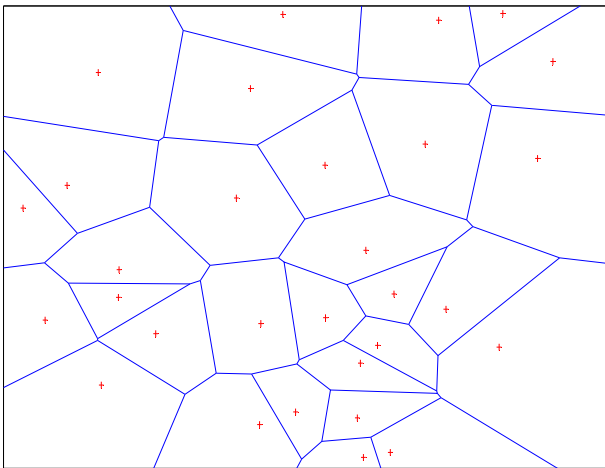Figure 2. Let $BVD(S)$ be the graph by removing all edges of $VD(S)$ that no longer than the given area A, i.e. $BVD(S) = VD(S) \bigcap A$.

3. Problem Definition

Sensors are randomly deployed when we initialize the network, so the whole monitoring area is not always covered by this initial deployment. Further, it is unnecessary that all sensors are active. Our goal is to schedule the orientations in order to cover maximal area while activating as few sensors as possible, called the optimal coverage in directional sensor networks (OCDSN) problem. This problem can be defined as follows.

Definition 2 *Optimal coverage in directional sensor networks (OCDSN) problem*: Given a specified area A, a set of directional sensors S, and each sensor with four parameters $L_i R_s$, $α$ and $β$, find a subset Z of Φ, with the constraint that at most one $\varphi_{i,\beta}$ can be chosen for the same i (i.e. an active sensor has only one orientation), to maximize the union of chosen $\bigcup \varphi_{i,\beta}$ (i.e. the covered area), while minimizing the cardinality of $Z = \{\varphi_{i,\beta} \,|\, (i, \beta)$ is chosen$\}$ (i.e. the number of active directional sensors).

Definition 3 *Decision version of OCDSN*: Given a specified area A, a set of directional sensors S, and each sensor with four parameters $L_i R_s$, $α$ and $β$, determine if there exists a subset Z of Φ with u_0 elements of Φ, with the constraint that at most one $L_i R_s$, $α$ and $β$, can be chosen for the same i, covering at least $p_0 A$, where u_0 is a given positive integer from 1 to N and p_0 is a given positive pure decimal called the required ratio of coverage.

The following theorem shows the complexity of the OCDSN problem.

Theorem 1 *The OCDSN problem is NP-complete.*
Proof: We follow two steps to prove this theorem.

First, we prove that OCDSN $\in$ NP. The non-deterministic OCDSN algorithm is described as follows: Select u_0 elements of the set of directional sensors S and assign a random orientation to each of these selected sensors, so that 2-tuple (i, β) corresponds to $\varphi_{i,\beta}$. Then check that if the union of chosen sensors covers at least $p_0 A$, i.e. $\bigcup \varphi_{i,\beta} \geq p_0 A$. It is easy to see this non-deterministic algorithm can be verified in a polynomial time. Therefore, OCDSN $\in$ NP.

Second, to prove that the decision version of OCDSN is NP-hard, we show a polynomial time reduction 3-CNF-SAT to OCDSN, i.e. 3-CNF-SAT $\propto$ OCDSN. Before the proof, we suppose a special case where sensors are isotropic (i.e. $α = \pi$) and it requires the whole coverage (i.e. $p_0 = 1$). Furthermore, a plane can be regarded as the set of infinite points, so we assume that our goal to cover the area A is equivalent to covering infinite points.

Figure 2. Boundary Voronoi diagram.

