J Supercomput (2013) 65:365–382
DOI 10.1007/s11227-011-0646-9

Lifetime maximization considering target coverage and connectivity in directional image/video sensor networks

Yong-hwan Kim · Youn-Hee Han · Young-Sik Jeong · Doo-Soon Park

Published online: 23 July 2011

Abstract A directional sensor network consists of a large number of directional sensors (e.g., image/video sensors), which have a limited angle of sensing range due to technical constraints or cost considerations. In such directional sensor networks, the power saving issue is a challenging problem. In this paper, we address the Directional Cover and Transmission (DCT) problem of organizing the directional sensors into a group of non-disjoint subsets to extend the network lifetime. One subset in which the directional sensors cover all the targets and forward the sensed data to the sink is activated at one time, while the others sleep to conserve their energy. For the DCT problem proven to be the NP-complete problem, we present a heuristic algorithm called the Shortest Path from Target to Sink (SPTS)-greedy algorithm. To verify and evaluate the proposed algorithm, we conduct extensive simulations and show that it can contribute to extending the network lifetime to a reasonable extent.

Keywords Directional sensor networks · Scheduling · Target coverage · Connectivity · Energy efficiency

Y.-h. Kim · Y.-H. Han (✉)
Laboratory of Intelligent Networks, Advanced Technology Research Center, Korea University of Technology and Education, Cheonan, Korea
e-mail: yhhan@kut.ac.kr

Y.-h. Kim
e-mail: cherish@kut.ac.kr

Y.-S. Jeong
Grid Computing Laboratory, Wonkwang University, Iksan, Korea
e-mail: ysjeong@wonkwang.ac.kr

D.-S. Park
Division of Computer Science and Engineering, SoonChunHyang University, Asan, Korea
e-mail: parkds@sch.ac.kr

1 Introduction

Recently, multimedia sensor networks have attracted considerable research interests due to their vast and significant applications such as physical phenomenon or target detection, classification, and tracking [2, 9, 18, 20]. In contrast to the conventional sensor networks in which omni-directional sensors are deployed, the directional sensor networks, e.g., radar or image/video sensor networks [19, 21], have different sensing features. The most distinguishing characteristic of it is the limited sensing angle due to the constraints of manufacturing techniques, size and cost [1, 22]. Each directional sensor can sense only a sector of the disk, centered at itself, with the radius being equal to the sensing range. Moreover, in many types of directional sensors, the rotation function of directional sensors enables their directional coverage region to rotate and face to the selected orientation, and thus allows them to work in distinct orientations in cooperation with neighboring directional sensors. Unlike omni-directional sensor networks, the discrete target coverage in directional sensor networks is determined by both location and orientation of the sensors. This feature of directional sensor networks makes a network more complex. So, many methods for omni-directional sensor networks are not suitable for directional sensor networks.

In directional sensor networks, a power saving is still a critical issue because the sensor batteries can only store limited power and in most cases, they are hardly possible to replace their battery by a new one or be recharged [12, 25]. We assume that each directional sensor is non rechargeable and dies when it runs out its power. This issue is commonly resolved using a sensor wake-up scheduling protocol by which some sensors stay active to provide sensing services while the others sleep to conserve their energy. We consider the sensor scheduling problem to maximize network lifetime while maintaining both the target coverage and network connectivity in directional sensor networks.

In this paper, we propose a new problem, called the Directional Cover and Transmission (DCT) problem, the objective of which is to maximize the lifetime of a directional sensor network while not only continuous monitoring of all targets (target coverage) and forwarding the sensed data to the sink (connectivity) but also resolving the overlapped target issue. The overlapped target issue will be handled in Sect. 3.

Our strategy to solve this problem is to group all the deployed directional sensors into a number of sensor subsets. This subset of directional sensors is activated successively. Only sensors in an active subset are used to sense targets and to forward sensed data to the sink, and all the other sensors go into a sleep state. We call such a subset of directional sensors as the Directional Cover and Transmission (DCT) graph. The energy consumption of each sensor is directly related to the amount of data sensed and relayed by the directional sensor. Accordingly, we propose a new sensor's energy consumption model that is applicable to properties of directional sensor networks. We also reformulate the DCT problem by the sensor energy consumption model and developing a new graph model. We call the problem of finding DCT graphs and allocating the work time for each of them as Maximum Directional Cover and Transmission Graph (MDCTG) problem. Due to the NP-completeness of the MDCTG problem, we design a heuristic algorithm to solve the MDCTG problem, called the Shortest Path from Target to Sink (SPTS)-greedy, which uses a greedy method to generate a maximum number of DCT graphs. To verify and evaluate the proposed algorithm, we conduct extensive simulation and show that our problem formulation and the proposed algorithm can contribute to extending the network lifetime largely.

Some applications such as target classification and surveillance systems are required the accuracy and reliability of the observations [10]. For these conditions, we simply extend our problem to k-target coverage, in which each target should be covered by directions of at least k ($k \geq 1$) different sensors at any time.

The remainder of this paper is organized as follows. Section 2 briefly reviews the related works. In Sect. 3, we introduce our system model and describe the DCT problem with an example. Section 4 presents our solution, which consists of six parts: we introduce the DCT graph in Sect. 4.1, propose the sensor's energy consumption model in Sect. 4.2, present the MDCTG problem in Sect. 4.3, propose our SPTS-greedy algorithm in Sect. 4.4, and finally present extended version of our problem by considering k-target coverage in Sect. 4.5. Section 5 shows the simulation results for our algorithm, and Sect. 6 concludes our paper.

2 Related works

For omni-directional sensor networks, many scheduling algorithms to prolong the network lifetime while guaranteeing certain coverage has been studied [5]. A scheduling problem on the complete area coverage, which represents how well a region of interest is monitored, is studied in [8, 11, 23]. The target coverage is one of the fundamental measures of the quality of service (QoS) of the sensing function. The goal is to have each target in the physical space of interest within the sensing range of at least one sensor. [4] introduced the target coverage problem, where disjoint sensor sets are modeled as disjoint cover sets, such that every cover set completely monitors all the targets. This problem was called MSC (Maximum Set Covers). The MSC problem was proved to be NP-complete in the study. This problem was further extended in [6], where sensors were not restricted to participation in only disjoint sets, that is, a sensor could be active in more than one set.

