Proceedings of the 33rd Chinese Control Conference
July 28-30, 2014, Nanjing, China

Maximum Network Lifetime of Directional Sensor Networks with Dynamic Coverage Constraints

GU Zhiyue, WANG Yiyin, YANG Bo, CHEN Cailian, GUAN Xinping
Department of Automation, Shanghai Jiao Tong University,
and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai 200240
E-mail: {guzhiyue, yiyinwang, bo.yang, cailianchen, xpguan}@sjtu.edu.cn

Abstract: Directional sensors, such as cameras, are limited in their sensing angles, but are able to switch to different directions and cover the area of interest cooperatively. In directional sensor networks, how to prolong the network lifetime while satisfying certain coverage requirements is a fundamental problem since the sensors are powered by battery. In this paper, we propose scheduling schemes for directional sensor networks to maximize the network lifetime and satisfy the coverage requirements, which are designed to cover the point of interest as well as a specified ratio of area. Since the position of the point of interest may change time to time, dynamic coverage constraints are considered in this paper. Centralized and distributed scheduling algorithms are developed, respectively. Simulation results show that both algorithms can prolong the network lifetime significantly by balancing the energy consumption of the whole network.

Key Words: Directional Sensor Network, Network Lifetime, Point of Interest, Area Coverage

1 Introduction

Nowadays, directional sensors such as video sensors and cameras [1] have been applied for various applications, such as indoor surveillance, scene surveillance, and traffic surveillance. Embedded platforms are flexible and convenient for these complex environments so that directional sensors are usually deployed in embedded sensor networks. Different from the omni-directional sensors, directional sensors are limited in sensing angles, but they are able to switch to different directions. Coverage of directional sensor networks becomes more challenging than omni-directional ones and it depends on practical applications.

Recently, coverage of directional sensor networks has attracted increasing interests from fields of computer vision and sensor networks [2]. As directional sensors are limited in sensing angles, full area coverage is hardly achieved. In most directional sensor networks, only partial area coverage can be achieved, thus how to design sensor scheduling schemes to enlarge the coverage area is critical. In [3], D-Greedy algorithm is proposed to maximize the area coverage. In [4], a greedy algorithm is proposed under k-coverage condition. These works show that since the directional sensors are rotatable, coverage requirements can be achieved by effective sensor scheduling schemes. Target coverage is another fundamental basic issue for coverage problems. Directional Cover Set (DCS) is studied in [5] to cover all the targets. Ref. [6] solves the problem of maximizing the number of covered targets with minimized number of sensors. Nevertheless, few articles consider both area coverage and target coverage. In practical scenarios such as factory or building surveillance, it is required that the specific target, called point of interest (POI), has to be covered, as well as the specific coverage ratio for the whole area has to be achieved. Furthermore, as the position of POI may change time to time, sensor scheduling schemes for this dynamic scenario have to be adaptive by on-line tuning. It is quite different from traditional off-line schemes for static scenarios.

In embedded directional sensor networks, the energy powered by battery is limited for sensors, thus prolonging network lifetime while satisfying the coverage requirements become another important issue [7] [8]. In [9], the maximum lifetime effective-sensing partial target-coverage (ML-EPT) problem is considered to maximize lifetime and meet the target coverage requirement. In [10] [11], Multiple Directional Cover Sets (MDCS) are applied to organize sensors into different sets. Network lifetime can then be prolonged through properly allocating running hours for different sensor sets. Refs. [12] and [13] consider the connectivity as well as the lifetime, and the coverage information has to be delivered to the sink through sensors. Most of the aforementioned works focus on network lifetime prolonging with the coverage over all static targets or a certain area. In these static scenarios, scheduling schemes only need to allocate sensor duty cycles off-line. However, as far as the dynamic POI is considered, the appearance of POI is not predictable. Therefore, scheduling schemes have to activate sensors on-line as well as balance the energy consumption of the network.