For the 3-CNF-SAT problem, a Boolean formula F consisting of infinite clauses and N variables is in 3-conjunctive normal form, i.e. $F = \bigwedge_{j=1}^{\infty} c_j$, where each clause $c_j = l_{j,1} \vee l_{j,2} \vee l_{j,3}$ and each literal $l_{j,k} \in \{l_1, \bar{l}_1, ..., l_N, \bar{l}_N\}$. $(k = 1, 2, 3)$ From the given formula F, an instance of OCDSN is constructed as follows:

- $A = \{c_j \mid j = 1, 2, ...\}$.

- $\varphi_{i,0} = \{c_j \mid l_i \text{ is a literal in } c_j, j = 1, 2, ...\}$.

- $\varphi_{i,\pi} = \{c_j \mid \bar{l}_i \text{ is a literal in } c_j, j = 1, 2, ...\}$.

This construction can be finished in a polynomial time.

Now we prove that F is satisfiable if and only if A is covered. If F is satisfiable, then there is a set of truth values for $l_i, i = 1, 2, ..., N$, such that each clause is true with this assignment. Thus, with this assignment at least one literal is true in each clause. Since each literal corresponds to a $\varphi_{i,0}$ or $\varphi_{i,\pi}$, picking these true literals yields a subset Z of Φ, where the cardinality of Z is the number of active sensors, i.e. $u_0 = \|Z\|$. Note that l_i and $\bar{l}_i$ cannot both be true, so the corresponding $\varphi_{i,0}$ and $\varphi_{i,0}$ in Φ cannot both be chosen into Z. Therefore, A is covered by Z.

Conversely, if A is covered by a subset Z of Φ and we suppose that $\|Z\| = N$, then we assign true to the literals which elements in Z correspond to. Obviously, every clause is true because at least one of its literals is true. Therefore, F is satisfiable.

We conclude that 3-CNF-SAT $\propto$ OCDSN. The 3-CNF-SAT problem is known to be NP-complete, so the OCDSN problem is NP-hard.

Since the OCDSN problem is NP and NP-hard, the result follows.

4. Solution

Since the OCDSN problem is NP-complete, we propose two algorithms, the centralized algorithm and the distributed algorithm based on Voronoi diagram, both of which solve the OCDSN problem efficiently.

4.1. Voronoi-Based Centralized Approximation (VCA) Algorithm

To solve the NP-complete OCDSN problem as well as possible, we present the VCA algorithm that needs the global information.

The main idea of VCA is based on the greedy principle and can be described as follows. Initially, we deploy a set of directional sensors S randomly in the monitoring area A, all of which are inactive (i.e. not active). Sec-

ondly generate $BVD(S)$ and construct $F = \{maxlen\ (S_i), i = 1, 2, ..., N\}$, where $maxlen\ (S_i)$ is the maximal length of uncovered edges of $BVD(S)$ which S_i can be possible to cover. VCA runs in loops. In each loop, calculate the maximal element of F and let S_j, the corresponding sensor, be active. Rotate the orientation of S_j to make it cover the maximal length of uncovered edges, remove S_j from F and update F. If there are no more edges of $BVD(S)$ to be covered or no more inactive directional sensors remaining, i.e. F is empty, the algorithm terminates; otherwise, directional sensors are activated iteratively according to the above greedy rule.

The pseudo-code of the VCA algorithm is shown below.

VCA algorithm

Deploy N sensors randomly in monitoring area
  location$(i) = (x_i, y_i)$
  state$(i) =$ 'inactive'   // Initialize the network, and
let all sensors be inactive.
Generate $BVD(S)$
Construct $F = \{maxlen(s_i), i = 1, 2, ..., N\}$
while F is not empty && there is one edge uncovered
    Calculate the maximal element of F
    $s_j = \arg\max\{maxlen(s_i)\}$
    state$(j) =$ 'active'
    Rotate S_j orientation to make it cover the maximal
    length of uncovered edges
    Remove S_j from F
    Update F
end while