In [23], both area coverage and communication connectivity are considered in the scheduling algorithms for omni-directional sensor networks. If the communication radius is at least twice of the sensing radius, complete area coverage implies network connectivity among the active sensors. Other previous works [14, 15, 17, 24] addressed network lifetime maximization problems by grouping sensors into cover trees, each of which can guarantee both the target coverage and the connectivity to a sink.

The scheduling problems in directional sensor networks have recently gained intense interest. Compared to omni-directional sensors, directional sensors are obviously different in that the coverage region of a directional sensor is determined by both its location and orientation. The work relevant to the coverage issue in directional sensor networks is presented in [1, 3, 22] which aim to maximize the network lifetime by finding cover sets in each of which the directions cover all the targets. The scheduling problem addressed in this paper differs from the existing works because we consider the following issues at the same time: (1) directional coverage, (2) connectivity, and (3) overlapped target. For simplicity, directional sensors henceforth will be called sensors.

3 System model and problem statement

In our system model, a directional sensor network is composed of N sensors, each of which has W directions and operates only one direction (current direction) with a uniform sensing range at any instance. All sensors are randomly scattered to cover M targets in a two-dimensional plane. When the sensors are randomly deployed, each sensor initially faces to one of its directions. That is, directions of a sensor do not work at the same time. The active sensing region is determined by the chosen direction of the sensor. A direction of sensor can cover a target if the sensor faces in the direction and there is the target within the active sensing region. Each sensor is able to vary its communication ranges, using a possibly large set of transmission power levels. The maximum communication range is achieved when the maximal allowable transmission power is used. Two sensors are connected if they are within each other's maximum communication range. All the sensors can communicates with the sink via single-hop or multi-hop communication. All deployed sensors are homogeneous in terms of sensing and communication range and initial energy. We also assume that each sensor is aware of their location by using an arbitrary localization method [16]. The active sensors are divided into two types: source or relay sensor. A source sensor covers one or more targets and generates messages to send the sensed data to the sink. A relay sensor delivers the messages to other relay sensor or the sink.

We show an example of a directional sensor network in Fig. 1. s_n $(1 \leq n \leq 3)$ and p_m $(1 \leq m \leq 5)$ denote the set of sensors and targets, respectively. Each sensor has three directions in Fig. 1. $d_{n,i}\,(1 \leq i \leq 3)$ represents the directions of s_n. In this figure, $d_{1,3}$, $d_{2,3}$, and $d_{3,2}$ denote the current direction of s_1, s_2, and s_3, respectively. A target

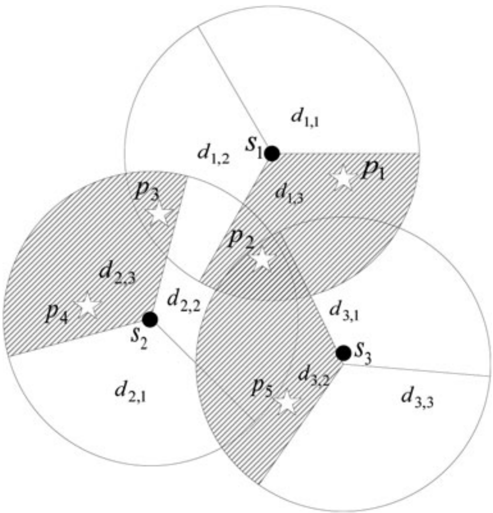


Fig. 1 An illustrative example for a directional sensor network with five targets of interest

can be sensed (or covered) only when it is within at least one active sensing region. For example, p_1 and p_2 are covered simultaneously by $d_{1,3}$ of s_1.

In this paper, the network lifetime is defined as the duration up to the time when there exists a target that can no longer be covered by current diction of at least one source sensor or sensed data cannot be forwarded to the sink any longer due to the depletion of energy of the sensors.

It is noted that there are overlapped targets that simultaneously are covered by directions of adjacent sensors. The adjacent sensors can usually gather the same data from the overlapped targets and deliver them to the sink. In Fig. 1, p_2 is an overlapped target since it is covered by both $d_{1,3}$ and $d_{3,2}$. Although such data duplication might be helpful to enhance the data reliability, multiple transmissions of the same data obviously have an adverse effect on the extension of network lifetime. To prevent such a redundancy, only one direction among directions of adjacent sensors that cover an overlapped target should transmit the sensed data to a relay sensor or the sink.

The Connected Target Coverage (CTC) problem has been considered as a representative network lifetime maximization problem that considers the connectivity to a sink as well as the continuous monitoring of all targets with known locations in omni-directional sensor networks [24]. In this paper, we present a new problem, the DCT problem, of which objective is to maximize the network lifetime in directional sensor networks. The DCT problem is more constrained than the CTC problem and it is defined as follows.

Definition 1 *DCT (Directional Cover and Transmission) Problem*: Given a set of targets located at fixed known positions and a set of sensors that are randomly deployed in the targets' vicinity, schedule the sensors' activity to satisfy the following four requirements:

(1) *Target coverage requirement:* all targets should always be covered by at least one current direction of source sensor;

(2) *Connectivity requirement:* there should be a route to deliver sensed data from each target to the sink through a subset of sensors;

(3) *Redundancy removal requirement:* only one among directions of adjacent sensors that cover an overlapped target should transmit the sensed data to a relay sensor or the sink;

(4) *Lifetime maximization requirement:* the lifetime of a directional sensor network should be maximized.

Figure 2 illustrates two possible schedules to satisfy the requirements of the DCT problem. In the figure, there are nine sensors, three targets, and one sink. There exists a sensing link (a dashed edge) between a source sensor and a target if a direction of the source sensor can cover the target. Some sensing links are represented as directed dashed edges if the source sensor processes and transmits data sensed from the target. For each target, it is noted that there should be only one directed dashed edge to satisfy the overlapped target requirement of the DCT problem. There exists a communication link (a solid edge) between two sensors if the two sensors are connected. Some communication links between two sensors are represented as directed solid edges if the two sensors exist on a route that can be used to relay data from

