An Efficient Deployment Heuristic to Support Temporal Coverage of Heterogeneous Objects in Rotatable and Directional (R&D) Sensor Networks

You-Chiun Wang and Shin-En Hsu
Department of Computer Science and Engineering,
National Sun Yat-sen University, Kaohsiung 80424, Taiwan
E-mail: ycwang@cse.nsysu.edu.tw, m013040055@student.nsysu.edu.tw

***Abstract*—In rotatable and directional (R&D) sensor networks, each sensor has sector-like sensing range and rotation ability. These sensors can support temporal coverage of objects by periodically rotating to cover them. We define that an object is δ_i-*time covered* if it is covered by a sensor for at least $0 < \delta_i \leq 1$ portion of each period. Given a set of objects where they may have different δ_i-time covered demands, the paper proposes an *R&D sensor deployment for heterogeneous objects (RSD-HO)* problem which determines how to use the minimum number of R&D sensors to make every object be δ_i-time covered. The RSD-HO problem is NP-hard, so we propose an efficient heuristic by considering the distribution and δ_i values of objects. Simulation results show that our RSD-HO heuristic requires fewer sensors compared with existing schemes. The paper contributes in defining an NP-hard RSD-HO problem and developing an efficient heuristic to solve it.**

Keywords*—directional sensor, node deployment, rotation, temporal coverage, wireless sensor network.

I. INTRODUCTION

Wireless sensor networks possess many charming features such as self-organizable networking, collaborative sensing, and distributed computing [1]. They provide a convenient way to persistently monitor physical environments and have been adopted in various real-life applications [2]–[5]. A lot of research on wireless sensor networks assume that sensors have disk-like sensing range. However, certain kinds of sensors, for instance, infrared, camera, sonar, and ultrasonic sensors have sector-like sensing range because of their hardware property. Such sensors are called *directional sensors*.

Some research efforts [6]–[8] suggest using stepper motors to support the rotation ability to directional sensors in order to provide *temporal coverage* of target objects. We call these sensors *rotatable and directional (R&D) sensors* and they can periodically rotate to monitor the objects around them. Unlike traditional coverage problems where target objects or locations have to be *always* monitored by sensors [9], this temporal coverage behavior allows sensors to monitor different objects or locations as time goes on, which gives sensors more flexibility. Besides, it has been shown in [8] that using R&D sensors to monitor objects can significantly reduce the deployment cost by adopting the temporal coverage behavior, comparing with that using static directional sensors.

In this paper, we propose an *R&D sensor deployment for heterogeneous objects (RSD-HO) problem* to model the above temporal coverage behavior in an R&D sensor network. Assume that each R&D sensor can rotate 360 degrees and spend constant time T to monitor objects in a period. We define that an object is δ_i-*time covered* if it can be monitored by one R&D sensor for at least $\delta_i T$ time in every period, where $0 < \delta_i \leq 1$. Given a set of heterogeneous objects where they may have different δ_i-time covered demands, the RSD-HO problem asks how to place the minimum number of R&D sensors to monitor all objects such that their coverage demands can be satisfied.

The RSD-HO problem is NP-hard, and thus we develop an efficient heuristic to solve it. Our idea is to first calculate a set of disks to cover all objects. Then, the RSD-HO heuristic selects a subset of disks according to the positions and δ_i-time covered demands of objects and places R&D sensors accordingly. Finally, it adds some relay nodes to maintain the network connectivity. Simulation results show that our RSD-HO heuristic can significantly reduce the number of sensors compared with existing schemes, which helps save the deployment cost of R&D sensor networks.

We organize this paper as follows: Section II surveys the related work. The RSD-HO problem is defined in Section III and we give our heuristic to this problem in Section IV. Experimental results are presented in Section V. Finally, we conclude the paper in Section VI.

II. RELATED WORK

Sensor deployment is an important research issue in wireless sensor networks since it decides the construction cost and detection capability of the network [10]. For wireless sensor networks where sensors have disk-like sensing range, [11] proposes an optimal solution to deploy sensors in an infinite 2D plane such that the construction cost is minimized. The work of [12] considers how to deploy sensors in a bounded sensing field with obstacles. Both [13], [14] develop hexagon-like deployment strategies to provide k-coverage of a sensing field, where every point can be covered by at least k sensors.