In this paper, our goal is to maximize network lifetime in the dynamic scenario. We formulate a Constrained Regional Coverage (CRC) problem to maximize the network lifetime while satisfying the constrained coverage requirements for both POI coverage and the specified ratio of area coverage. Network lifetime is re-defined to take these two requirements into consideration. The requirement of coverage ratio in CRC problem is described as a Maximal Regional Coverage (MRC) subproblem. The solution to this subproblem aims to schedule sensors effectively so that the coverage ratio requirement is achieved efficiently. Priority is considered in the proposed Saving Factor (SF) algorithm as a criterion of sensor scheduling to solve MRC problem. Moreover, the energy consumption balance is considered as another criterion in the proposed Extended Saving Factor (ESF) algorithm and Maximal Neighbor (MN) algorithm to solve CRC

This work was partially supported by Ministry of Science and Technology of China under National Basic Research Project 2010CB731803, by the NSF of China under the grants 61221003, 61290322, 61174127, 61273181, 61301223, and by Science and Technology Commission of Shanghai Municipal, China under the grants 13ZR1421800 and 13QA1401900.

problem. Both algorithms share the priority factors closely related to those factors designed in SF algorithm for sensor scheduling. These two algorithms can be applied to different applications. Simulation results show that both algorithms can prolong the network lifetime significantly.

The rest of the paper is organized as follows: In Section II, CRC problem is described in details. In Section III, SF algorithm is proposed to solve MRC problem. In Section IV, ESF algorithm and MN algorithm are proposed to solve CRC problem. Simulation results and algorithm evaluations are shown in Section V. In Section VI, we conclude the paper.

2 Problem Formulation

In this section, the system model is defined and CRC problem is described in details. We assume an area of size A, where N sensors are deployed in fixed positions. The whole sensing area of a sensor is a disk with the radius R and centered at the sensor. The sensing disk is divided into W non-overlapping sectors called sensing directions. Each time only a single sensing direction can be activated by each sensor. This sensor model is similar to that in [14] and every sensor shares the same model. Furthermore, the position of POI changes randomly in the area. The notations employed in this paper are defined in Table 1 as follows:

Table 1: Notations

A	The size of the specific area.
N	The number of available sensors.
W	The number of directions per sensor.
s_i	The ith sensor, $1 \leq i \leq N$.
$d_{i,j}$	The jth direction of s_i, $1 \leq i \leq N$, $1 \leq j \leq W$.
S	The set of sensors. $S = \{s_1, s_2, \cdots, s_N\}$
D	The set of all the directions. $D = \{d_{i,j} \mid 1 \leq i \leq N, 1 \leq j \leq W\}$

2.1 Cover Conditions and Related Definitions

Since at any moment a sensor can activate no more than one direction, the cover conditions are derived as follows. POI at the position (x, y) can be covered by $d_{i,j}$ if the following conditions (1) and (2) are fulfilled.

$$(x - x_i)^2 + (y - y_i)^2 < R^2 \tag{1}$$

$$\boldsymbol{v_{(i,j)}} \cdot \boldsymbol{u} > R \cdot ||\boldsymbol{u}|| \cdot \cos\frac{\pi}{W} \tag{2}$$

where the sensing direction vector $\boldsymbol{v_{(i,j)}}$ denotes the directional edge from the ith sensor location (x_i, y_i) to the surface center of its jth direction and the vector $\boldsymbol{u}$ is the directional edge from the ith sensor location to POI. The condition (1) indicates that POI is in the sensing range of the ith sensor, while the condition (2) implies that POI is within the jth direction of the ith sensor. (A concrete example is shown in Fig. 1, where POI is covered by the jth direction of the ith sensor.)

To facilitate the problem formulation, we clarify several definitions here.

Definition 1. *(Unique Block, UB) : An unique block is an area covered by one and only one set of sensing directions.*

The whole area of size A is divided into M unique blocks. The ith unique block is denoted by b_i. The set of unique blocks is denoted by B ($B = \{b_1, \ldots, b_M\}$). Fig. 2 illustrates an example of UBs. The area of interest is a square of size $R \times R$, where four sensors with four possible directions per sensor are deployed at the corner of the square. The sensing range of the directional sensor is R. Therefore, the area is divided into nine unique blocks by these sensing directions as shown in Fig. 2.

