

Article

## On Connected Target Coverage for Wireless Heterogeneous Sensor Networks with Multiple Sensing Units

Kuei-Ping Shih <sup>1</sup>, Der-Jiunn Deng <sup>2</sup>, Ruay-Shiung Chang <sup>3,\*</sup>, and Hung-Chang Chen <sup>4</sup>

<sup>1</sup> Department of Computer Science & Information Engineering, Tamkang University, Tamshui 251, Taipei, Taiwan; E-Mail: kpshih@mail.tku.edu.tw

<sup>2</sup> Department of Computer Science & Information Engineering, National Changhua University of Education, Changhua, Taiwan; E-Mail: djdeng@cc.ncue.edu.tw

<sup>3</sup> Department of Computer Science & Information Engineering, National Dong Hwa University, Hualien, Taiwan

<sup>4</sup> Department of Information Technology, Ching Kuo Institute of Management and Health, Keelung, Taiwan; E-Mail: gileschen@ems.cku.edu.tw

\* Author to whom correspondence should be addressed; E-Mail: rschang@mail.ndhu.edu.tw; Tel.: +886-3-8632031; Fax: +886-3-8632030

Received: 2 May 2009; in revised form: 14 June 2009 / Accepted: 23 June 2009 /

Published: 30 June 2009

**Abstract:** The paper considers the connected target coverage (CTC) problem in wireless heterogeneous sensor networks (WHSNs) with multiple sensing units, termed MU-CTC problem. MU-CTC problem can be reduced to a connected set cover problem and further formulated as an integer linear programming (ILP) problem. However, the ILP problem is an NP-complete problem. Therefore, two distributed heuristic schemes, REFS (remaining energy first scheme) and EEFS (energy efficiency first scheme), are proposed. In REFS, each sensor considers its remaining energy and its neighbors' decisions to enable its sensing units and communication unit such that all targets can be covered for the required attributes and the sensed data can be delivered to the sink. The advantages of REFS are its simplicity and reduced communication overhead. However, to utilize sensors' energy efficiently, EEFS is proposed. A sensor in EEFS considers its contribution to the coverage and the connectivity to make a better decision. To our best knowledge, this paper is the first to consider target coverage and connectivity jointly for WHSNs with multiple sensing units. Simulation results show that REFS and EEFS can both prolong the network lifetime effectively. EEFS outperforms REFS in network lifetime, but REFS is simpler.

**Keywords:** connectivity; heterogeneous sensors; multiple sensing units; target coverage; wireless sensor networks

## 1. Introduction

With the advancements of Micro-Electro-Mechanical System (MEMS), digital electronics, as well as wireless communication technology, a kind of small-size, low-cost, and low-power device with sensing, processing and wireless transmission capabilities, called *sensor*, is widely developed and deployed in a variety of applications. A wireless sensor network (WSN) is an auto-configured network consisted of many sensors deployed in a sensing field in an ad hoc or prearranged fashion. The purposes of WSNs include sensing, monitoring, or tracking environmental events. WSNs have been widely used in battlefield surveillance, environmental monitoring, biological detection, home automation, industrial diagnostics, etc. [1].

A wireless heterogeneous sensor network (WHSN) is a sub-class of wireless sensor networks in which each sensor may have different capabilities, such as various transmission capabilities, different number of sensing units, etc. [2, 3]. In the paper, a WHSN with multiple sensing units is considered, which means each sensor in the WHSN may be equipped with more than one sensing unit, and the *attribute* that each sensing unit can sense may also be different. In fact, sensors equipped with multiple sensing units are very common in many commercial products. For example, each MICA2 mote [4] is equipped with several sensing units for temperature, humidity, light, sound, vibration, etc. A WHSN with multiple sensing units is inherently formed in nature because some sensing units in a sensor may be malfunctioned after running for a long time. The remaining sensing units on each sensor may be different. As a result, how to utilize the sensors with the remaining sensing units efficiently to continue the original sensing task is a very important concern.

Furthermore, using a WHSN with multiple sensing units is also cost-effective and power-efficient if multiple attributes are required to be sensed in the sensing field. On one hand, in addition to the sensing unit, a sensor, in general, consists of a control unit, a power unit, a radio unit, etc. If a sensor is equipped with only one sensing unit, it will increase the cost substantially to deploy all kinds of sensors to sense all required attributes. On the other hand, if too many sensing units are equipped in a sensor, the sensor will quickly run out of energy. Therefore, a WHSN with multiple sensing units is a promising deployment if multiple attributes are required to be sensed in the sensing field [2, 3]. Moreover, it is very likely that several different kinds of sensors have been deployed in the sensing field for different purposes. These sensors can collaborate for additional sensing purposes to increase the sensor utilization.

*Coverage* and *connectivity* are two key factors to a successful WSN. In general, coverage problems either deploy sensors to cover the sensing field completely [5, 6], or make sure that all the sensing field is covered by a certain amount of sensors, such as 1-coverage or  $k$ -coverage [7, 8], or select active sensors in a densely deployed sensor networks to cover all the sensing field [9–12]. On the other hand, the connectivity issue emphasizes how well sensors connect to the sink and if the sensed data can be properly delivered to the sink. The connected target coverage (CTC) problem is one of the target coverage (TC)

problems, but also takes the connectivity issue into consideration simultaneously. In this paper, CTC problem in a WSN with multiple sensing units is termed MU-CTC problem (where MU means multiple sensing units) and is defined as below.