Many studies adopt mobile sensors to automate network deployment. For example, [15], [16] assume that sensors can exert repulsive forces on each other, so that they can be eventually distributed over the sensing field in a uniform manner. A Voronoi diagram is adopted in [17] to search coverage holes, and then sensors are moved to reduce or eliminate these holes.

The work of [18] first calculates the target locations to place with sensors in the sensing field, and then moves sensors to these locations such that they can spend less energy. In [19], [20], [21], mobile sensors are dynamically dispatched to visit event locations pointed out by static sensors. These studies support point coverage (i.e., event locations) to some extent.

Several studies address the deployment problems by static directional sensors. For example, [22] formulates both the *connected region-coverage (CRC) problem* and *connected point-coverage (CPC) problem* that ask how to deploy the minimum number of directional sensors to cover an infinite 2D plane and a set of point-locations, respectively. The CRC problem is solved by deploying sectors in a hexagon-like manner on the plane. To solve the CPC problem, [22] deploys sensors to cover the sectors anchored by one, two, and three point-locations that cover the maximum number of point-locations. On the other hand, [23] models the sensing field by grid points and selects a set of *critical points* that should be covered by directional sensors. Then, an integer linear programming solution is proposed to select the minimum number of grid points to deploy with sensors. However, these studies do not exploit the rotation ability of R&D sensors.

Given a set of objects with the same δ-time covered demand, [8] proposes an *R&D sensor deployment problem*. It determines how to deploy the minimum number of R&D sensors such that every object can be δ-time covered, where $0 < \delta \leq 1$. This problem is proved to be NP-hard and a *maximum covering deployment (MCD) scheme* is proposed by iteratively deploying an R&D sensor to cover the maximum number of objects. Notice that since the R&D sensor deployment problem assumes homogeneous objects, it in fact can be treated as a special case of our RSD-HO problem. Therefore, the RSD-HO problem is also NP-hard because it is more difficult than the R&D sensor deployment problem. We will also compare our proposed RSD-HO heuristic with the MCD scheme by simulations in Section V.

III. PROBLEM DEFINITION

Suppose that there is a set of static objects $\mathcal{O} = \{o_1, o_2, \cdots, o_m\}$ that should be monitored by sensors in the sensing field. Each object is modeled by a point-location for simplification. The sensing range of an R&D sensor is a sector with the angle of $\theta \in (0, \pi)$ and the radius of r_s. In addition, its communication range is a disk whose radius is r_c. We assume the *binary sensing model*, where an object is covered by a sensor if it locates in the sensing range of the sensor. Each sensor can rotate 360 degrees and keep the constant rotation speed. During rotation, a sensor can stop to monitor objects for a while. We assume homogenous sensors so that they have the same θ, r_s, and r_c values and the equal rotation speed. Besides, all sensors rotate in the same direction (e.g., counterclockwise).

The time axis is divided into equal-length *periods*, during which every R&D sensor rotates 360 degrees and spends total time T to monitor the objects that it takes care of. Fig. 1 presents an example, where sensor s_i rotates to cover sectors A, B, and C in a period and it stops to monitor the objects in each sector for $T/3$ time. Similarly, in each period sensor s_j rotates to cover both sectors D and E, each being monitored for $T/2$ time. In this case, we say that the objects in sectors A, B, and C are 1/3-time covered while the objects in sectors D and E are 1/2-time covered.