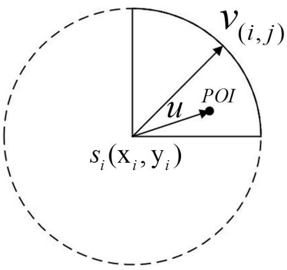

Fig. 1: cover conditions

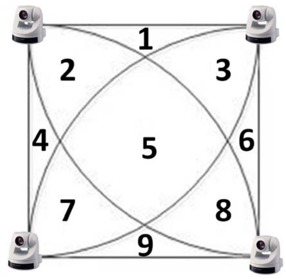

Fig. 2: scenario model

Definition 2. *(η-Coverage Set, D_c) : A η-coverage set is a subset of D satisfying the coverage ratio requirement η, $0 \leq \eta \leq 1$. Any two directions of a sensor cannot belong to the same η-coverage set.*

Not only the coverage ratio is taken into account, but also the coverage over POI is considered in this work. For the dynamic position of POI changes time to time, the network lifetime is defined as follows:

Definition 3. *(Network Lifetime, L_N) : A period of time in which POI is covered and the coverage ratio is satisfied. The network exhausts if any one of the two requirements is not satisfied.*

2.2 Problem Formulation

In this subsection, we give the problem formulation to be solved in this paper.

Problem. *(CRC Problem) : Given N directional sensors deployed in an area A, assume that each sensor initially has the same energy, CRC problem is to schedule the directional sensors so that the network lifetime defined in this paper is maximized.*

Subproblem 1. *(MRC Subproblem) : Given N directional sensors deployed in an area of size A, MRC subproblem is to find a set of sensing directions that maximizes the area coverage.*

Subproblem 2. *(η-MRC Subproblem) : Given N directional sensors deployed in an area of size A, η-MRC subproblem is to find the η-coverage set with highest priority.*

3 Saving Factor Algorithm for MRC

We propose a heuristic algorithm called Saving Factor (SF) algorithm to solve MRC problem in this section. UBs are determined when the sensors are deployed.

The main idea of SF algorithm is to schedule sensors according to priority factors. In SF algorithm, two lists, RankUB and RankD, are created. RankUB is a list to record

UB information including the sizes of UBs and the list of sensing directions cover each UB. RankUB is sorted by the sizes of UBs in the decreased order.

Definition 4. *(Effective Factor, $\gamma_{i,j}$) : Effective Factor is defined as the size of a specific area, which could be covered by $d_{i,j}$, but is not covered by any other activated directions.*

Definition 5. *(Potential Factor, φ_i) : Potential Factor is defined as the size of a specific area, which could be covered by one of the directions of s_i, but is not covered by any other activated directions. All directions of s_i share the same φ_i, and we have*

$$\varphi_i = \sum_{1 \le j \le W} \gamma_{i,j}. \tag{3}$$

Both $\gamma_{i,j}$ and φ_i change through the scheduling process. RankD is a list to record sensing direction information including $\gamma_{i,j}$, φ_i and the list of UBs covered by $d_{i,j}$. RankD is sorted by Saving Factor ($\lambda_{i,j}$) in the decreased order.

$$\lambda_{i,j} = \frac{\gamma_{i,j}}{\varphi_i} \tag{4}$$

The utility of the direction is shown in $\gamma_{i,j}$. The cost of the direction is shown in φ_i. The potential factor is similar to the opportunity cost. When $d_{i,j}$ is activated, the other directions of s_i can never be activated. Therefore, the effective factor of directions of s_i is the cost of activating $d_{i,j}$.

When RankUB and RankD are created, SF algorithm works in three steps.

Step 1 *Sensor Selecting*

According to the first block b_{top} in RankUB, if λ_{i^*,j^*} of d_{i^*,j^*} is the largest one among those of all directions that cover b_{top}, then d_{i^*,j^*} is activated. If no available directions in RankD covers b_{top}, the block b_{top} is removed from RankUB and the algorithm jumps to Step 3.

Step 2 *List Updating*

In Step 2, two lists are updated according to b_{top} and d_{i^*,j^*}. First, b_{top} is removed from RankUB and the weight of b_{top} (i.e the size of b_{top}) is added into the coverage counter A_C. The block b_{top} is also added into a temporary set B_{temp}. Second, d_{i^*,j^*} is added into the set of activated directions D'. Activating d_{i^*,j^*} means that all UBs it covers will be covered. All these covered UBs are removed from RankUB and their weights are added to A_C. These UBs will also be added into B_{temp}. Third, as d_{i^*,j^*} has been activated , the other directions of s_{i^*} are no longer available. These directions are removed from RankD. Finally, the blocks in B_{temp} have been covered in this round. Therefore, according to B_{temp}, $\gamma_{i,j}$ and φ_i in RankD are updated.