Given N sensors, the best known algorithms for the generation of the Voronoi diagram have $O(N\log N)$ complexities. In 2D, Voronoi diagrams are essentially linear complexity in terms of vertices and edges. For N sensors, $\|E\|$ (numbers of edges) in the Voronoi diagram is $O(N)$, so constructing and updating F need $O(N^2)$. The best order algorithms have $O(N\log N)$ complexities, thus calculating the maximal element needs $O(N\log N)$. The complexity of while loop is $O(N(N\log N + N^2)$. Therefore, the total time of the VCA algorithm is as follows:

$$O(N\log N) + O(N^2) + O(N(N\log N + N^2))$$
$$= O(N(N\log N + N^2))$$

4.2. Voronoi-Based Distributed Approximation (VDA) Algorithm

As above, we can achieve the solution to OCDSN problem approximately in polynomial time. However, the VCA algorithm needs the global information, which in-

curs high cost. Without global information available in a centralized location, each sensor must make its decision independently based only on local information gathered from the *neighbors*. In this subsection, we present the VDA algorithm.

Definition 5 *The set of neighbors of* S_i:

$$Neighbor(i) = \{s_j \in S \mid dis(s_i, s_j) \le 2R_S, j = 1, 2, ..., N\}.$$

Definition 6 *The weight of* S_i:

$$w_i = \begin{cases} \dfrac{\sum\limits_{j=1}^{\|E_i\|} \|e_{i,j}\|}{\|E_i\|}, & \text{if } \|E_i\| \ne 0, \\ 0, & \text{otherwise} \end{cases}$$

where $\|e_{i,j}\|$ is the length of e_{ij} and e_{ij} is the jth uncovered edge of $BVD(S)$ that S_i has the capability of sensing, and E_i is the set of all e_{ij}. Indeed, w_i represents the priority among neighbors of S_i.

Definition 7 *Local Voronoi diagram of* S_i:

$$LVD(s_i) = C(s_i) \bigcap BVD(S),$$

where $C(S_i)$ is the communication region of S_i, i.e. $C(s_i) = \{P \in A \mid dis(s_i, P) \le R_C\}$.

As shown in Figure 3, each directional sensor can calculate its local Voronoi diagram by collecting locations of all sensors in its communication region. Note that we assume $R_C \ge 2R_S$, so $\varphi_{i,\beta} \in C(s_i)$. Therefore we can discuss the coverage of each sensor in its local Voronoi diagram respectively.
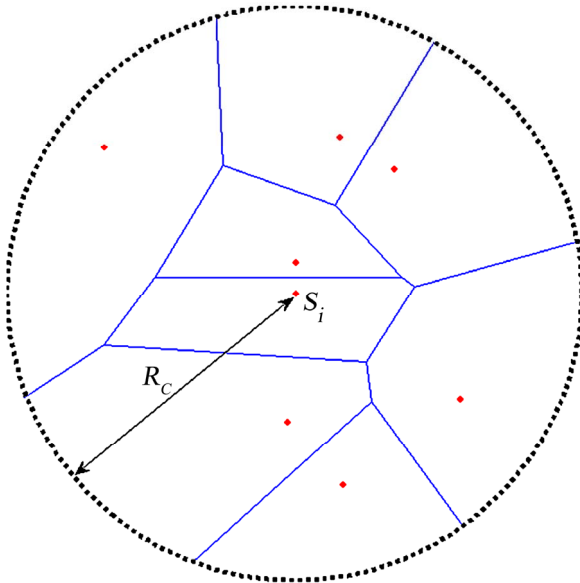
The VDA algorithm is simply described as follows.