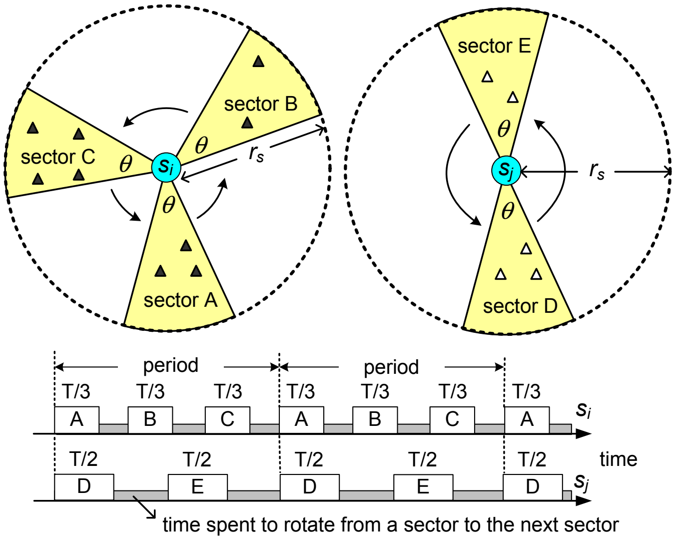


Fig. 1. The temporal coverage of objects supported by R&D sensors, where objects are marked by triangles and sensors rotate countercolockwise.

Given the position and δ_i-time covered demand of each object o_i in $\mathcal{O}$, where $0 < \delta_i \leq 1$, the RSD-HO problem asks how to use the minimum number of R&D sensors and determine their locations in the sensing field such that every object o_i can be monitored by one R&D sensor for at least $\delta_i T$ time in each period.

IV. THE PROPOSED RSD-HO HEURISTIC

Given the set of heterogeneous objects $\mathcal{O}$, our RSD-HO heuristic first deploys R&D sensors to cover all objects such that the δ_i-time covered demand of each object in $\mathcal{O}$ is satisfied. Then, we add some relay nodes to maintain the network connectivity. Specifically speaking, the RSD-HO heuristic contains the following five steps:

In the first step, each object is set to *unchecked*. Then, we find a set of *possible* disks $\mathcal{D}$ to cover all objects in $\mathcal{O}$, where each disk in $\mathcal{D}$ has the radius of r_s. Here, we modify the solution to the *geometric disk cover (GDC) problem* in [24] to find the set $\mathcal{D}$, which is composed of the three rules:

- **R1:** If two objects have a distance shorter than $2r_s$, we place two disks to let their peripheries intersect at the two objects.
- **R2:** If two objects have a distance of $2r_s$, we place a disk to let its periphery pass the two objects.
- **R3:** If the distance between any neighbor of an object and that object is longer than $2r_s$, we place a disk whose center is on the object.

The modified GDC solution will calculate at most

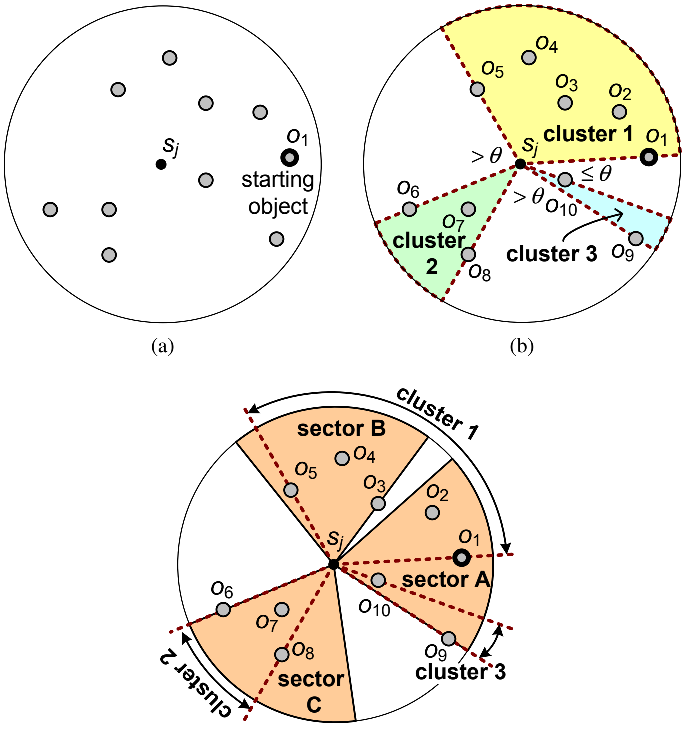
$$2 \cdot C(m, 2) = \frac{2 \cdot m!}{(m-2)! \cdot 2!} = m(m-1) = O(m^2)$$