**Definition 1 (MU-CTC Problem)** *Given a set of targets (or points) of interest and a number of sensors with multiple sensing units randomly deployed in the sensing field, MU-CTC problem is to schedule the on/off of the sensing units as well as the communication unit on each sensor such that (1) the attributes required to be sensed at each target can be sensed at all time, (2) the sensed data can be delivered to the sink, and (3) the network lifetime is maximized.*

The network lifetime is defined as the time interval from the beginning to the time that either the condition (1) or (2) above is not satisfied.

MU-CTC problem can be represented by a bipartite graph and be reduced to a connected set cover problem, named MU-CSC (Multiple sensing Units for Connected Set Cover) problem. The MU-CSC problem can be formulated as an integer linear programming (ILP) problem and solved by an ILP solver. However, solving the ILP problem is NP-complete [13]. Therefore, two distributed schemes, named REFS (remaining energy first scheme) and EEFS (energy efficiency first scheme), are proposed to deal with the MU-CTC problem. In REFS, a sensor enables its sensing and communication units based on its remaining energy and its neighbors' decisions. The advantages of REFS are its simplicity and reduced communication overhead. However, redundant sensing is the most significant weakness of REFS.

Generally in the CTC problem, a sensor not only undertakes the sensing task, but also needs to relay the sensed data for others. Therefore, to make the best use of a sensor's energy, target coverage and sensed data relay should be considered simultaneously. Consequently, EEFS is proposed, where a sensor enables its sensing and communication units by considering not only its target coverage but also its relay role. As a result, the network lifetime of EEFS can be prolonged accordingly. Simulation results also verify that EEFS outperforms REFS in network lifetime. In addition, to our best knowledge, this is the first paper to discuss such a problem in the literature.

The rest of the paper is organized as follows. Section 2 describes the related work that treated the CTC problem with different network models and assumptions. Section 3 formulates MU-CTC problem as an ILP problem. Section 4, two distributed schemes, REFS and EEFS, are proposed to deal with the MU-CTC problem. Simulation results are presented in Section 5. Section 6 concludes the paper.

## 2. Related Work

Coverage problem is of critical importance for wireless sensor networks. As described above, the coverage problem either deploys sensors to cover the sensing field, or selects active sensors and schedule them to cover all the sensing field, or analyzes how well the sensing field is covered. Recently, the target coverage problem has received extensive attention [13–20]. Nevertheless, none of them considered sensors equipped with multiple sensing units. Moreover, some of them do not take the connectivity issue into account. These researches are summarized as follows.

In [13], the authors transformed the TC problem into a *Maximal Set Cover* (MSC) problem, where sensors are organized into set covers. To maximize the number of set covers is equivalent to maximizing the network lifetime. Every set cover is activated in turn and the sensors in an activated set cover are

responsible for sensing all the targets at a specific time, while all the other sensors are in the sleep state. The MSC problem was proved to be an NP-complete problem and two heuristics were proposed to solve the MSC problem using linear programming and greedy techniques, respectively. However, these schemes are centralized. Similar to [13] except for fixed sensing range, the TC problem with adjustable sensing range is addressed in [14], where the goals of the problem are to schedule sensors to alternate between the active and the sleep states and adjust their sensing ranges so that all targets are covered by active sensors and the network lifetime is maximized. The authors transformed the problem to an *adjustable range set covers* (AR-SC) problem and formulated it by ILP constraints. The authors then solved it using relaxation and rounding techniques. A greedy heuristic is proposed, where both centralized and distributed (localized) solutions are given for computing the set covers. However, connectivity is not considered in [13, 14].

In [15], the TC problem is considered. The proposed approach consists the following three steps. A linear programming method is used to compute the maximum network lifetime. A workload matrix was used to specify the total length of time that a sensor should watch a target. The workload matrix is further decomposed into a sequence of schedule matrices by using the perfect matching method. Finally, the target watching timetable is obtained for each sensor based on the schedule matrices. A similar approach is also used in [16, 17], where the connectivity issue is jointly considered with the TC problem. In [16],  $k$ -coverage is additionally taken into account. That is, given  $k$ , it requires each target to be covered by at least  $k$  sensors and those active sensors to be connected. However, the proposed approaches in [15–17] are centralized. Moreover, they also assume that a sensor can only monitor at most one target at a time, which simplifies the difficulty in dealing with the problem.

Similarly, the CTC problem is also considered in [18]. The authors model the CTC problem as a Maximum Cover Tree (MCT) problem, where the sensors in a cover tree can cover all the targets and relay the sensed data to the sink. Based on the MCT problem, the upper bound of the CTC problem in terms of lifetime is derived by using a linear programming model. The Communication Weighted Greedy Cover (CWGC) algorithm is further proposed to construct the cover trees. However, the CWGC algorithm is centralized and practically hard to implement.