Step 3 *Algorithm Terminating*

If RankUB or RankD is empty after Step 2, the algorithm terminates. A_C is the maximal coverage. D' is the set of activated sensors. If none of the two lists is empty, the algorithm continues.

Algorithm 1 is the pseudo-code of SF algorithm.

Algorithm 1 SF algorithm $(RankUB, RankD)$

1: A_C = 0, D' = $\emptyset$
2: **while** $RankUB \neq \emptyset$ & $RankD \neq \emptyset$ **do**
3: Select b_{top} from $RankUB$, B_{temp} = $\emptyset$
4: **if** $\mid \{d_{i,j} \mid d_{i,j} \in RankD$ & $d_{i,j}$ covers b_{top} $\} \mid = 0$ **then**
5: $RankUB$=$RankUB$ - $\{b_{top}\}$
6: Go to Line 26
7: **end if**
8: **for** each $d_{i,j} \in \{d_{i,j} \mid d_{i,j} \in RankD$ & $d_{i,j}$ covers $b_{top}\}$ **do**
9: Activate d_{i^*,j^*} with the largest λ_{i^*,j^*}
10: **end for**
11: $RankUB$=$RankUB$ - $\{b_{top}\}$
12: A_C = $A_C + size(b_{top})$
13: B_{temp} = $B_{temp} \cup \{ b_{top} \}$
14: D' = $D' \cup \{ d_{i^*,j^*} \}$
15: **for** each $b_i \in \{b_i \mid b_i \in RankUB$ & d_{i^*,j^*} covers $b_i\}$ **do**
16: A_C = $A_C + size(b_i)$
17: B_{temp} = $B_{temp} \cup \{ b_i \}$
18: $RankUB$ = $RankUB - \{b_i\}$
19: **end for**
20: $RankD$ = $RankD - \{d_{i^*,j} \mid 1 \le j \le W\}$
21: **for** each $b_i \in B_{temp}$ **do**
22: **for** each $d_{i,j}$ covers b_i **do**
23: Updating $\gamma_{i,j}$ and φ_i of these directions
24: **end for**
25: **end for**
26: **end while**
27: **return** A_C, D'

4 Solutions of CRC

To solve CRC problem, directional sensors are scheduled to maximize the network lifetime with the satisfaction of the coverage ratio and the coverage over POI. The exhausted sensors cause blind areas and decrease the possibility of covering POI later. Since the appearance of POI is not predictable, the energy consumption of the network should be balanced to keep as many sensors alive as possible. In this section, two algorithms are proposed to solve CRC problem, these algorithms share the priority factors for sensor scheduling. These factors are closely related to those designed in SF algorithm. The first algorithm is an extension of SF algorithm, and a centralized one. In the second distributed algorithm called Maximal Neighbor (MN) algorithm, sensors decide sensing direction activation by exchanging information locally. $P(t)$ denotes the position of POI at time t.

4.1 Extended Saving Factor Algorithm for CRC

The coverage ratio needs to be fulfilled in CRC problem. Therefore, η-MRC subproblem needs to be solved, thus the coverage ratio η is added into SF algorithm to find a η-coverage set of the area.

The energy left in s_i at time t is denoted by $E_i(t)$. Therefore, Extended Saving Factor ($\lambda^E_{i,j}(t)$) is defined as

$$\lambda^E_{i,j}(t) = \frac{\gamma_{i,j} E_i(t)}{\varphi_i} \tag{5}$$

A priority factor called Link Factor ($\delta_i(t)$) is used to sort UBs. Eq. 6 shows that the priority of UB is related not only to its size but also the energy left in the sensors that cover it. ($b_i \subset d_{m,n}$ means that $d_{m,n}$ covers b_i).

$$\delta_i(t) = size(b_i) \sum_{b_i \subset d_{m,n}} E_m(t) \tag{6}$$

In the previous section, RankUB is sorted simply by the sizes of UBs. Static RankUB does not take the energy consumption into account so that some sensors may exhaust quickly and then the network lifetime is shortened. Therefore, RankUB is re-sorted by Link Factor ($\delta_i(t)$) in the decreased order in ESF algorithm. Through this process, energy consumption is balanced among the sensors.