Figure 3. Local Voronoi diagram of S_i**.**

Initially, each directional sensor is in the active state and collects locations of all sensors in its communication region. Then each sensor generates local Voronoi diagram respectively, and calculates its weight. Each sensor starts to collect the information of its neighbors, i.e. weights, covered edges, and states. Upon receiving the information and updating its weight, each sensor makes its decision independently as follows. If its weight is maximal, it chooses the orientation for the purpose of covering the maximal edge and sends out a new message to inform its neighbors. If its weight is zero, it triggers a transition timer, with random duration T_r. The timer is canceled if new information from the neighbors arrives and changes the weight to a non-zero value. Note that the purposes of setting the transition timer T_r. are 1) to prevent a sensor finalizing its decision before its neighbors with higher weight and 2) to transfer its state to inactive in time.

The pseudo-code of the VDA algorithm is shown below.

VDA algorithm

1. Initialization phase (only performed once)
Deploy N sensors randomly in A

$\quad location(i) = (x_i, y_i)$

$\quad state(i) = \text{'active'}$

S_i collects locations of sensors in $C(S_i)$, and generates $LVD(s_i)$

Calculate w_i

Send the information message to neighbors

2. Decision phase
for receiving message from neighbors
   Update w_i
   if w_i is maximal among neighbors of S_i
     Rotate the orientation to make covering the maximal edge
     Send the information message to neighbors
     if transition timer of S_i is on
       turn the transition timer off
     end if
     continue
   end if
   if $w_i = 0$ and transition timer of S_i is off
     turn the transition timer on
   end if
   if transition timer of S_i remains on for longer than T_r
     $state(i) = \text{'inactive'}$
     return
   end if
end for

Table 1. Parameters setting.

Parameter	Default	Variation
A Area	500*500	500*500
Sensor number N	200	70-1000
Sensing range R_S	50	30-120
Offset angle α	$\pi/3$	$\pi/6$–π

It is clear that the VDA algorithm terminates in finite time when all edges are covered or all sensors make their decisions. By analyzing the pseudo-code, the initialization phase has $O(N\log N)$ complexities, while the time of decision phase is $O(N)$.

5. Simulations

In this section, we evaluate the performance of two algorithms by simulations executed in MATLAB 7.4.0. The simulation parameters are summarized in Table 1.

We compare the performance of VCA and VDA with the coverage-enhancing algorithm which Tao [13] presented and the random algorithm in which every sensor separately selects its orientation randomly. For random algorithm, we run it 100 times and get the average value.

First, we examine the effect of the number of sensors N. As shown in Figure 4, with the increase of sensors deployed, both the ratio of coverage and number of active sensors for all three algorithms increase linearly until N approaches 400; upon passing such a value, the number of active sensors increases slowly or even decreases whereas the ratio of coverage continuously increases and then becomes saturated when N is above 400 or so. To state the difference: for the ratio of coverage, VCA always behaves better than VDA, obviously Tao's and random algorithm; for the number of active sensors,

VCA activates larger number of sensors than VDA, and the gap between two algorithm is nearly unfluctuating till N exceeds a value ($\geqslant 400$ in this simulation). The reasons that incur the above difference, are that 1) VCA and VDA decrease overlapped area in dense network by rotating orientations of directional sensors, so both of them work better than the random algorithm; and 2) VCA knows global information, while in VDA every sensor makes its decision independently based only on local information, so VCA achieves higher ratio of coverage with more sensors whereas VDA activates less sensors with lower ratio of coverage.

Figure 5 and Figure 6 show the influence of the sensing range and the offset angle on our two algorithms respectively. Clearly with the increase of R_s or α, the ratio of coverage increases for VCA, VDA and random algorithm, while the number of active sensors decreases. Also, for the ratio of coverage, VDA works better than VCA, Tao's and random algorithm, and VCA excels VDA in the number of active sensors. The intuitive reasons for results are similar to the effect of number of sensors. Notice that when R_s or α passes such a great value (R_s approaches 120 and α approaches π in simulations), the number of active sensors of VDA is near to that of VCA. This is because the ratio of coverage is saturated till R_s or α approaches a great value, and both of two algorithms need a certain number of sensors.