In [19], the CTC problem with  $k$ -coverage is considered. Two non-global schemes, cluster-based and pruning-based, are proposed in the paper. The cluster-based scheme selects the backbone sensors to form a  $k$ -connected coverage set. In the pruning-based scheme, each sensor determines its status (marked or unmarked) based on its two-hop neighborhood information. All marked sensors form a  $k$ -connected coverage set. In [20], the CTC problem in a WHSN is considered, where the WHSN means that the network consists of two types of sensors. One is the resource-rich sensors called supernodes used for data relaying and the other is the energy constrained sensors. In the paper, supernodes are assumed to have two transceivers, one for communication with sensors and the other for communication with other supernodes. All supernodes form a connected network and the active sensor connects to at least one supernode, via either a direct or a multi-hop connection. The problem is transformed to a heterogeneous connected set covers (HCSC) problem and the HCSC problem is proved to be NP-complete. An ILP approach as well as a distributed and localized approach are proposed in the paper.

Table 1 briefly summarizes the related work. Although there are a lot of related work in the literature that deal with TC or CTC problem, all of the above works [13–20] only considered each sensor to be

equipped with one sensing unit. As described earlier, a WSN with multiple sensing units is a very common, useful and important. Thus, the TC or CTC problem deserve receiving more attention toward a WSN with multiple sensing units. In our previous research [21], the TC problem in a WSN with multiple sensing units was investigated, but the connectivity issue was not taken into account. This paper now discusses the CTC problem in a WSN with multiple sensing units. The CTC problem aiming at heterogeneous sensors with multiple sensing units turns out to be more complicated than those focusing on homogeneous sensors with only one sensing unit. The reasons are as follows. In WSNs with multiple sensing units, the CTC problem needs to consider not only which sensors need to be activated, but also which sensing units on those sensors need to be activated. Moreover, the sensing attributes that need to be covered at each target are different. In addition, the connectivity issue is also considered in the paper. In [13], the maximal set cover problem considering a sensor with single sensing unit has been proven to be an NP-complete problem. The CTC problem in WSNs with multiple sensing units is a superset of that with only one sensing unit. Thus, the CTC problem in a WSN with multiple sensing units is also an NP-complete problem.

**Table 1.** Related work.

Scheme	TC / CTC	Distributed	Sensing Unit
[13]	TC	No	Single
[14]	TC	Yes	Single
[15]	TC	No	Single
[16]	CTC	No	Single
[17]	CTC	No	Single
[18]	CTC	No	Single
[19]	CTC	Yes	Single
[20]	CTC	Yes	Single
[21]	TC	Yes	Multiple

As a result, two heuristic schemes are proposed in the paper to schedule the sensors' sensing units as well as the communication unit such that the individually required attributes of a given set of targets can be covered all the time, the sensed data can be relayed to the sink, and the network lifetime can be maximized.

### 3. Problem Statements and Formulations

#### 3.1. Assumptions

In a sensing field  $\mathcal{A}$ , there are  $M$  stationary targets at known locations to be continuously sensed. Let  $t_m$  denote the  $m$ th target,  $m = 1, 2, \dots, M$ . Suppose there are  $L$  attributes, denoted  $a^l, l = 1, 2, \dots, L$ , to be sensed for these targets. The attributes to be sensed on each target are not altogether the same. Attribute  $a^l$  can be sensed by the sensing unit  $u^l, l = 1, 2, \dots, L$ . The energy consumption of the sensing unit  $u^l$  in sensing attribute  $a^l$  for a time unit is  $e^l, l = 1, 2, \dots, L$ . Suppose there are  $N$  stationary sensors equipped with different numbers of sensing units randomly deployed in the sensing field. Let  $s_n$  denote the  $n$ th sensor,  $n = 1, 2, \dots, N$ . The initial energy of each sensor is  $E$  and assumed the same. All the sensed data are required to be forwarded to the sink. Without loss of generality, let  $s_0$  denote the sink for

easy description.

In this paper, each sensor knows its location and can obtain one-hop neighbor information via communication. Moreover, the locations of the targets to be sensed are known by the sensors in advance and will not change during the whole sensing period. In reality, the sensing ranges of different kinds of sensing should be different. However, the sensing range of each sensing unit as well as the communication range of each sensor are all assumed to be the same and unadjustable. The assumption can be easily relaxed as follows. Knowing the equipped sensing units as well as the location information of its one-hop neighbors and itself, each sensor can determine the attributes that can be sensed by itself or by its one-hop neighbors via communication. Therefore, the proposed protocols are expected to work well even when the assumption is relaxed.

Let  $R_s$  denote the sensing range of a sensing unit and  $R_c$  the communication range of a sensor. For simplicity, the paper also assumes that  $R_c \geq 2R_s$ . It is to make sure that two sensors can communicate directly if their sensing ranges overlap. If  $R_s > \frac{1}{2}R_c$ , for any sensor, the information of whether a required sensing attribute at a target can be sensed by its neighbors has to be collected through multi-hop communication. However, how to acquire the information through multi-hop communication is not discussed here and will be of future work.

Target  $t_m$  is said to be monitored by  $u^l$  of  $s_n$ , if  $d(t_m, s_n) \leq R_s$  and  $s_n$  is equipped with  $u^l$ , where  $d(x, y)$  is the Euclidean distance between  $x$  and  $y$ . Two sensors, say  $s_n$  and  $s_{n'}$ , can communicate if  $d(s_n, s_{n'}) \leq R_c$ . Namely, they are neighbors.