disks in $\mathcal{D}$, which is larger than the number of objects in $\mathcal{O}$ (i.e., m). Thus, in step 2 we select only a subset of $O(m)$

disks $\mathcal{D}_s$ from $\mathcal{D}$. Our idea is to select each disk such that 1) the disk covers the maximum number of objects and 2) the disk covers more objects with larger δ_i values. Specifically, let δ_i^{sum} denote the sum of δ_i values of all unchecked objects covered by a disk d_i. Then, we select the disk from $\mathcal{D}$ whose δ_i^{sum} value, say, $\delta_{\max}^{sum}$ is the maximum. A disk d_j is called a *candidate disk* if $\delta_{\max}^{sum} - \delta_j^{sum} \leq \varepsilon$, where ε is a predefined threshold. From all candidate disks, we select the disk, say, d_i that covers the maximum number of unchecked objects. We then add d_i to $\mathcal{D}_s$, set all objects covered by d_i as *checked*, and remove d_i from $\mathcal{D}$. The above iteration is repeated until all objects in $\mathcal{O}$ are marked by checked (in other words, we have found a subset $\mathcal{D}_s$ of disks from $\mathcal{D}$ to cover each object).

For each disk in $\mathcal{D}_s$, we cut it into sector(s) with an angle of θ based on the object distribution in the disk. Here, we adopt the *sector cutting operation* [8] in step 3, whose idea is to first group the objects in a disk into cluster(s) and then place sector(s) to cover each cluster. We use Fig. 2 to explain the operation. In particular, an object is randomly picked from the disk and indexed by o_1, as shown in Fig. 2(a). Then, we add o_1 to cluster 1 and scan other non-indexed objects from o_1 in a counterclockwise direction. When finding a non-indexed object, it is indexed by o_2. Then, we decide which cluster o_2 belongs to. Specifically, if $\angle o_1 s_j o_2 \leq \theta$, o_2 belongs to cluster 1; otherwise, o_2 is added to a new cluster 2, where s_j is the disk center. Similarly, suppose that an object is indexed by o_i and it belongs to cluster k. Then, the next non-indexed object is indexed by o_{i+1}. If $\angle o_i s_j o_{i+1} \leq \theta$, o_{i+1} belongs to cluster k; otherwise, it is added to a new cluster $k+1$. Fig. 2(b) gives the clustering result. Since clusters 1 and 3 have the included angle smaller than θ, we start placing sectors from cluster 3, as shown in Fig. 2(c). Thus, three sectors A, B, and C are found to cover all objects in the disk.

In step 4, we place R&D sensors on some disks in $\mathcal{D}_s$ to meet the coverage demand of each object. Our idea is to first deal with those objects with larger δ_i values as they are critical. In particular, we pick the disks that contain the objects with the maximum δ_i value. In this case, each R&D sensor can rotate to cover no more than $\lfloor 1/\delta_i \rfloor$ sectors if it wants to handle these objects. Thus, we place an R&D sensor to cover (at most) $\lfloor 1/\delta_i \rfloor$ sectors in the disk such that it can cover the maximum number of objects (including those objects with the maximum δ_i value). Then, we remove the objects covered by the R&D sensor from $\mathcal{O}$. This iteration is repeated until $\mathcal{O}$ is empty. Fig. 3 gives an example, where there are three types of objects in $\mathcal{O}$ whose δ_i values are 0.7, 0.5, and 0.3 (they are respectively denoted by 'X', 'Y', and 'Z'). In this case, an R&D sensor can rotate to cover at most $\lfloor 1/0.7 \rfloor = 1$, $\lfloor 1/0.5 \rfloor = 2$, and $\lfloor 1/0.3 \rfloor = 3$ sectors containing X, Y, and Z objects, respectively. In iteration 1, both disks d_i and d_j have X objects. However, sector E contains three X objects while sector C contains two objects (with one X object). Thus, we place a sensor on s_j to cover sector E. Similarly, in iteration 2, we place one sensor on s_i to cover sector C. In iteration 3, both disks d_i and d_j have Y objects. However, we place a sensor on s_i to cover sectors A and D since they contain the maximum number of objects (i.e., five objects). In iteration 4, since only disk d_j has Y objects, we place a sensor on s_j to cover both sectors F and G, which contains four objects. Finally, we place a sensor on s_i to cover sector B. Therefore, it totally requires five R&D sensors in the example (three sensors on s_i and two sensors on s_j).