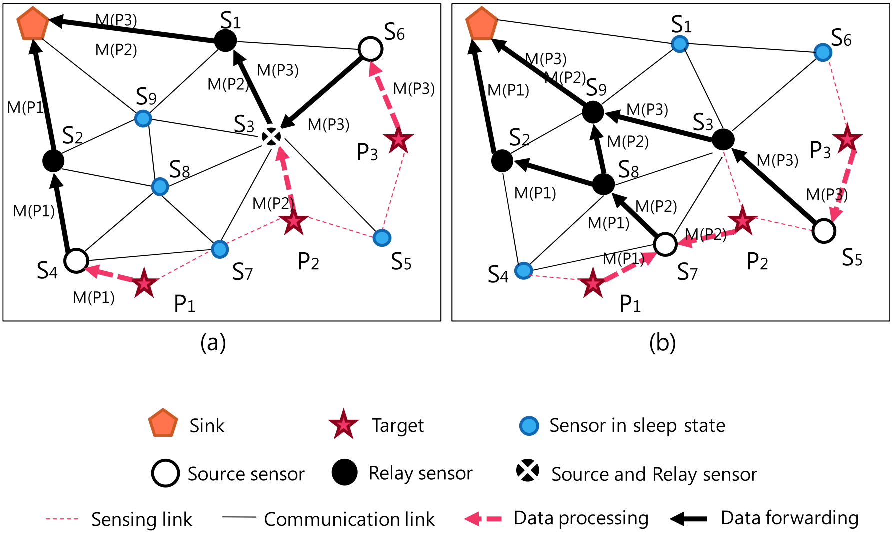


Fig. 2 Illustration of DCT problem ($M(P_x)$ denotes a message including data sensed from target P_x)

sources to the sink. This figure illustrates that only a subset of the deployed sensors is sufficient to satisfy the requirements of the DCT problem and different subsets can be activated in different intervals. The other sensors that are not included in the subsets will go into the sleep state to conserve power.

4 Proposed algorithm

In this section, we first introduce the DCT graph and a new sensor's energy consumption model. Then, we reformulate the DCT problem into the MDCTG problem that has a form of classical maximization problems and propose our greedy algorithm for solving the MDCTG problem. Finally, we present extended versions of our problem by considering k-target coverage.

4.1 Network model

Let $S = \{s_1, s_2, \ldots, s_N\}$ ($|S| = N$) and $P = \{p_1, p_2, \ldots, p_M\}$ ($|P| = M$) denote the set of deployed N sensors and the set of M targets, respectively. We also define $D = \{d_{ij} | i = 1 \ldots N, j = 1 \ldots W\}$ ($|D| = N * W$) as the set of directions. $d_{i,j}(1 \leq j \leq W)$ represents the jth direction of s_i. We use R to denote the sink node. The directional cover and transmission structure of the given sensor network are modeled as a weighted graph $G = (V, E)$, where V represents the sensors, targets and sink node ($V = S \bigcup P \bigcup \{R\}$), and E contains the set of sensing links and communication links with an associated edge cost function based on the energy consumption.

We assume that the active/sleep state of each sensor does not change in an *OTI (Operation Time Interval)*, denoted by τ. Let $S_s(\tau)$ and $S_r(\tau)$ denote the sets of

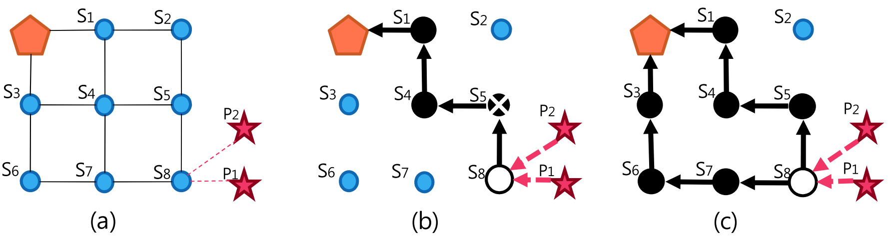


Fig. 3 An illustrative example for cover tree and DCT graph: (**a**) initial network, (**b**) cover tree, and (**c**) DCT graph

active sensors selected as sources and relays, respectively, in an OTI τ. Therefore, it is evident that the set of all active sensors in τ is $S_s(\tau) \bigcup S_r(\tau)$. Now, we define the DCT graph $G(\tau)$ as follows.

Definition 2 *DCT (Directional Cover and Transmission) graph* $G(\tau)$: A DCT graph $G(\tau) = ((S_s(\tau) \bigcup S_r(\tau) \bigcup P \bigcup \{R\}), E'(\tau))$ is a DAG (directed acyclic graph) that satisfies the requirements (1), (2), and (3) of the DCT problem in OTI τ. $E'(\tau)$ is the set of sensing links (between targets and source sensors) and communication links (between active sensors) in OTI τ.

This DCT graph is similar to the cover tree defined in [24]. However, the vertices in the cover tree represent only the source and relay sensors (and not the targets), whereas the vertices of the DCT graph represent the targets as well as the source and relay sensors. This implies that the cover tree does not take the overlapped target into consideration, and therefore, it allows sensors to transmit the same data and causes them to waste energy because of such redundancy. In the DCT graph, on the other hand, it should be noted that only one direction among the directions of adjacent sensors that monitor an overlapped target gathers the sensed data and transmits them to a relay sensor or the sink. Actually, Fig. 2 shows two DCT graphs where $E'(\tau)$ is represented by the directed dashed edges and the directed edges.

In addition, DCT graph based on DAG can lift the assumption of tree-based approaches. That is, DCT graph can find more possible route than the tree-based approaches. This feature allows the network load to be more distributed to many sensors so that the network lifetime can be further extended. From Fig. 3(c), we can observe that different data sensed from each target can be delivered to the sink through different routes. In Fig. 3, for example, sensor S_8 transmits the data sensed from target P_1 to relay sensor S_5, while sensor S_8 transmits the data sensed from another target, P_2, to a different relay sensor, S_7. In tree-based approaches, on the other hand, sensor S_8 should always transmit the data sensed from two targets to relay sensor S_5.

4.2 Sensor's energy consumption model

We assume that the scheduling algorithm is pre-computed at the sink, and the results are disseminated to sensors by the sink at the system initialization. When the system starts operation, all sensors work according to the schedule, such as when and for what duration to sleep, monitor targets, or relay data. To pre-computing, we suggest the energy consumption model for the sensors that mainly reflects the energy consumed for sensing/relaying data and rotating the orientation of a sensor. This model is used to calculate the residual energy of each sensor in our scheduling algorithm.

A sensor consumes energy depending on how much data are generated, transmitted, and received [13]. In the case of the target coverage scenario, each target needs to be monitored by at least one sensor, which transmits the data to the sink. We assume that all sensors have the same data generation rate for a target and therefore a fixed number of bits, $BR(\tau)$, is generated by each direction in OTI τ. This means that all sensors use the same sampling frequency, quantization, modulation, and coding scheme for each target. This also indicates that each sensor consumes a different amount of energy according to the number of targets that is covered by its current direction.