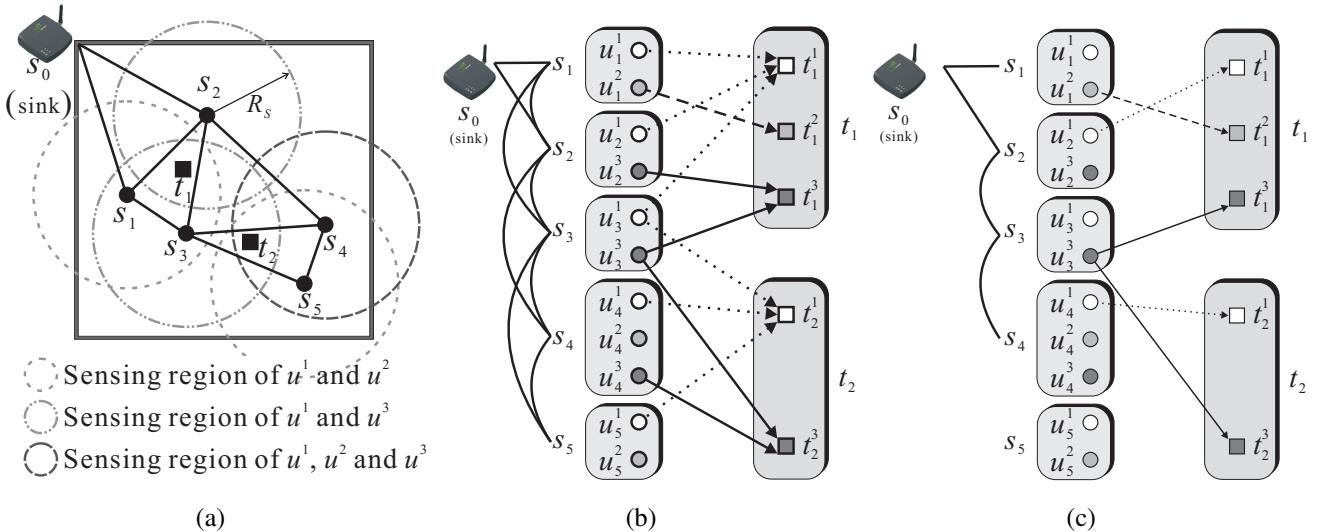
The MU-CTC problem can be represented by a quasi-bipartite graph and be further formulated as a connected set cover problem, named **M**ultiple **S**ensing **U**nits for **C**onnected **S**et **C**over (MU-CSC) problem. Take Figure 1 as an example, where five sensors,  $s_1, s_2, \dots, s_5$ , are required to sense two targets,  $t_1$  and  $t_2$ . The attributes required to be sensed at  $t_1$  are  $a^1, a^2$ , and  $a^3$  and those at  $t_2$  are  $a^1$  and  $a^3$ . As shown in Figure 1(a),  $t_1$  is located within the sensing ranges of  $s_1, s_2$ , and  $s_3$ , and  $t_2$  is within the sensing ranges of  $s_3, s_4$ , and  $s_5$ . The sensing units equipped by  $s_1, s_2, s_3, s_4$ , and  $s_5$  are  $\{u^1, u^2\}$ ,  $\{u^1, u^3\}$ ,  $\{u^1, u^3\}$ ,  $\{u^1, u^2, u^3\}$ , and  $\{u^1, u^2\}$ , respectively. Figure 1(a) illustrates the network topology, where different circles represent the different sensing capabilities of sensors and the black solid lines represent the connectivity among sensors.

### 3.2. The MU-CSC Problem

Let  $u_n^l$  denote the sensing unit  $u^l$  on sensor  $s_n$ , if sensor  $s_n$  is equipped with sensing unit  $u^l$ , for  $n = 1, 2, \dots, N$  and  $l = 1, 2, \dots, L$ . Moreover, let  $t_m^l$  stand for the attribute  $a^l$  required to be sensed at target  $t_m$ ,  $m = 1, 2, \dots, M$  and  $l = 1, 2, \dots, L$ . The MU-CTC problem can be represented by a quasi-bipartite graph, as illustrated in Figure 1(b). There exists a ray from  $u_n^l$  to  $t_m^l$  if the sensing unit  $u^l$  on sensor  $s_n$  can sense the attribute  $a^l$  at target  $t_m$ . In Figure 1, the different types of rays mean different types of sensing units. In addition, the black solid lines between two sensors mean that these sensors can communicate with each other directly.

**Figure 1.** An illustrated example, where five sensors are scheduled to cover two targets.

(a) The topology. (b) The coverage and connectivity relationships. (c) An illustration of a connected set cover.



Consequently, the MU-CTC problem can be regarded as a maximal connected set cover problem. Sensors are organized as connected set covers. In each connected set cover, the sensors turn on some sensing units to cover the required attributes of the targets, or turn on their communication units to relay the sensed data to the sink. Figure 1(c) is an example of a connected set cover for the example shown in Figure 1(a). The set  $\{u_1^2, u_2^1, u_3^3, u_4^1\}$  is a connected set cover, where  $u_1^2$  covers  $t_1^2$ ,  $u_2^1$  covers  $t_1^1$ ,  $u_3^3$  covers  $t_1^3$  and  $t_2^3$ , and  $u_4^1$  covers  $t_2^1$ . In addition,  $s_1$ ,  $s_2$ ,  $s_3$ , and  $s_4$  can relay the sensed data to the sink via multi-hop transmission. The MU-CSC problem is defined as follows.