ESF algorithm works in this way: When the scheduling process starts, RankUB and RankD are updated through the current network state. The network checks the existence of POI. If POI does not exist, SF algorithm is executed with η which means that SF algorithm terminates when coverage ratio η is achieved. If a η-coverage set is found, the network keeps alive, otherwise it exhausts. If POI exists, first, the direction $d_{i',j'}$ to cover POI will be determined according to the cover conditions and $\lambda^E_{i',j'}(t)$. If no available sensor covers POI, the network exhausts, otherwise ESF algorithm updates RankUB and RankD according to $d_{i',j'}$. Since $d_{i',j'}$ has covered part of the area, the coverage ratio requirement for the other sensors is then adjusted to η'. SF algorithm is then executed with η' among these sensors. If a η'-coverage set is found, the network keeps alive, otherwise it exhausts.

Algorithm 2 shows the pseudo-code of ESF algorithm.

Algorithm 2 ESF algorithm $(B, D, P(t), A, \eta)$

```
1:  L_N = 0
2:  while the network is alive do
3:      Update RankUB, RankD according to the priority factors
4:      if POI does not exist then
5:          Execute SF algorithm with η
6:          if η-coverage set is not found then
7:              return L_N
8:          else
9:              L_N increases, go to Line 25
10:         end if
11:     else
12:         Determine d_{i',j'} with largest λ^E_{i',j'}(t) to cover POI
13:         if d_{i',j'} cannot be found then
14:             return L_N
15:         end if
16:         η' = η − γ_{i',j'}/A
17:         Update RankUB and RankD according to d_{i',j'}
18:         Execute SF algorithm with η' among other sensors
19:         if η'-coverage set is not found then
20:             return L_N
21:         else
22:             L_N increases
23:         end if
24:     end if
25: end while
```

4.2 Maximal Neighbor Algorithm for CRC

MN algorithm is proposed in this section. In this distributed algorithm, directions of sensors are activated by exchanging information locally. Priority factors are also considered for sensor scheduling.

Definition 6. *(Neighbor) : Two sensors are neighbors if and only if they share at least one UB in some directions. We assume that two sensors can communicate with each other directly if and only if they are neighbors. Any state change of one sensor only affects its neighbors.*

Definition 7. *(Instructor) : The instructor delivers the POI information to the available sensors and collects the direction activation information to check the state of the network.*

Algorithm 3 MN algorithm $(B, D, P(t), A, \eta)$

```
1:  L_N = 0 A_C = 0
2:  while the network is alive do
3:      if POI exists then
4:          Instructor delivers P(t) to available sensors
5:          Determine d_{i',j'} with largest λ^E_{i',j'}(t) to cover POI
6:          if d_{i',j'} cannot be found then
7:              return L_N
8:          end if
9:          Neighbors of s_{i'} adjust λ^E_{max(i)}(t)
            Instructor adds increased coverage to A_C
10:     end if
11:     for each available s_i do
12:         Calculate λ^E_{max(i)}(t) and broadcast to its neighbors
13:         if λ^E_{max(i*)}(t) in s_{i*} is the largest one of the neighbors
            then
14:             Activate d_{i*,j*} corresponding to λ^E_{max(i*)}(t)
15:             Neighbors of s_{i*} adjust λ^E_{max(i)}(t)
                Instructor adds increased coverage to A_C
16:         end if
17:     end for
18:     if A_C > ηA then
19:         L_N increases, Go to Line 25
20:     end if
21:     if A_C < ηA & A_C is not updated then
22:         return L_N
23:     end if
24:     Go to Line 11
25: end while
```

When the scheduling process starts, the instructor delivers the POI information to available sensors. If POI exists, sensors that can cover POI calculate $\lambda^E_{i',j'}(t)$ corresponding to the directions cover POI and broadcast it to the neighbors. The sensor with largest $\lambda^E_{i',j'}(t)$ activates the corresponding direction $d_{i',j'}$ and broadcasts this information to its neighbors and the instructor. The neighbors adjust direction states according to this information. The coverage counter A_C increases. If the instructor does not receive the POI coverage information in a period of time, the networks exhausts.