6. Conclusions and Future Work

Coverage is always an important issue for sensor networks. Different from many other papers designed for isotropic sensor networks, this paper has studied the problem of optimal coverage of directional sensor net
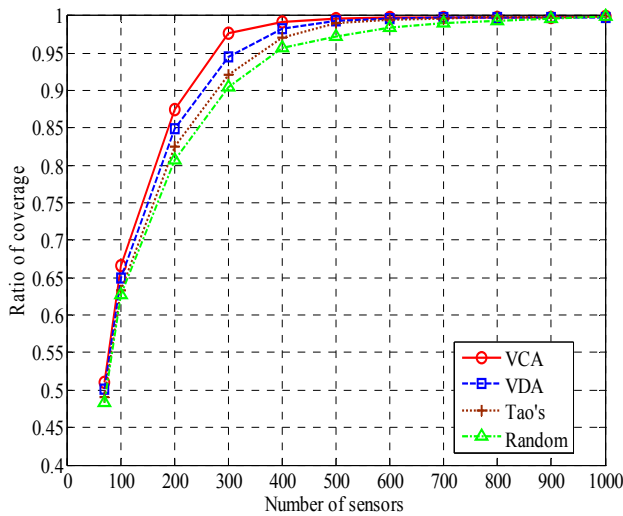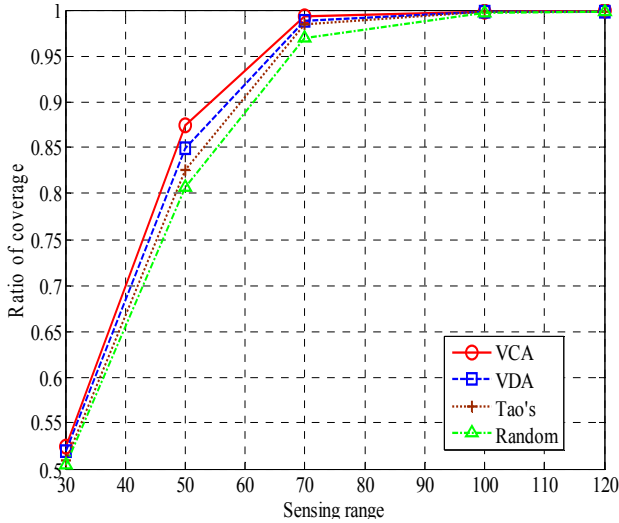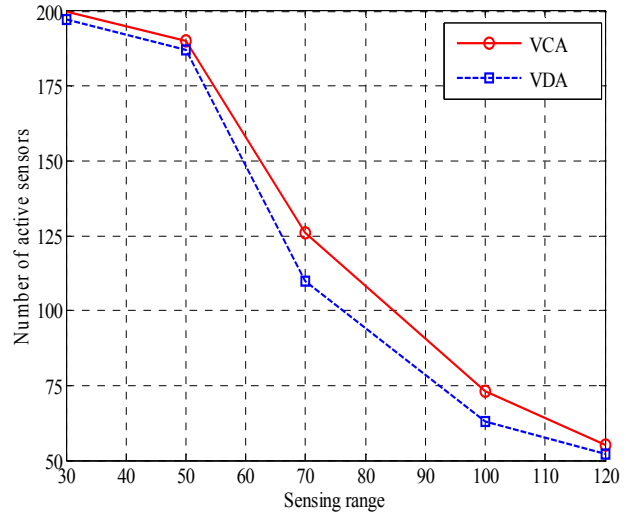