**Definition 2 (MU-CSC Problem)** Given  $M$  targets and  $N$  sensors with multiple sensing units, the MU-CSC problem is to find a family of connected set covers  $C_1, C_2, \dots, C_\kappa$ , so that (1) the required attributes at every target can be completely covered by each connected set cover, (2) all sensors in a connected set cover can connect to the sink directly or via the sensors in the same connected set cover, (3) the energy consumption of each sensor in all connected set covers is at most  $E$ , the initial energy, (4)  $\kappa$  is maximized.

Notice that, each connected set cover corresponds to a working period, say a *round* (to be described later). Thus, maximizing  $\kappa$  is equal to maximizing the network lifetime.

### 3.3. ILP Constraints for the MU-CSC Problem

In modeling ILP, some assumptions are made to simplify the formulations. Assume that a sensing unit on an active sensor periodically generates a sensed data in a round, where an active sensor is the sensor selected to undertake the sensing task. Data aggregation is not considered in the model. That is, if a sensor receives  $f$  sensed data, it needs to relay the  $f$  data to the sink, or transmit  $f+1$  data out if its data is counted. Without loss of generality, the index variables listed below are used for their corresponding meanings, if not otherwise specified.

- $m$ :  $m$ th target, where  $1 \leq m \leq M$ ,
- $n$ :  $n$ th sensor, where  $1 \leq n \leq N$ , and
- $l$ :  $l$ th attributes, sensing unit, or energy consumption of the  $l$ th sensing unit, where  $1 \leq l \leq L$ .

By Definition 2, the MU-CSC problem can be modeled by the ILP constraints as follows.

### ILP Constraints for the MU-CSC Problem

*Given:*

- $M$  targets:  $t_m, m = 1, 2, \dots, M$ ,
- $N$  sensors:  $s_n, n = 1, 2, \dots, N$ ,
- The sink:  $s_0$ ,
- Initial energy of each sensor:  $E$ ,
- $L$  sensing attributes:  $a^l, l = 1, 2, \dots, L$ ,
- $L$  sensing units:  $u^l$ , which respectively senses the attribute  $a^l$ , and the energy consumed is:  $e^l, l = 1, 2, \dots, L$ ,
- $e_t$ : the energy consumption of a sensor to transmit a sensed data,
- $e_r$ : the energy consumption of a sensor to receive a data,
- $\mathbb{N}(n)$ : the neighbor set of  $s_n$  and is defined as  $\{n' | d(s_n, s_{n'}) \leq R_c, n' = 0, 1, \dots, N, n' \neq n\}$ , where  $d(s_n, s_{n'})$  is the Euclidean distance between  $s_n$  and  $s_{n'}$ ,
- $\nu_n^l$ : a coefficient that indicates whether  $s_n$  is equipped with  $u^l$ ;  $\nu_n^l = 1$  if  $s_n$  is equipped with  $u^l$ ; otherwise,  $\nu_n^l = 0, n = 1, 2, \dots, N$ , and  $l = 1, 2, \dots, L$ ,
- $\mu_{n,m}$ : a coefficient that indicates whether  $s_n$  can cover  $t_m$ ;  $\mu_{n,m} = 1$  if  $d(s_n, t_m) \leq R_s$ ; otherwise,  $\mu_{n,m} = 0, n = 1, 2, \dots, N$ , and  $m = 1, 2, \dots, M$ ,
- $\tau_m^l$ : a coefficient that indicates whether  $t_m$  needs to be covered by  $a^l$ ;  $\tau_m^l = 1$  if  $t_m$  needs to be covered by  $a^l$ ; otherwise,  $\tau_m^l = 0, m = 1, 2, \dots, M$ , and  $l = 1, 2, \dots, L$ .

*Variables:*

- $c_k$ : a boolean variable;  $c_k = 1$  if  $C_k$  is a connected set cover; otherwise,  $c_k = 0, \forall k = 1, 2, \dots, K$ , where  $K$  is an upper bound of the number of connected set covers,
- $\hat{\nu}_{n,k}^l$ : a boolean variable;  $\hat{\nu}_{n,k}^l = 1$  if sensor  $s_n$  enables the sensing unit  $u^l$  in set  $C_k$ ; otherwise,  $\hat{\nu}_{n,k}^l = 0, \forall n, l, k$ ,
- $f_{n \rightarrow n'}^k$ : a non-negative integer that indicates the number of sensed data relayed from  $s_n$  to  $s'_n$  in set  $C_k$ .

*Objective:* Maximize  $c_1 + c_2 + \dots + c_K$ .

*Subject to:*

$$\mathbf{(C1)} \sum_{n=1}^N (\mu_{n,m} * \nu_n^l * \hat{\nu}_{n,k}^l) \geq c_k * \tau_m^l, \forall m, l, k,$$

$$\mathbf{(C2)} \sum_{k=1}^K \left( \sum_{l=1}^L (\hat{\nu}_{n,k}^l * e^l) + e_t \sum_{n' \in \mathbb{N}(n)} f_{n \rightarrow n'}^k + e_r \sum_{n' \in \mathbb{N}(n)} f_{n' \rightarrow n}^k \right) \leq E, \forall n,$$

$$\mathbf{(C3)} \sum_{n' \in \mathbb{N}(n)} f_{n \rightarrow n'}^k - \sum_{n' \in \mathbb{N}(n)} f_{n' \rightarrow n}^k = \sum_{l=1}^L \hat{\nu}_{n,k}^l, \forall n, k,$$

$$\mathbf{(C4)} \sum_{n \in \mathbb{N}(0)} f_{n \rightarrow 0}^k = \sum_{n=1}^N \sum_{l=1}^L \hat{\nu}_{n,k}^l, \forall k,$$

$$\mathbf{(C5)} \hat{\nu}_{n,k}^l \leq c_k, \forall n, l, k,$$

$$\mathbf{(C6)} 0 \leq f_{n \rightarrow n'}^k \leq N * L, \forall n, n', k,$$

$$\mathbf{(C7)} \hat{\nu}_{n,k}^l \in \{0, 1\}, \forall n, l, k,$$

$$\mathbf{(C8)} c_k \in \{0, 1\}, \forall k.$$

Remarks:

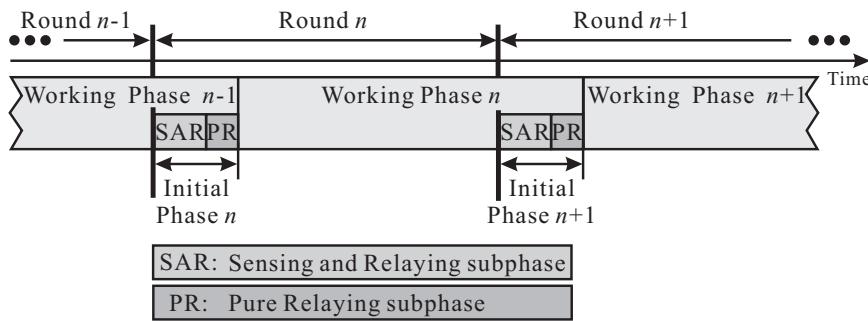
- Constraint **C1** corresponds to the first requirement in Definition 2. The constraint guarantees that the required attributes at every target can be completely covered by any selected connected set cover.
- Constraint **C2** corresponds to the third requirement in Definition 2. The constraint indicates that the total energy consumption of a sensor should not exceed its initial energy  $E$ .
- Constraint **C3** is to ensure that the generated and the received data of a sensor in a round can be forwarded by its neighbors.
- Constraint **C4** is used to make sure that the amount of the sensed data generated by the sensors belonging to a connected set cover is equal to that received by the sink in a round. Therefore, with constraints **C3** and **C4**, it can be ensured that the sensed data of a sensor can be relayed to the sink, that is, the second requirement in Definition 2.
- Constraint **C5** is to require a sensor to not turn on its sensing units if the sensor does not belong to a connected set cover. It helps to further reduce the energy consumption of a sensor.
- Constraints **C6-C8** indicate the ranges of these variables.

Note that, it is well-known that solving ILP is an NP-complete problem[13]. Therefore, two distributed schemes to solve the MU-CTC problem are proposed.

#### 4. Distributed Schemes for the MU-CTC Problem

It is well-known that solving ILP is an NP-complete problem. Therefore, two distributed schemes, named REFS (remaining energy first scheme) and EEFS (energy efficiency first scheme), are proposed to solve the MU-CTC problem. As shown in Figure 2, time is divided into rounds of equal length. A round consists of an *initial phase* and a *working phase*. The initial phase is further divided into a *sensing and relaying (SAR)* subphase and a *pure relaying (PR)* subphase. Let  $\mathcal{D}_{Init}$ ,  $\mathcal{D}_{SAR}$ , and  $\mathcal{D}_{PR}$  denote the durations of the initial phase, the SAR subphase and the PR subphase, respectively. Clearly,  $\mathcal{D}_{Init} = \mathcal{D}_{SAR} + \mathcal{D}_{PR}$ . Note that  $\mathcal{D}_{Init}$  is much shorter than the duration of a round.

**Figure 2.** Time structure.



During  $\mathcal{D}_{SAR}$ , each sensor will determine which of its sensing units should be turned on. If the sensor needs to turn on some sensing units, it still needs to find an appropriate neighbor to relay the sensed data. Similarly, the relay node also needs to find its relay node to continue relaying the sensed data to the sink. Basically, whether a sensor needs to be activated and which of its sensing units need to be turned on will be decided during  $\mathcal{D}_{SAR}$ . Consequently,  $\mathcal{D}_{PR}$  is only for the sensor chosen as the relay to find its relay to the sink. The working phase begins at the end of the initial phase and ends at the end of the initial phase of the next round so that the targets can be continually monitored. In addition, in both REFS and EEFS, each sensor makes the decision only by one-hop neighbor information, such as the location, the remaining energy, and the sensing capability of neighbors, as well as the requests for relaying.

##### 4.1. A Generic Approach to the MU-CTC Problem

Algorithm 1 is a generic approach to the MU-CTC problem, which is performed by each sensor during every initial phase in a distributed fashion. Initially, in Step 1, each sensor will set a waiting time, say  $W_n$  for sensor  $s_n$ , in order to receive the neighbors's decision and then make its own decision after  $W_n$  expires. It is worth mentioning that  $W_n$  heavily impacts the performance of the proposed schemes. As a result, in designing  $W_n$ , REFS takes the sensor's remaining energy into account so that the sensor with more remaining energy can make the decision sooner to take the coverage burden. On the other hand, in EEFS, both the coverage and connectivity are taken into account to efficiently utilize the sensing and communication units of its neighbors' and its own.