(a) (b)

(c)

Fig. 2. The sector cutting operation: (a) pick one object and index it by o_1, (b) scan all objects counterclockwise and group them into three clusters, and (c) starting from cluster 3, place sectors to cover each cluster.

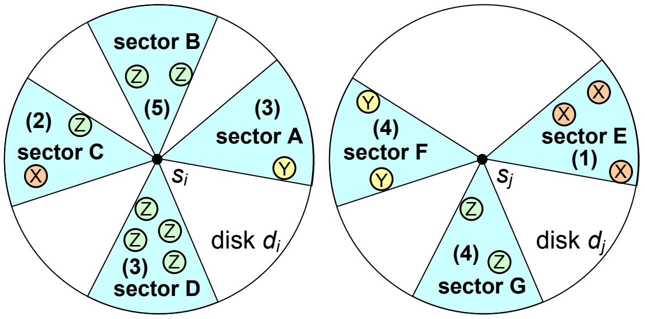


Fig. 3. Place R&D sensors to satisfy the coverage demand of each object in step 4. The number in each sector indicates the sensor deployment sequence.

The above four steps only deploy R&D sensors to cover all objects, but they may not form a connected network. In the last step, we thus add *relay nodes* to keep the network connectivity. The idea is to compute a minimum spanning tree to connect all R&D sensors. Then, for each tree edge whose length, say, l_i is longer than r_c, we add $\lceil l_i/r_c \rceil - 1$ relay nodes on the edge, where any two adjacent relay nodes are separated by a distance of r_c, to connect the two R&D sensors located on the end-points of that edge.

V. EXPERIMENTAL RESULTS

We develop a Java-based simulator to measure the performance of our RSD-HO heuristic. Our simulator considers a 400×400, rectangle-shape sensing field, inside which there are 100 to 400 static objects that should be monitored by

R&D sensors. Objects are arbitrarily placed in the sensing field following the uniform distribution. We assume three types of objects, namely *X, Y, and Z objects*, which have δ_X-time, δ_Y-time, and δ_Z-time covered demands, respectively. Two scenarios are considered in our simulator: 1) $\delta_X = 0.7$, $\delta_Y = 0.5$, and $\delta_Z = 0.25$; 2) $\delta_X = 0.7$, $\delta_Y = 0.3$, and $\delta_Z = 0.25$. Both the sensing and communication distances of an R&D sensor are set to 50. Besides, the sector angle θ is set to 30 degrees and 60 degrees.

We compare our RSD-HO heuristic with the MCD scheme [8] mentioned in Section II, whose objective is to deploy the minimum number of R&D sensors to let every (homogenous) object be δ-time covered. The numbers of R&D sensors and relay nodes are evaluated in both schemes by our simulations. In the RSD-HO heuristic, we set the threshold $\varepsilon = 1$.

Fig. 4 presents the number of nodes (including R&D sensors and relay nodes) deployed by our RSD-HO heuristic and the MCD scheme. Obviously, the RSD-HO heuristic significantly saves the number of required R&D sensors compared with the MCD scheme. The reason is that the MCD scheme is designed for homogeneous objects where they have the same δ-time covered demand. When objects have different demands for coverage, the MCD scheme would not perform well. On the contrary, by taking the object heterogeneity into consideration, our RSD-HO heuristic can adaptively calculate the proper positions to deploy R&D sensors according to the locations and δ_i values of objects in the sensing field.