(a) Ratio of coverage vs. number of sensors
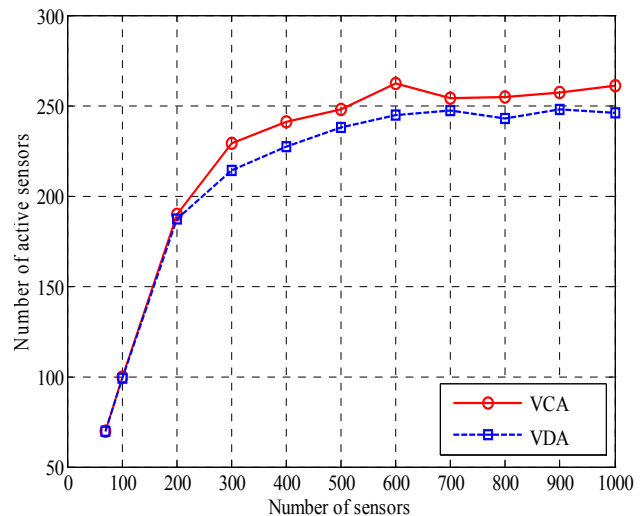
(b) Number of active sensors vs. number of sensors

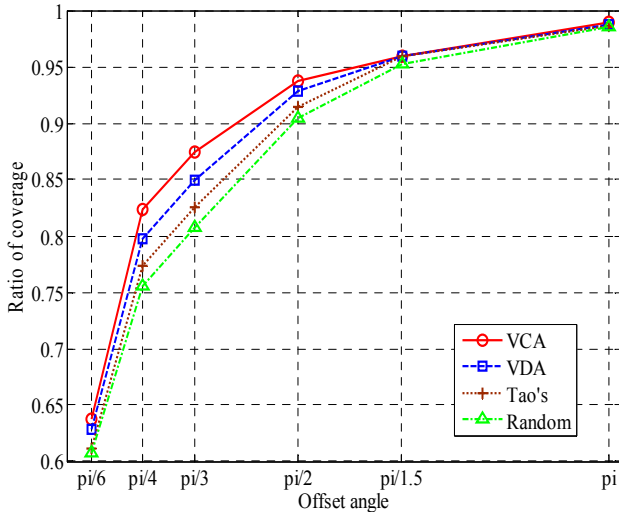**Figure 4. Effect of number of sensors with $R_S = 50, \alpha = \pi/3$.**
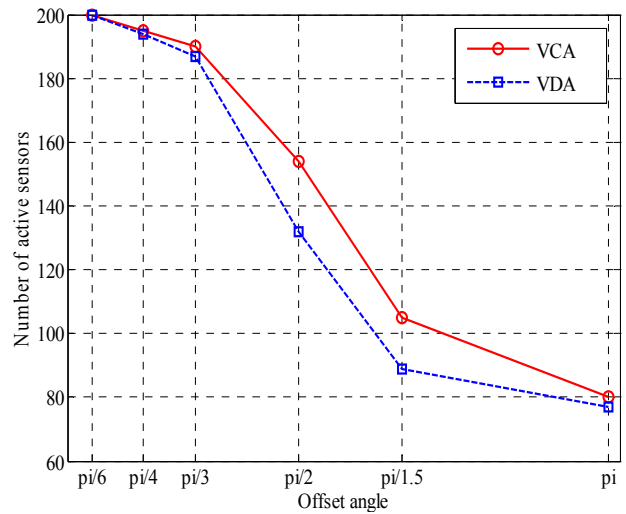
(a) Ratio of coverage vs. sensing range

(b) Number of active sensors vs. sensing range

Figutre 5. Effect of sensing range with $N = 200, \alpha = \pi/3$ **.**

(a) Ratio of coverage vs. offset angle

(b) Number of active sensors vs. offset angle

Figure 6. Effect of offset angle with $N = 200, R_S = 50$ **.**

works and proved this problem is NP-complete. After the definition of sensing model and some assumptions, we present a centralized approximation algorithm and a distributed approximation algorithm to solve the OCDSN problem, based on the boundary Voronoi diagram. As our algorithms finish, we can obtain the optimizing covered area. Finally, we systematically evaluate the performance of two algorithms through extensive simulations. The results show that VCA and VDA work better than the coverage-enhancing algorithm which Tao presented and the random algorithm, and fewer sensors can cover more area so that more sensors can be inactive. Moreover, we analyze the advantage and disadvantage of VCA and VDA respectively. As a future work, we plan

to design energy efficient algorithms to prolong the lifetime of directional sensor networks.