**Algorithm 1:** A generic approach to the MU-CTC problem

---

```

/* To be performed by each sensor, say  $s_n$ , in a distributed fashion at each initial phase. */  

begin  

    Step 1: Set  $W_n$ ;  

    Step 2: Countdown  $W_n$  until  $W_n$  expires;  

        Step 2.1 Overhear neighbors' decisions;  

        Step 2.2 Remove the sensing responsibilities that have been taken by the neighbors;  

    Step 3: Remove incapable or redundant sensing responsibilities;  

    Step 4: Select the relay;  

    Step 5: Broadcast my decision;  

end

```

---

In Step 2, each sensor counts down  $W_n$  and waits for the neighbors' decision until  $W_n$  expires. After  $W_n$  expires, the sensor can decide which sensing and communication units need to be turned on. Therefore, in Step 3, some strategies are adopted to remove the redundant sensing responsibilities or to make the sensing responsibilities more efficiently performed. In Step 4, the relay selection is performed to find an appropriate sensor to relay the sensed data. Finally, in Step 5, the decision, including which sensing units need to be turned on, whether the communication unit needs to be turned on, and which neighbor is selected as the relay, is announced to its neighbors. Based on the generic approach shown in Algorithm 1, the following two subsections describe the two distributed algorithms, REFS and EEFS, in detail. Notice that the proposed algorithms do not require all sensor to be accurately synchronized. Each sensor only needs to be synchronized with its neighbors. The local synchronization can be achieved through the decision announcement in Step 5.

#### 4.2. Remaining Energy First Scheme (REFS)

REFS is a self-pruning approach, which takes the sensor's remaining energy and neighbors' decisions into account to enable its sensing and communication units. Based on Algorithm 1, the details of REFS are illustrated as follows.

Set  $W_n$

Let  $\delta_n^l$  be the sensing capability of the sensing unit  $u^l$  on sensor  $s_n$  and be defined as  $\delta_n^l = \{t_m^l | \mu_{n,m} * \nu_n^l * \tau_m^l = 1, \forall m\}$ . Furthermore, let  $\Delta_n$  be the sensing capability of sensor  $s_n$ , which is the union of the sensing capabilities of all sensing units equipped on  $s_n$ . That is,  $\Delta_n = \bigcup_{\forall \nu_n^l=1} \delta_n^l$ . For the example shown in Figure 1,  $\delta_1^2 = \{t_1^2\}$  and  $\delta_3^1 = \{t_1^1, t_2^1\}$ . In addition,  $\Delta_1 = \delta_1^1 \cup \delta_1^2 = \{t_1^1, t_1^2\}$  and  $\Delta_3 = \delta_3^1 \cup \delta_3^3 = \{t_1^1, t_2^1, t_1^3, t_2^3\}$ . Initially, a sensor, say  $s_n$ , will take  $\Delta_n$  as its sensing responsibility. Let  $\Gamma_n$  denote the *sensing responsibility* of  $s_n$ . Thus,  $\Gamma_n$  is initialized to  $\Delta_n$ .

In REFS, the setting of  $W_n$  solely depends on the remaining energy of the sensor. The more the energy remains, the shorter is the waiting time. As a result, the sensor with more energy will turn on more sensing units to sense the attributes of the targets. Let  $E_n^r$  stand for the remaining energy of  $s_n$ .  $W_n$  is set as follows:

$$W_n = \left(1 - \frac{E_n^r}{E}\right) * \mathcal{D}_{SAR}.$$

### Countdown $W_n$

At the beginning of every initial phase, each sensor, say  $s_n$ , will wait for a waiting time of its own ( $W_n$ ) and overhear the decisions of the neighbors with smaller waiting time. While receiving the neighbors' decision packets, the sensor will prune away from  $\Gamma_n$  those  $t_m^l$  indicated in the neighbors' decision packets. Note that, in the neighbors' decision packets, the next relay is also included. If  $s_n$  is indicated as a relay,  $s_n$  should turn on its communication unit and also need to find the next relay for itself in order to relay the sensed data for the neighbors.

### Remove incapable sensing responsibilities

After  $W_n$  expires, the remaining  $\Gamma_n$  is the sensing responsibility of  $s_n$  at this round. However, it is possible that all remaining energy of  $s_n$  is still not enough to support all sensing units indicated in  $\Gamma_n$ . As a result,  $s_n$  will orderly remove the sensing units whose sensing capability is the least. If  $S$  is a set,  $|S|$  means the cardinality of  $S$ . Therefore, the deletion of  $\Gamma_n$  can be formally represented as follows.

$$\Gamma_n = \Gamma_n - \{t_m^{l'}, \forall m \mid l' = \arg \min_{l \in L} |\{t_m^l \in \Gamma_n, \forall m\}|\}.$$

Finally,  $s_n$  enables the corresponding sensing units indicated in the remaining  $\Gamma_n$  to cover the attributes for the indicated targets.

### Select the relay

Upon deciding to turn on the sensing units or being indicated as other's relay,  $s_n$  will select a neighbor  $s_{n'}$  to relay the sensed data to the sink. In general, the sensors closest to the sink are the best. Let  $nextRelay(n)$  denote the relay sensor for  $s_n$ . Therefore, in REFS,  $nextRelay(n)$  is set as below.

$$nextRelay(n) = \arg \min_{n' \in N(n)} d(s_{n'}, s_0).$$

Note that the way to find the relay is the same as geographic routing. However, in geographic routing, a local minimum (dead end) problem will occur. Many approaches have been proposed to handle the problem [22, 23]. Therefore, the paper does not address the problem. In addition, if  $s_n$  does not have enough energy to enable any sensing unit, it will quit executing REFS and turn off all sensing units. The remaining energy of  $s_n$  will be left for communication only.