Let e_s and e_r denote the energy consumed for sensing and receiving a bit of data, respectively. Assuming that e^t_{ij} denotes the energy consumed by a sender s_i to transmit a bit to a receiver s_j, it is expressed as

$$e^t_{ij} = e_t + b \times d^{\alpha}_{ij}, \tag{1}$$

where e_t and b are constants, d_{ij} is the Euclidean distance between sensor s_i and s_j, and α is the path loss factor.

Given the DCT graph $G(\tau)$ for OTI τ, the energy consumption model for sensor s_i is given by

$$E(s_i, G(\tau)) = \begin{cases} BR(\tau)e_s\theta_{s_i} + BR(\tau)e^t_{ij}\phi_{s_i}(\tau), \quad \text{if } s_i \in S_s(\tau) \text{ and } s_i \notin S_r(\tau) \\ BR(\tau)(e_r + e^t_{ij})\psi_{s_i}(\tau), \quad \text{if } s_i \notin S_s(\tau) \text{ and } s_i \in S_r(\tau) \\ BR(\tau)e_s\theta_{s_i} + BR(\tau)e^t_{ij}\phi_{s_i}(\tau) + BR(\tau)(e_r + e^t_{ij})\psi_{s_i}(\tau), \\ \quad \text{if } s_i \in S_s(\tau) \text{ and } s_i \in S_r(\tau) \\ 0, \quad \text{if } s_i \notin S_s(\tau) \text{ and } s_i \notin S_r(\tau) \end{cases} \tag{2}$$

where θ_{s_i} denotes the number of targets covered by current direction of a source sensor s_i; $\phi_{s_i}(\tau)$, the number of targets of which data are transmitted by a source sensor s_i ($\phi_{s_i}(\tau) \le \theta_{s_i}$); and $\psi_{s_i}(\tau)$, the number of targets of which data are delivered by a relay sensor s.

4.3 Problem formulation

In this section, we modify the DCT problem into a new problem called the MDCTG problem by using the proposed DCT graph and sensor's energy consumption model from the above discussion. The MDCTG problem has the form of classical maximization problems. The MDCTG problem is defined as follows.

Definition 3 MDCTG (Maximum Directional Cover and Transmission Graph) Problem: Given a graph $G = (V, E)$ and the initial energy $E_0(s)$ of each sensor s, where $V = S \bigcup \{R\} \bigcup P$ and E includes both sensing links and communication links, find a set of DCT graphs $G(\tau_1), G(\tau_2), \dots, G(\tau_x)$ and their OTIs $\tau_1, \tau_2, \dots, \tau_x$ such that the network lifetime, denoted as $L(S, P, R, E)$, is maximized.

Mathematically, the MDCTG problem is defined as

$$\text{Maximize} \quad L(S, P, R, E) \equiv \sum_{i=1}^{x} \tau_i \tag{3}$$

$$\text{subject to} \quad \sum_{i=1}^{x} E(s, G(\tau_i)) \leq E_0(s), \quad \forall s \in S, \tag{4}$$

$$\text{where} \quad \tau_i \geq 0. \tag{5}$$

In this problem definition, (4) guarantees that the total energy consumed by each sensor si across all DCT graphs is not larger than its initial energy. Given a directional sensor network deployed in an area, the number of DCT graphs, denoted by x, is finite but unknown. It should be noted that the duration of any two OTIs may be different. In addition, a sensor can appear in different DCT graphs, i.e., the sets of sensors in different DCT graphs need not be disjoint.

The decision version of the MDCTG problem is to determine whether there exists a set of DCT graphs $G(\tau_1), G(\tau_2), \ldots, G(\tau_x)$ and their OTIs $\tau_1, \tau_2, \ldots, \tau_x$ such that for a given initial energy of each deployed sensor, $\sum_{i=1}^{x} \tau_i$ is larger than or equal to a given value T. In this paper, for simplicity, we omit the complete proof of the fact that the MDCTG problem is NP-complete.

The MDCTG problem is proved to be NP-Complete by simple reduction to a Maximum Cover Tree (MCT) problem, which is NP-Complete in a previous study [24]. The MCT problem is to find a number of cover trees each of which can cover all the targets and can send all the sensed data to the sink such that the lifetime of omni-directional sensor networks is maximized. The MDCTG problem is more complex than the MCT problem because the MCT problem does not take the overlapped target issue into consideration when finding a cover tree. Moreover, unlike omni-directional sensor networks, target coverage in directional sensor networks is further complex. Consequently, the MDCTG problem is NP-Complete. A sophisticated proof can be found in [24].

4.4 Proposed sensor scheduling algorithm

In this section, we propose a new heuristic algorithm called the SPTS-greedy algorithm to solve the MDCTG problem. The SPTS-greedy algorithm uses a greedy method to produce DCT graphs and their OTIs by finding the best data routes from each target to the sink in the weighted graph $G = (V, E)$ with the associated edge cost function based on the energy consumption model proposed in Sect. 4.2.

The SPTS-greedy algorithm takes $S, D, P, R, E, E_0(s_i)$ of each sensor s_i, and maximum τ as the input parameters. The output of the proposed algorithm is a sequence of DCT graphs $G(\tau_1), G(\tau_2), \ldots, G(\tau_x)$ and their OTIs $\tau_1, \tau_2, \ldots, \tau_x$. We define the following notations used in the SPTS-greedy algorithm.