7. Acknowledgments

8. References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," ACM Trans. on Multimedia Computing, Communications and Applications, Vol. 40, No. 8, pp. 102–114, August 2002.

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

[3] M. Li and Y. Liu, "Underground structure monitoring with wireless sensor networks," Information Processing in Sensor Networks, IPSN, 6th International Symposium on, pp. 69–78, April, 2007.

[4] M. Rahimi, R. Baer, O. I. Iroezi, J. C. Garcia, J. Warrior, D. Estrin, and M. Srivastava, "Cyclops: In situ image sensing and interpretation in wireless sensor networks," SenSys, pp. 192–204, 2005.

[5] W. Feng, E. Kaiser, W. C. Feng, and M. L. Baillif, "Panoptes: Scalable low-power video sensor networking technologies," ACM Transactions on Multimedia Computing, Communications and Applications, Vol. 1, No. 2, pp. 151–167, May 2005.

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

[7] M. Marengoni, B. A. Draper, A. Hanson, and R. A. Sitaraman, "A System to Place Observers on a Polyhedral Terrain in Polynomial Time," Image and Vision Computing, Vol. 18, pp. 773–80, December 1996.

[8] A. Ghosh, "Estimating coverage holes and enhancing coverage in mixed sensor networks," Local Computer Networks, 29th Annual IEEE International Conference on, pp. 68–76, November 2004.

[9] S. S. Dhillon, K. Chakrabarty, and S. S. Iyengar, "Sensor placement for grid coverage under imprecise detections," Information Fusion, Proceedings of the Fifth International Conference on, Vol. 2, pp. 1581–1587, July 2002.

[10] S. S. Dhillon and K. Chakrabarty, "Sensor placement for effective coverage and surveillance in distributed sensor networks," in Proceedings of IEEE WCNC, pp. 1609–1614, March 2003.

[11] H. Ma and Y. Liu, "On coverage problems of directional sensor networks," in International Conference on Mobile Ad-hoc and Sensor Networks (MSN), Vol. 3794, pp. 721–731, 2005.

[12] D. Tao, H. Ma, and L. Liu, "A virtual potential field based coverage-enhancing algorithm for directional sensor networks," Journal of Software, Vol. 18, No. 5, pp. 1152−1163, May 2007.

[13] D. Tao, H. Ma, "Coverage-enhancing algorithm for directional sensor networks," Stojmenovic I, Cao JN, eds. Proceedings of the 2nd International Conference on Mobile Ad-Hoc and Sensor Networks, pp. 256−267, 2006.

[14] 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, February 2006.

[15] Y. Cai, W. Lou, and M. Li, "Cover set problem in directional sensor networks," Proceedings of the Future Generation Communication and Networking (FGCN 2007), Vol. 01, pp. 274–278, 2007.

[16] J. Wen, L. Fang, J. Jiang, and W. Dou, "Coverage optimizing and node scheduling in directional wireless sensor networks," Wireless Communications, Networking and Mobile Computing, WiCOM'08, 4th International Conference on, pp. 1–4, October 2008.

[17] S. Megerian, F. Koushanfar, M. Potkonjak, and M. Srivastava, "Worst- and best-case coverage in sensor networks," IEEE Transactions on Mobile Computing, Vol. 4, No. 1, pp. 84–92, February 2005.

[18] J. Adriaens, S. Megerian, and M. Potkonjak, "Optimal worst-case coverage of directional field-of-view sensor networks," Sensor and Ad Hoc Communications and Networks, SECON'06, 3rd Annual IEEE Communications Society on, Vol. 1, pp. 336–345, September 2006.

[19] F. Aurenhammer, "Voronoi diagrams-A survey of a fundamental geometric data structure," ACM Computing Surveys, Vol. 23, No. 3, pp. 345–405, September 1991.