The largest Extended Saving Factor among the directions of s_i is called Maximal Extended Saving Factor ($\lambda^E_{max(i)}(t)$). After covering POI, MN algorithm continues in several scheduling rounds until the coverage ratio is fulfilled or the network exhausts. At the beginning of each scheduling round, the instructor delivers a command to available sensors. Available sensors calculate $\lambda^E_{max(i)}(t)$ and broadcast it to the neighbors. The sensor with largest $\lambda^E_{max(i^*)}(t)$ among its neighbors activates the corresponding direction d_{i^*,j^*} and broadcasts this information to its neighbors and the instructor. The neighbors adjust their $\lambda^E_{max(i)}(t)$ according to these information and wait for the next command. The coverage counter A_C increases. If the coverage ratio η is fulfilled, the scheduling process finishes. If the instructor does not receive updating information in one scheduling round and the coverage ratio is not fulfilled, the network exhausts.

Algorithm 3 shows the pseudo-code of MN algorithm .

5 Simulation Results

Our algorithms for CRC problem and MRC subproblem are evaluated in this section. The configuration of the network is : In an area of $4R \times 9R$, $R = 100$, 50 sensors are uniformly deployed on the grids. The default sensing range of a sensor with 4 directions is 100. We also compare our algorithms to DGreedy algorithm proposed in [3].

5.1 Simulation Results of MRC

Fig. 3 shows the relationship between the maximal coverage and the number of sensors. The maximal coverage is proportional to the number of sensors. SF algorithrm performs better than DGreedy algorithm. When the number of sensors is greater than 30, SF algorithm outperforms DGreedy algorithm by 10%.

Fig. 4 shows the relationship between the maximal coverage and the number of directions per sensor. The more directions one sensor is divided into, the smaller sensing area each direction has. Therefore, the maximal coverage is inversely proportional to the number of directions per sensor. The maximal coverage ratio in SF algorithm falls slower than that in DGreedy algorithm.

Fig. 5 shows the relationship between the maximal coverage and the sensing range. When the sensing range increases, the maximal coverage in SF algorithm increases faster than that in DGreedy algorithm. SF algorithm achieves the full area coverage when the sensing range is 125.

Simulation results show that directional sensors are effectively scheduled through the well-designed priority factors in our algorithm so that the coverage ratio requirement is achieved efficiently.

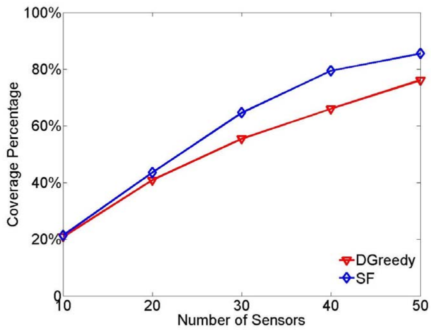

Fig. 3: maximal coverage vs. number of sensors

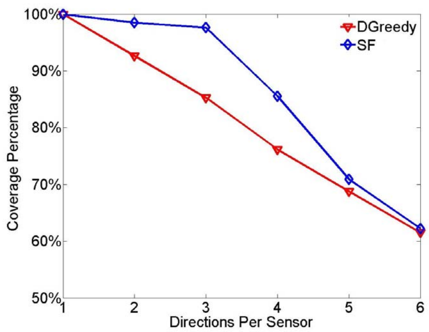

Fig. 4: maximal coverage vs. directions per sensor

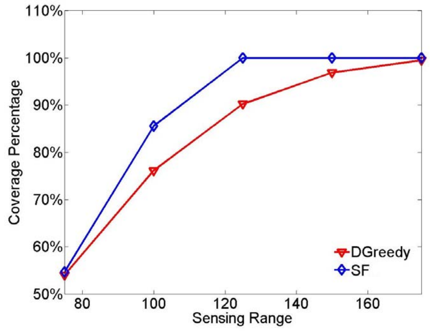

Fig. 5: maximal coverage vs. sensing range

5.2 Simulation Results of CRC

In this subsection, the initial energy in each sensor is 1000 unit. Both algorithms consider $\lambda_{i,j}^{E}(t)$ as the criteria of sensor scheduling and the surrounding information is used to make decisions.