- S_l: set of living sensors;
- D: set of directions of living sensors. $D = \{d_{ij} | i = 1 \ldots N, j = 1 \ldots W\}$;
- $P(d_{ij})$: set of targets covered by direction d_{ij};

SPTS-greedy algorithm $(S,\ D,\ P,\ R,\ \tau)$

$S_l = S;\ S = \phi;\ x = 1;$
while $\bigcup_{s \in S_l} P_s = P$,
  for each $d_{ij} \in D$
    for each target $p \in P$
      if target p is covered by d_{ij} **then**
        $w(p, d_{ij}) = e_s \times |P(d_{ij})|;\ LW = LW \cup \{w(p, d_{ij})\};$
      end if
    end for
  end for
  for each link(s_i, s_j)
    $w_{i,j} = E_0(s_i)/E_r(s_i) \times e^t_{ij};\ LW = LW \cup \{w_{i,j}\};$
  end for
  $x = x + 1;\ D = \phi;\ P' = \phi;\ \tau_x = \tau;\ G(\tau_x) = \{R\};$
  while $P' \neq P$
    Find a critical target $p \in P;\ P' = P' \cup \{p\};\ G(\tau_x) = G(\tau_x) \cup \{p\};$
    Find a route $R(p)$ with the minimum total weight;
    $LW = LW - \overline{w}(p, d_{ij});$
    for each $s \in \overline{R}(p)$, $G(\tau_x) = G(\tau_x) \cup \{s\}$; **end for**
    for each $link(s_i, s_j) \in L(p)$, $w_{i,j} = w_{i,j} + e_{\text{trans}}$; **end for**
  end while
  for each $s \in G(\tau_x)$, $\tau_x = \min(\tau_x, \frac{E_r(s)}{E(s, G(\tau_x))}\tau_x)$; **end for**
  for each $s \in G(\tau_x)$, $E_r(s) = E_r(s) - E(s, G(\tau_x))$; **end for**
  Remove dead and isolated sensors from S_l;
  for each sensor $s_i \in S_l$, $D = \cup_{j=1}^{W}\{d_{i,j}\}$; **end for**
end while
return $G(\tau_1), G(\tau_2), \ldots, G(\tau_x)$ and $\tau_1, \tau_2, \ldots, \tau_x$

Fig. 4 SPTS-greedy algorithm

- x: index of DCT graphs;
- $E_r(s)$: residual energy of sensor s;
- LW: set of link weights;
- $w(p, d_{ij})$: weight of sensing link between target p and direction d_{ij};
- $\overline{w}(p, d_{ij})$: weights of sensing links between a target and directions of sensor s_i except d_{ij} covering target p;
- $w_{i,j}$: weight of communication link between sensor i and sensor j;
- $R(p)$: set of sensors in route from target p to the sink (e.g., $R(p) \equiv s_1, s_2, \ldots, R$);
- $\overline{R}(p)$: set of sensors in route $R(p)$ excluding the sink;
- $L(p)$: set of communication links in route $R(p)$.

Figure 4 shows the SPTS-greedy algorithm. The algorithm repeatedly builds the DCT graphs and stops once the entirety of each target is covered by at least one direction of live sensors. The algorithm consists of the following steps:

Step (1) Assign a weight value of sensing link (lines 3–9). we also assign a weight value $w_{i,j}$ to each communication link (s_i, s_j) as follows (lines 11–13):

$$w_{i,j} = E_0(s_i)/E_r(s_i) \times e_{\text{trans}}. \tag{6}$$

In (6), $w_{i,j}$ reflects both the communication energy consumption on the link and the residual energy level of the sending node s_i.

Step (2) Use a greedy method to choose active sensors to construct a DCT graph until all the targets are covered. A target covered by a larger number of directions is selected as the *critical target* (line 15). Once the critical target is selected, our algorithm finds a unique route from the critical target to the sink in the weighted graph $G = (V, E)$ such that the sum of the link weights of the route is minimized (line 16). To determine the unique route, a well-known technique such as Dijkstra's algorithm can be used to find the shortest path. Once a unique route from the critical target to the sink is found, we have to remove $\overline{w}(p, d_{ij})$ because each sensor can face to only one of its directions at any instance (line 17), and a set of sensors in the route are added to the current DCT graph (line 18), and the weight values of communication links in the route are updated (line 19). If unique routes from all targets to the sink are found, a new DCT graph is constructed.

Step (3) Each DCT graph operates during the given fixed time duration OTI τ for which all sensors remain in the same state. The OTI of a DCT graph is determined by the sensor that has the least operational time until death (line 21). If an active sensor will die soon because of a lack of energy, the operational time of a DCT graph depends on such a sensor, e.g., the sensor that has the least operational time until death. Thus the OTI of the xth DCT graph is given by

$$\tau_x = \min\left(\tau, \min_{s \in G(\tau_x)}\left(\frac{E_r(s)}{E(s, G(\tau))}\tau\right)\right). \tag{7}$$

Step (4) After a DCT graph $G(\tau_x)$ and its OTI τ_x are obtained, the residual energy of each sensor in the DCT graph $G(\tau_x)$ is updated by using the energy consumption model presented in Equation (2) (line 22). If a sensor has no residual energy, we call it a dead sensor. If a sensor has residual energy but cannot find a route from itself to the sink node in the weighted graph $G = (V, E)$, we call it an isolated sensor. The dead and isolated sensors are removed from S_l (line 23). Before finding a new DCT graph, the D is updated based on the S_l (line 24).

Step (5) After obtaining all DCT graphs and their OTIs, the algorithm returns them.

4.5 k-target coverage version

To maximize network lifetime, we eliminate the redundancy of data by considering the overlapped target. However, some applications are required the accuracy and reliability of data. For these conditions, we simply extend our problem to k-target coverage where each target should be covered by at least k directions at any time.

As the DCT problem cannot support k-target coverage in directional sensor networks, we modify four requirements of the DCT problem and the corresponding DCT graph.

(1) *k-target coverage requirement*: all targets should be always covered by at least k directions at any time;
(2) *Connectivity requirement*: there should be routes to deliver sensed data from source sensor to the sink through a subset of sensors;
(3) *Redundancy removal requirement*: only k current directions among directions of adjacent sensors that cover an overlapped target should transmit the sensed data to a relay sensor or the sink;
(4) *Lifetime maximization requirement*: the lifetime of a directional sensor network should be maximized.

That is, there should be k current directions to cover one target at any time, which can be represented as sensing links between k source sensors and one target in the DCT graph. And there should be the routes for the source sensors to send data back to sink. Thus, our problem is transformed into the sensor scheduling problem to maximize network lifetime while maintaining both the k-target coverage and network connectivity. We call this problem as k-target coverage version of the DCT problem.

For solving k-target coverage version of the DCT problem, we simply modify the SPTS-greedy algorithm. The modification is that we produce the DCT graphs by finding k routes from each target to the sink in the weighted graph $G = (V, E)$ with the associated edge cost function based on the energy consumption model, provided that sensors are redundantly deployed.

5 Simulation results

In this section, we first verify the SPTS-greedy algorithm and evaluate its performance. To conduct simulations, we implemented a simulator with JDK 6.0. Using the simulator, we constructed the simulation environments to build a directional sensor network environment.