By comparing the four simulation subfigures in Fig. 4, we have the following three observations:

- It is apparent that the RSD-HO heuristic and MCD scheme have to deploy more R&D sensors when the number of objects grows in the sensing field. Thus, we will also require (slightly) more relay nodes to connect all R&D sensors in both deployment methods.
- A larger sector angle θ helps reduce the number of R&D sensors used in the RSD-HO heuristic and MCD scheme. The reason is that each disk can be covered by at most $\lceil 2\pi/\theta \rceil$ sectors. Obviously, when θ increases, both deployment methods require fewer sectors (and thus fewer R&D sensors) to monitor the objects in every disk. On the other hand, because both deployment methods select disks to deploy with R&D sensors, it would have insignificant effect on the number of relay nodes by changing the θ angle.
- Our RSD-HO heuristic can save more R&D sensors in scenario 2. The reason is that an R&D sensor can cover more sectors with Y objects but without X objects in this scenario. In particular, each R&D sensor can rotate to cover at most $\lfloor 1/0.5 \rfloor = 2$ and $\lfloor 1/0.3 \rfloor = 3$ sectors containing Y objects (but without X objects) of a disk in scenarios 1 and 2, respectively. On the other hand, changing the δ_Y value in difference scenarios will only affect the number of R&D sensors deployed in a disk. In this case, it has no effect on the number of relay nodes.

By calculating the average number of R&D sensors required by our RSD-HO heuristic and the MCD scheme in each subfigure of Fig. 4, we can summarize the *sensor reduction ratio* of the RSD-HO heuristic to the MCD scheme in Table I, which is defined by:

TABLE I. AVERAGE SENSOR REDUCTION RATIO OF THE RSD-HO HEURISTIC TO THE MCD SCHEME IN THE EXPERIMENTS.

	scenario 1		scenario 2	
θ	30 degrees	60 degrees	30 degrees	60 degrees
ratio	20.73%	15.37%	26.25%	18.24%

$$\frac{(\text{\# of sensors by MCD}) - (\text{\# of sensors by RSD-HO})}{\text{\# of sensors by MCD}} \times 100\%.$$

In sum, our RSD-HO heuristic performs better with a smaller θ angle and in scenario 2 according to the simulation results.

VI. CONCLUSION

To model the temporal coverage behavior of R&D sensors, this paper formulates an NP-hard RSD-HO problem which determines how to deploy the minimum number of R&D sensors to make each object in the sensing field be δ_i-time covered. A five-step heuristic is proposed by first calculating a set of disks to cover all objects and then placing R&D sensors on some of these disks in order to meet the coverage demand of each object. Simulation results verify the effectiveness of our RSD-HO heuristic in terms of the network deployment cost, where it can save approximately 23.49% and 16.81% of R&D sensors compared with the MCD scheme in average when the sector angle θ is 30 and 60 degrees, respectively.

ACKNOWLEDGEMENTS

You-Chiun Wang's research is co-sponsored by the Ministry of Science and Technology under grant no. 100-2218-E-110-006-MY3, Taiwan.

REFERENCES

[1] Y. C. Wang, F. J. Wu, and Y. C. Tseng, "Mobility management algorithms and applications for mobile sensor networks," *Wireless Communications and Mobile Computing*, vol. 12, no. 1, pp. 7–21, 2012.

[2] Y. C. Tseng, Y. C. Wang, K. Y. Cheng, and Y. Y. Hsieh, "iMouse: an integrated mobile surveillance and wireless sensor system," *IEEE Computer*, vol. 40, no. 6, pp. 60–66, 2007.

[3] L. W. Yeh, Y. C. Wang, and Y. C. Tseng, "iPower: an energy conservation system for intelligent buildings by wireless sensor networks," *International Journal of Sensor Networks*, vol. 5, no. 1, pp. 1–10, 2009.