In Fig. 6, simulations are executed in the situation that no POI appears in the scenario. In this traditional static scenario, there is no need to balance the energy consumption. Both our algorithms perform better than DGreedy algorithm.

For Fig. 7 and Fig. 8, POI is introduced into the scenario and $P(t)$ changes time to time.

In Fig. 7, the instantaneous coverage over POI is required which means that POI only needs to be covered at the moment that it changes its position. In this scenario, our algorithms maintain high performance. However, the performance of DGreedy algorithm drops significantly. ESF algorithm without re-sorting RankUB is also evaluated. The algorithm is called N-Extended Saving Factor (N-ESF) algorithm. The network lifetime in N-ESF algorithm is shortened dramatically. This result illustrates the necessity of the re-sorting process. Fig. 7 shows that both ESF algorithm and MN algorithm prolong the network lifetime significantly by balancing the energy consumption of the whole network.

In Fig. 8, the continuous coverage over POI is required which means that POI needs to be covered at any time if it exists. ESF algorithm maintains a high performance while the performance of MN algorithm drops.

A group is a set of sensors. Any two sensors in one group can deliver information to each other through other sensors. In our scenario, there is only one group at first. If two sensors are not neighbors, any change of one sensor does not affect the other. In MN algorithm, by exchanging information with neighbors, sensors receive necessary information and update their state in time. Therefore, in each scheduling round, the optimal sensing directions emerge in each group. If there is only one group in the network, these sensing directions are global optimums. Since the algorithm balances the energy consumption of the network, none of sensors exhausts very quickly. Optimal sensing directions can be achieved in a long period of time. Therefore, in Fig. 6 and Fig. 7, MN algorithm performs excellently. In Fig. 8, the long-term POI coverage leads to the unbalance of the energy consumption. Some sensors are forced to exhaust very quickly, then the network is separated into small groups. Blind areas appear because of the quick exhaustion of the small groups. The network is likely to miss POI as the blind areas increase. Since ESF algorithm can balance energy consumption among different groups in the network, its performance remains excellent and stable.

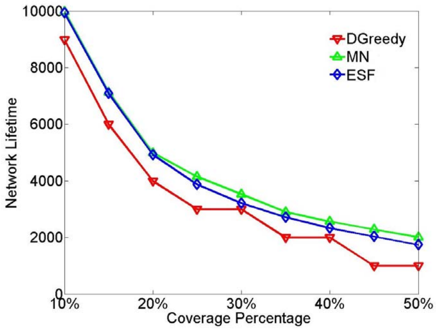

Fig. 6: network lifetime without POI

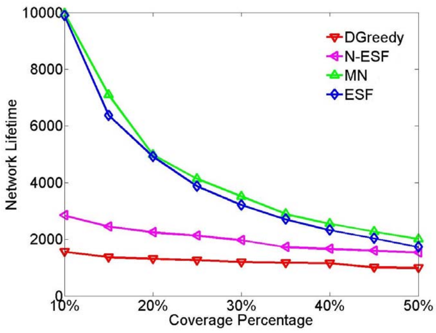

Fig. 7: network lifetime with instantaneous coverage

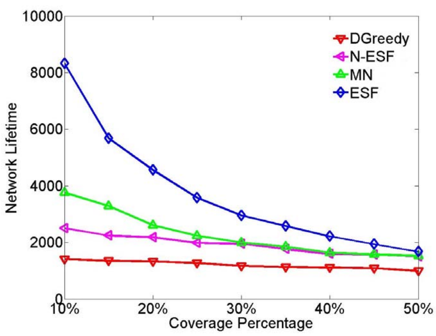

Fig. 8: network lifetime with continuous coverage

6 Conclusions

In this paper, a Constrained Regional Coverage problem of maximizing network lifetime has been solved by taking target coverage and area coverage into consideration. The dynamic POI is covered and the coverage ratio is achieved at the same time. It is noted that the dynamic scenario is more realistic and complicated than traditional static ones so that our scheduling schemes are more effective for practical applications. Several scheduling schemes are proposed to prolong the network lifetime through the well-designed priority factors. Meanwhile, network energy consumption is balanced by using our algorithms. Simulation results show that our algorithms prolong the network lifetime significantly. In our future work, more complicated POI model will be considered and our algorithms are to be applied to real systems with embedded smart cameras.