As the connectivity issue is not considered in most existing works dealing with the target coverage problem in directional sensor networks, to demonstrate the effectiveness of our algorithm, we compare its performance with that of the Communication Weighted Greedy Cover (CWGC) algorithm proposed in [24] with suitable modification to adapting in directional sensor networks. For this, a sensor operates only one direction to have the limited sensing angle at any instance. The CWGC algorithm greedily the source sensor set to cover the targets and constructs cover tree to using the shortest path. For simplicity, we denote the network lifetimes obtained by the SPTS-greedy and CWGC algorithms as NT_{SPTS} and NT_{CWGC}, respectively.

We simulate a stationary network with sensors and targets randomly deployed in a 100 m × 100 m area. The sink node is placed in the middle of the area (at the position (50 m, 50 m)). The initial energy of each sensor is set to be 20 J. The values of various parameters are chosen as follows: $\tau = 60$ sec, $e_t = 50$ nJ/bit, $b = 100$ pJ/bit/m^4, $\alpha = 4$, $e_s = 150$ nJ/bit, and $e_r = 150$ nJ/bit [7, 24]. The data sensed from a target is

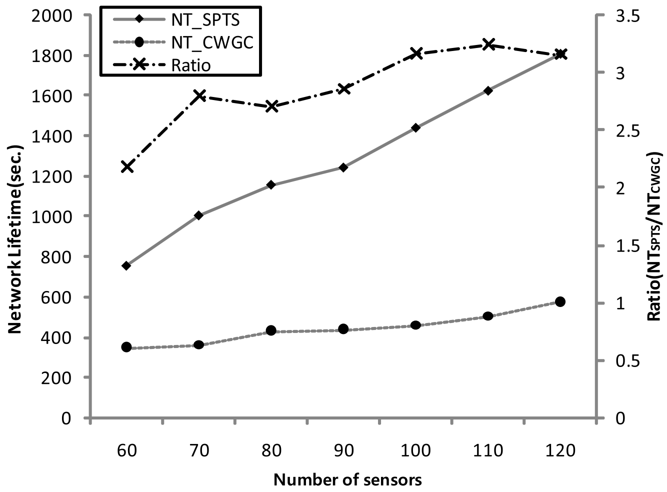


Fig. 5 Effect of the number of sensors on the network lifetime ($M = 20$)

generated by each source node at the rate of 10 Kbps. We assume that all sensors have the same sensing range (20 m), the same maximum communication range (40 m) and the number of directions is 3 ($W = 3$). By default, we consider 20 targets ($M = 20$) and 100 sensors ($N = 100$), where the numbers will be changed to verify the variation in the performance versus the number of sensors and targets.

5.1 Impact of network parameters

Figure 5 shows the network lifetime obtained by the SPTS-greedy algorithm relative to that obtained by the CWGC algorithm. Each value plotted on the figure is the average result of 100 randomly generated networks. The number of sensors N is varied between 60 and 120 to analyze the effect of sensor density on the network lifetime. It is observed that the network lifetime apparently increases with the number of sensors because more sensors can be scheduled to cover targets and relay data. It is evident that the proposed SPTS-greedy algorithm always outperforms the CWGC algorithm and the performance gap between NT_{SPTS} and NT_{CWGC} increases with the number of sensors (see the ratio of NT_{SPTS} to NT_{CWGC}). When more sensors are deployed, each target can be covered by more directions, and thus, the redundancy caused by the overlapped targets increases. In this situation, the proposed algorithm selects only one direction that deals with an overlapped target to eliminate such redundancy. The proposed scheme also finds a more energy-efficient route to a sink for sensed data from a target by utilizing per-target routing policy when the number of deployed sensors increases.

Figure 6 shows a plot of the variation in the network lifetime with the number of targets. With an increase in the number of targets, a sensor consumes more energy because the amount of data produced by the targets increases in the network. Therefore, the network lifetime apparently decreases with an increase in the number of targets. The figure also shows that NT_{SPTS} is always greater than NT_{CWGC} and the

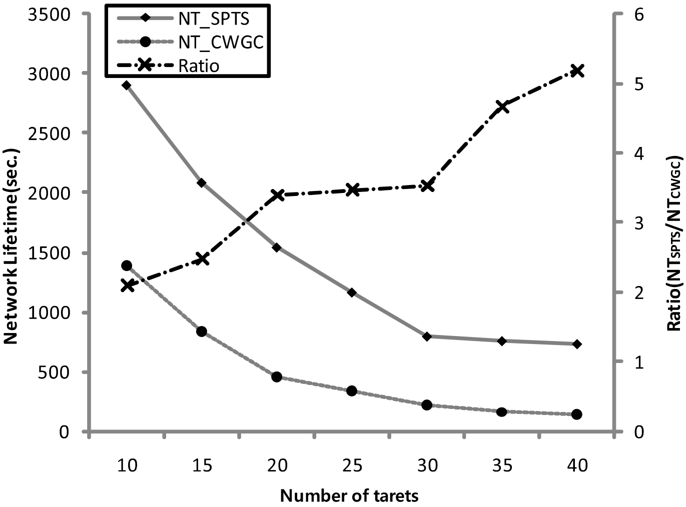
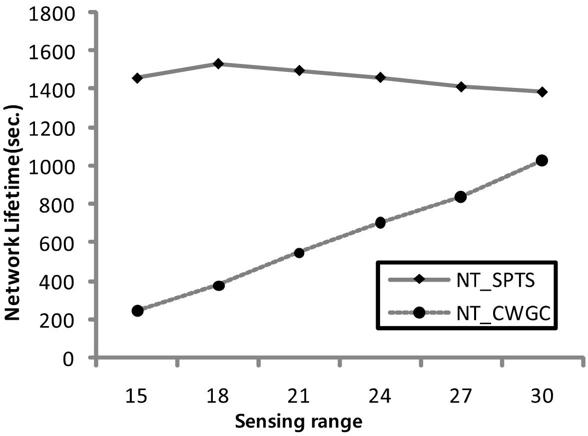


Fig. 6 Effect of the number of targets on the network lifetime ($N = 100$)

Fig. 7 Effect of the sensing range on the network lifetime ($N = 100$ and $M = 20$)



gap between the two increases with the number of targets. This is also attributable to the fact that the SPTS-greedy algorithm can remove the redundancy caused by overlapped targets and that it uses the approach based on the graph.