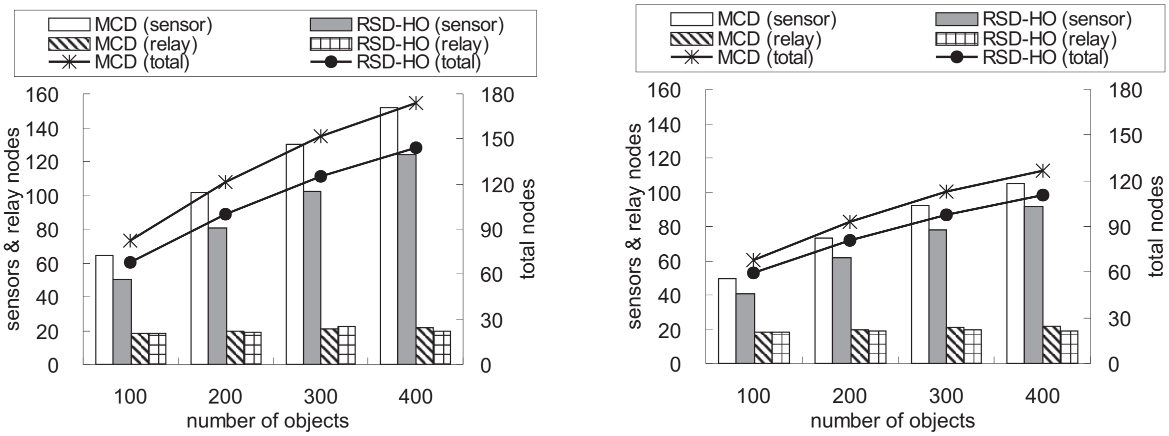
[4] B. M. Howe, Y. Chao, P. Arabshahi, S. Roy, T. McGinnis, and A. Gray, "A smart sensor web for ocean observation: fixed and mobile platforms, integrated acoustics, satellites and predictive modeling," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 3, no. 4, pp. 507–521, 2010.

[5] S. C. Hu, Y. C. Wang, C. Y. Huang, and Y. C. Tseng, "Measuring air quality in city areas by vehicular wireless sensor networks," *Journal of Systems and Software*, vol. 84, no. 11, pp. 2005–2012, 2011.

[6] S. Aoki, J. Nakazawa, and H. Tokuda, "Spinning sensors: a middleware for robotic sensor nodes with spatiotemporal models," *Proc. IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, pp. 89–98, 2008.

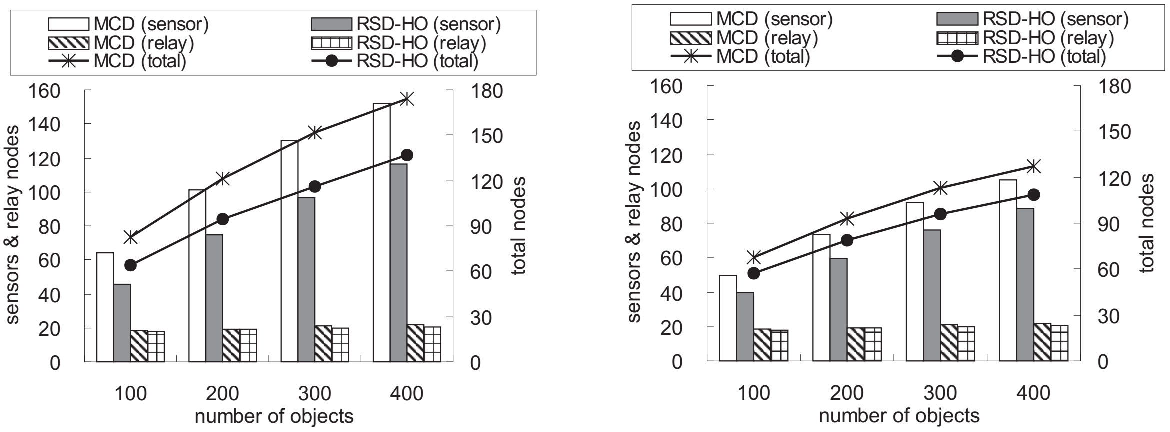
[7] G. Lee and N. Y. Chong, "Low-cost dual rotating infrared sensor for mobile robot swarm applications," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 2, pp. 277–286, 2011.