References

[1] S. Soro and W. Heinzelman, A Survey of Visual Sensor Networks, *Advances in Multimedia*, vol. 2009, pp. 1-21, 2009.

[2] M. Guvensan and A. Yavuz, On Coverage Issues In Directional Sensor Networks: A Survey, *Ad Hoc Networks*, vol. 9, no. 7, pp. 1238-1255, 2011.

[3] W. Cheng, S. Li, X. Liao, S. Changxiang and H. Chen, Maximal Coverage Scheduling in Randomly Deployed Directional Sensor Networks, in *Proc. of International Conference on Parallel Processing Workshops (ICPPW'07)*, Xi-An, China, pp. 68-73, Sept. 10-14, 2007.

[4] G. Fusco and H. Gupta, Selection and Orientation of Directional Sensors for Coverage Maximization, in *Proc. of the 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON'09)*, Rome, Italy, pp. 1-9, Jun. 22-26, 2009.

[5] Y. Cai, W. Lou and M. Li, Cover Set Problem in Directional Sensor Networks, in *Proc. of 2007 Future Generation Communication and Networking (FGCN'07)*, Jeju, Korea, pp. 274-278, Dec. 6-8, 2007.

[6] M. Wu and W. Lu, On Target Coverage Problem of Angle Rotatable Directional Sensor Networks, in *Proc. of the 7th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing(IMIS'13)*, Taichung, Taiwan, pp. 605-610, Jul. 3-5, 2013.

[7] I. Akyildiz, T. Melodia and K. Chowdhury, A Survey on Wireless Multimedia Sensor Networks, *Computer Networks*, vol. 51, pp. 921-960, 2007.

[8] C. Liu and G. Cao, Distributed Critical Location Coverage in Wireless Sensor Networks with Lifetime Constraint, in *Proc. of the 31st IEEE International Conference on Computer Communications (INFOCOM'12)*, Orlando, USA, pp. 1314-1322, Mar. 25-30, 2012.

[9] Y. Hong, D. Kim, D. Li, B. Xu, W. Chen and A. Tokuta, Maximum Lifetime Effective-sensing Partial Target-coverage in Camera Sensor Networks, in *Proc. of the 11th International Symposium on Modeling & Optimization in Mobile, Ad Hoc & Wireless Networks (WiOpt'13)*, Tsukuba Science City, Japan, pp. 619-626, May. 13-17, 2013.

[10] Y. Cai, W. Lou, M. Li and M. Li, Target-oriented Scheduling in Directional Sensor Networks, in *Proc. of the 26th IEEE International Conference on Computer Communications (INFOCOM'07)*, Anchorage, USA, pp. 1550-1558, May. 6-12, 2007.

[11] Y. Cai, W. Lou, M. Li and M. Li, Energy Effcient Target-oriented Scheduling in Directional Sensor Networks, *IEEE Trans. on Computers*, vol. 58, no. 9, pp. 1259-1274, 2009.

[12] X. Cao, X. Jia and G. Chen, Maximizing Lifetime of Sensor Surveillance Systems with Directional Sensors, in *Proc. of the 6th International Conference on Mobile Ad-hoc and Sensor Networks (MSN'10)*, Hangzhou, China, pp. 110-115, Dec. 20-22, 2010.

[13] Y. Kim, Y. Han, Y. Jeong and D. Park, Lifetime Maximization Considering Target Coverage and Connectivity in Directional Image/Video Sensor Networks, *The Journal of Supercomputing*, vol. 65, no. 1, pp. 365-382, 2013.

[14] Y. Hong, D. Kim, D. Li, W. Chen, A. Tokuta and Z. Ding, Target-temporal Effective-sensing Coverage in Mission-driven Camera Sensor Networks, in *Proc. of the 22nd International Conference on Computer Communications and Networks (ICCCN'13)*, Nassau, Bahamas, pp. 1-9, Jul. 30 - Aug. 2, 2013.