Figure 7 shows the effect of the sensing range and maximum communication range on the network lifetime. Usually, the number of sensing and communication links increases in the weighted graph $G = (V, E)$ with the sensing range and communication range. In such the graph with many sensing and communication links, there will be more energy-efficient routes from each target to the sink. From Fig. 7, we can observe that both algorithms can suitably find such energy-efficient routes. It is also noted that when the sensing range become high, the network lifetimes obtained by the SPTS-greedy algorithm do not change significantly and the performance gap between both algorithms becomes less significant. This indicates that the proposed scheme can find

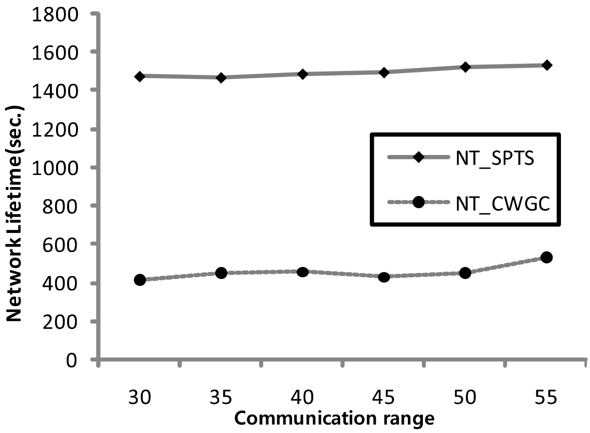


Fig. 8 Effect of the maximum communication range on the network lifetime ($N = 100$ and $M = 20$)

energy-efficient routes even when the sensing range are not high. In other words, the SPTS-greedy algorithm can also save more energy of sensors by reducing the sensing range, but it can still produce energy-efficient routes. Figure 8 shows the effect of the maximum communication range on the network lifetime. In general, more energy-efficient route will be found by increasing communication links. However, we found that the network lifetimes obtained by two algorithms do not change significantly even if many communication links are created by wider communication range. The reason is that these links are not used because of the inefficiency by long distance.

5.2 Impact of topology

The simulation results shown in the previous sections provided the network lifetimes of the SPTS-greedy and CWGC algorithms when the sensors are randomly (or uniformly) distributed at the initial deployment. Usually, sensors that are close to the sink are likely to consume more energy than others and quickly reach the dead state because such sensors are frequently selected to relay data to the sink. This will create a cascading effect that will shorten network lifetime. On the other hand, the sensors furthest from the sink are more likely to live when the network lifetime is expired. Therefore, the network lifetime will be high if more sensors are deployed in the vicinity of the sink node. To verify it, we simulate the performance on other types of topologies for the initial deployment. That is, we will use the normal distribution with diverse values of the standard deviation (σ) for the positions of all sensors.

Figure 9 shows the network lifetimes obtained by our algorithm under different σ when sensors are deployed following the normal distribution. The mean value for positions of all sensors is always the position of the sink (position (50 m, 50 m)) and σ is varied between 20 and 60 with an increment of five. From Fig. 9, we examine that the network lifetime will be high more than uniformly distributed when more sensors are deployed in the vicinity of the sink node (σ is between 20 and 40). However, its performance becomes low when most sensors are gradually situated at the region furthest from the sink (σ is between 45 and 60). If σ is less than 20, most sensors are concentrated around the position of the sink, and thus, some targets located at outer

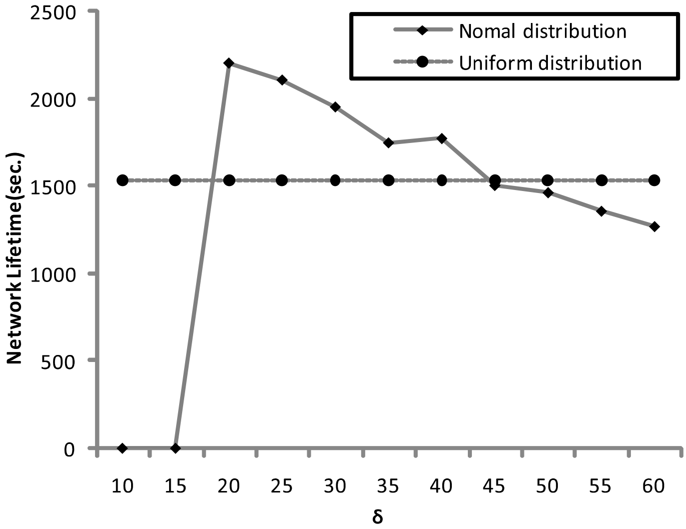


Fig. 9 Effect of sensor distribution on the network lifetime ($N = 100$ and $M = 20$)

positions are not covered by any direction from the initial deployment (implying that the network lifetime is zero).

5.3 Analysis of k-target coverage

The effect on the network by the k-target coverage as well as the overlapped target is studied by simulations in this section. Both the k-target coverage and overlapped target may be helpful to enhance the accuracy and reliability of data by duplicate transmission. However, there is much difference between them. In the case of k-target coverage, all targets are fairly covered by at least k directions. In contrast, the overlapped target may be unfairly covered by arbitrary directions. Therefore, duplicate data transmission for the overlapped target is not significant to enhance the accuracy and reliability. These results are shown in Table 1. It presents the number of data transmission for each target measured in two algorithms. In our solution for k-target coverage, each target is exactly covered by k directions because of the k-target coverage requirement and overlapped target requirement. On the other hand, we can show that the number of transmission varies from target to target in the CWGC algorithm.

Figure 10 shows the network lifetime varied by the required degree k of target coverage. The number of sensors is varied from 60 to 120 with an increment of 10. As expected, the network lifetime decreases sharply with the increment of coverage degrees since more sensors can be involved to cover a target and relay data resulting in the increment of sensor energy consumption. This implies that the network lifetime of the CWGC algorithm is similar to the one of our solution with degree 3 ($k = 3$) and it gradually becomes lower than the one of our solution with degree 3 when the number of sensors becomes high because the duplicate transmission caused by the overlapped target increases.