[8] Y. C. Wang, Y. F. Chen, and Y. C. Tseng, "Using rotatable and directional (R&D) sensors to achieve temporal coverage of objects and its surveillance application," *IEEE Transactions on Mobile Computing*, vol. 11, pp. 1358–1371, 2012.



(a) Scenario 1, $\theta = 30$ degrees

(b) Scenario 1, $\theta = 60$ degrees



(c) Scenario 2, $\theta = 30$ degrees

(d) Scenario 2, $\theta = 60$ degrees

Fig. 4. The experimental results.

[9] B. Wang, “Coverage problems in sensor networks: a survey,” *ACM Computing Surveys*, vol. 43, no. 4, pp. 32:1–32:53, 2011.

[10] Y. C. Wang, K. Y. Cheng, and Y. C. Tseng, “Using event detection latency to evaluate the coverage of a wireless sensor network,” *Computer Communications*, vol. 30, no. 14-15, pp. 2699–2707, 2007.

[11] X. Bai, S. Kumar, D. Xuan, Z. Yun, and T. H. Lai, “Deploying wireless sensors to achieve both coverage and connectivity,” *Proc. ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 131–142, 2006.

[12] Y. C. Wang, C. C. Hu, and Y. C. Tseng, “Efficient deployment algorithms for ensuring coverage and connectivity of wireless sensor networks,” *Proc. IEEE Wireless Internet Conference*, pp. 114–121, 2005.

[13] T. Sun, L. J. Chen, C. C. Han, and M. Gerla, “Reliable sensor networks for planet exploration,” *Proc. IEEE International Conference on Networking, Sensing and Control*, pp. 816–821, 2005.

[14] Y. C. Wang and Y. C. Tseng, “Distributed deployment schemes for mobile wireless sensor networks to ensure multilevel coverage,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 9, pp. 1280–1294, 2008.

[15] N. Heo and P. K. Varshney, “Energy-efficient deployment of intelligent mobile sensor networks,” *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 35, no. 1, pp. 78–92, 2005.

[16] H. Liu, X. Chu, Y. W. Leung, and R. Du, “Simple movement control algorithm for bi-connectivity in robotic sensor networks,” *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 7, pp. 994–1005, 2010.

[17] G. Wang, G. Cao, and T. F. La Porta, “Movement-assisted sensor deployment,” *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 640–652, 2006.

[18] Y. C. Wang, C. C. Hu, and Y. C. Tseng, “Efficient placement and dispatch of sensors in a wireless sensor network,” *IEEE Transactions on Mobile Computing*, vol. 7, no. 2, pp. 262–274, 2008.

[19] A. Verma, H. Sawant, and J. Tan, “Selection and navigation of mobile sensor nodes using a sensor network,” *Pervasive and Mobile Computing*, vol. 2, no. 1, pp. 65–84, 2006.

[20] Y. C. Wang, W. C. Peng, and Y. C. Tseng, “Energy-balanced dispatch of mobile sensors in a hybrid wireless sensor network,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 12, pp. 1836–1850, 2010.

[21] Y. C. Wang, “A two-phase dispatch heuristic to schedule the movement of multi-attribute mobile sensors in a hybrid wireless sensor network,” *IEEE Transactions on Mobile Computing*, vol. 13, no. 4, pp. 709–722, 2014.

[22] X. Han, X. Cao, E. L. Lloyd, and C. C. Shen, “Deploying directional sensor networks with guaranteed connectivity and coverage,” *Proc. IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pp. 153–160, 2008.

[23] Y. E. Osais, M. St-Hilaire, and F. R. Yu, “Directional sensor placement with optimal sensing range, field of view and orientation,” *Mobile Networks and Applications*, vol. 15, pp. 216–225, 2010.

[24] G. K. Das, R. Fraser, A. Lopez-Ortiz, and B. G. Nickerson, “On the discrete unit disk cover problem,” *Lecture Notes in Computer Science*, vol. 6552, pp. 146–157, 2011.