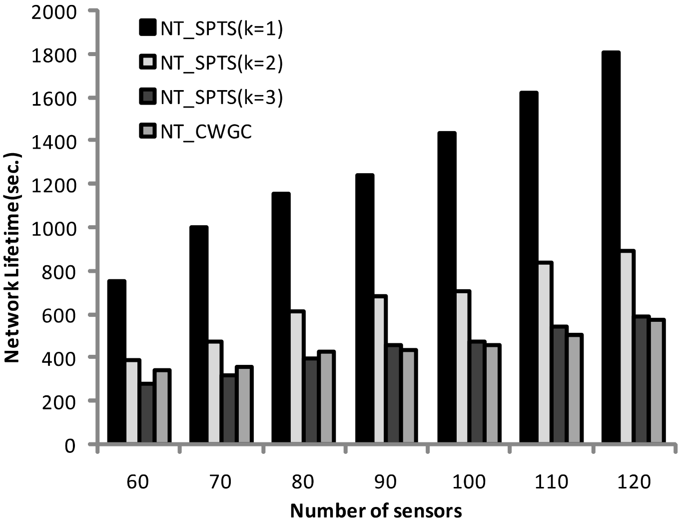


Fig. 10 The effect on the network lifetime by k-target coverage ($N = 100$ and $M = 20$)

Table 1 The number of data transmissions for ten targets ($N = 100$ and $M = 20$)

ID of a target	1	2	3	4	5	6	7	8	9	10	Average
SPTS-greedy ($k = 3$)	3	3	3	3	3	3	3	3	3	3	3
CWGC	2	1	3	3	1	4	4	2	2	3	2.5

6 Conclusion

In this paper, we have proposed an algorithm to schedule the active time of sensors such that (1) the directions of active sensors can cover all the targets, (2) only one among directions of adjacent sensors that cover an overlapped target should transmit the sensed data to a relay sensor or the sink, and (3) such sensors find energy-efficient and unique routes from each target to the sink. We first proposed a new scheduling problem called the DCT problem, to maximize the network lifetime such that each active set of sensors satisfies the above mentioned conditions. From the simulation study, we also showed that the proposed approximation algorithm, called SPTS-greedy, is suitable for solving the DCT problem and we analyze its performance in terms of network lifetime. The improved performance gain comes from (1) eliminating the transmission of the same data from overlapped targets and (2) constructing many DCT graphs that allow the network load to be distributed to many sensors by delivering different data sensed from each target to the sink through different routes. As a result, our study ensured that a directional sensor network can guarantee k-coverage and 1-connectivity while being energy efficient. As part of our future work, we will extend our study such that maximal lifetime scheduling algorithms guarantee both k-coverage and k-connectivity.

Acknowledgements This research was supported by the KCC (Korea Communications Commission), Korea, under the IT R&D support program supervised by the KCA (KCA-2011-10913-05004) and also supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0003896).

References

1. Ai J, Abouzeid AA (2006) Coverage by directional sensors in randomly deployed wireless sensor networks. J Comb Optim 11:21–41
2. Akyildiz IF, Melodia T, Chowdhury K (2007) A survey on wireless multimedia sensor networks. Comput Netw 51:921–960
3. Cai Y, Lou W, Li M, Li XY (2009) Energy efficient target-oriented scheduling in directional sensor networks. IEEE Trans Comput 58:1259–1274
4. Cardei M, Du DZ (2005) Improving wireless sensor network lifetime through power aware organization. Wirel Netw 11(3):333–340
5. Cardei M, Wu J (2006) Energy-efficient coverage problems in wireless ad-hoc sensor networks. Comput Commun 29(4):413–420
6. Cardei M, Thai MT, Li Y, Wu W (2005) Energy-efficient target coverage in wireless sensor networks. In: IEEE Infocom, pp 1976–1984
7. Chang JH, Tassiulas L (2004) Maximum lifetime routing in wireless sensor networks. IEEE/ACM Trans Netw 12(4):609–619
8. Chen J, Koutsoukos X (2007) Survey on coverage problems in wireless ad hoc sensor networks. In: IEEE SouthEastCon
9. Chen J, Salim MB, Matsumoto M (2011) A single mobile target tracking in Voronoi-based clustered wireless sensor network. J Inf Process Syst 7(1):17–28
10. Hall JL (2001) Handbook of multisensor data fusion. CRC Press, Boca Raton
11. Huang CF, Tseng YC (2005) A survey of solutions to the coverage problems in wireless sensor networks. J Internet Technol 6(1):1–8
12. Ilyas M, Mahgoub I (2004) Handbook of sensor networks: compact wireless and wired sensing systems. Florida Atlantic University, Boca Raton
13. Li D, Wong K, Hu Y, Sayeed A (2002) Detection, classification, tracking of targets. IEEE Signal Process Mag 19(2):17–29
14. Liu H, Wan PJ, Jia X (2006) Maximal lifetime scheduling for sensor surveillance systems with K sensors to 1 target. IEEE Trans Parallel Distrib Syst 17(12):1526–1536
15. Liu H, Jia X, Wan PJ, Yi CW, Makki SK, Pissinou N (2007) Maximizing lifetime of sensor surveillance systems. IEEE/ACM Trans Netw 15(2):334–345
16. Liu K, Xiong J (2010) A fine-grained localization scheme using a mobile beacon node for wireless sensor networks. J Inf Process Syst 6(2):147–162
17. Lu M, Wu J, Cardei M, Li M (2005) Energy-efficient connected coverage of discrete targets in wireless sensor networks. In: Int conf of computer networks and mobile computing (ICCNMC)
18. Ma H, Liu Y (2005) Correlation based video processing in video sensor networks. IEEE Wirel Commun 987–992. doi:10.1109/WIRLES.2005.1549547
19. Ma H, Liu Y (2005) On coverage problems of directional sensor networks. In: The first int conf on mobile ad-hoc and sensor networks, pp 721–731
20. Ma H, Tao D (2006) Multimedia sensor network and its research progresses. J Softw 17:2013–2028
21. Rahimi M, Baer R, Iroezi O, Garcia J, Warrior J, Estrin D, Srivastava M (2005) Cyclops: in situ image sensing and interpretation in wireless sensor networks. In: Proc ACM conf embedded networked sensor systems
22. Wang J, Niu C, Shen R (2009) Priority-based target coverage in directional sensor networks using a genetic algorithm. Comput Math Appl 57:1915–1922
23. Wang X, Xing G, Zhang Y, Lu C, Pless R, Gill C (2003) Integrated coverage and connectivity configuration in wireless sensor networks. In: Proc ACM int conf embedded networked sensor systems, pp 28–39
24. Zhao Q, Gurusamy M (2007) Lifetime maximization for connected target coverage in wireless sensor networks. IEEE/ACM Trans Netw 16(6). doi:10.1109/TNET.2007.911432
25. Zhou G, He T, Krishnamurthy S, Stankovic JA (2006) Models and solutions for radio irregularity. Wirel Sens Netw 2(2):221–262