### Broadcast my decision

Finally,  $s_n$  announces its decision as well as the selected relay.

### Summary

Overall, REFS is a simple scheme that can be easily implemented. Moreover, REFS incurs less control overhead. However, REFS only takes the sensor's remaining energy and the neighbors' decisions into account for making its own decision. The degree of contributions of the sensing units and the communication unit are not considered in REFS. Therefore, the sensor's energy can not be efficiently utilized. As a result, the sensor applying REFS has a high possibility to enable more redundant sensing or communication units. Consequently, the network lifetime can not be prolonged effectively.

#### 4.3. Energy Efficiency First Scheme (EEFS)

Similar to REFS, EEFS is also a self-pruning approach and is operated at every initial phase by each sensor to individually schedule the on/off state of the sensing and communication units for the following working phase. However, in addition to remaining energy, EEFS also takes coverage and connectivity into consideration. EEFS adds more heuristics to prune the redundant or inefficient sensing responsibilities away and select the relay intelligently.

Before executing EEFS, each sensor has to collect its neighbors' sensing capabilities and critical sensing responsibilities in advance in order to make the most efficient use of its sensing and communication units. The *critical sensing responsibility* of a sensor is the attributes of the targets (in terms of  $t_m^l$ ) that can be sensed only by that sensor. Let  $\Theta_n$  be the critical sensing responsibility of  $s_n$ .  $\Theta_n = \{t_m^l | \mu_{n,m} * \nu_n^l * \tau_m^l = 1 \text{ and } \sum_{n' \in \mathbb{N}(n)} \nu_{n'}^l = 0, \forall l, m\}$ . For the example shown in Figure 1,  $\Theta_1 = \{t_1^2\}$ . In other words, each sensor, say  $s_n$ , has to compute  $\Delta_n$  and  $\Theta_n$  as well as collect  $\Delta_{n'}$  and  $\Theta_{n'}, \forall n' \in \mathbb{N}(n)$ , in advance. Nevertheless, the collection needs to be performed only once. Moreover, to balance the energy consumption and efficiently utilize the energy of sensors, each sensor has to exchange  $E_n^r$  with its neighbors at the beginning of EEFS. Based on Algorithm 1, the details of EEFS are described as follows.

Set  $W_n$

In EEFS, sensor  $s_n$  calculates  $W_n$  by the following information: (1) the sensing capabilities, (2) the critical sensing responsibilities, (3) the remaining energy of itself and its neighbors, as well as (4) the probability of being others' relay. According to the sensing capabilities as well as the critical sensing responsibilities of itself and its neighbors, sensor  $s_n$  will rank its sensing priority among its neighbors. The sensor with higher sensing priority and lower probability of being others' relay has shorter  $W_n$  so that the sensor can make its decision sooner. A sensor's sensing priority and the probability of being others' relay are regarded as its *coverage contribution* and *connectivity contribution*, respectively.

Coverage contribution

The coverage contribution of a sensor reflects the ranking of the sensor among its neighbors in regard to the sensing capability. The ranking is processed as follows. If  $s_n$  is equipped with the sensing unit  $u^l$ ,  $s_n$  will sort the sensing capability of  $\delta_{n'}^l$  in an increasing order to a list  $\mathcal{L}^l$  according to  $|\delta_{n'}^l|, \forall n' \in \{n\} \cup \mathbb{N}(n)$ . If the sensing capabilities of two sensors are the same, the higher priority will be assigned to the one with more remaining energy. Otherwise, the sensor with a larger ID wins. Notice that only the sensor equipped with  $u^l$  is included in the ranking process. Let  $r_n^l$  be the order in  $\mathcal{L}^l$ , which represents the priority of the sensing unit  $u^l$  of  $s_n$  among its neighbors. The larger the  $r_n^l$  is, the higher its priority is. It is worth mentioning that the reason to take a rank among the neighbors is to normalize the  $|\delta_{n'}^l|$  for different  $u^l$  on  $s_n$ .

Take  $s_1$  in Figure 1 as an example. Since  $\mathbb{N}(s_1) = \{s_2, s_3\}$ , only  $s_1, s_2$ , and  $s_3$  are taken into account. Suppose  $E = 8$ ,  $E_1^r = 6$ ,  $E_2^r = 8$ , and  $E_3^r = 4$ . Since  $s_1$  is equipped with  $u^1$  and  $u^2$ ,  $r_1^1$  and  $r_1^2$  as well as  $r_{max}^1$  and  $r_{max}^2$  are to be calculated, where  $r_{max}^l = |\mathcal{L}^l|$ . Firstly,  $u^1$  is considered. Obviously,  $|\delta_1^1| = 1$ ,  $|\delta_2^1| = 1$ , and  $|\delta_3^1| = 2$ . Since  $|\delta_1^1| = |\delta_2^1| = 1$ , the remaining energy is taken into account. Therefore,

the priority of  $\delta_1^1$  among  $s_1$  and its neighbors is:  $\delta_1^1 < \delta_2^1 < \delta_3^1$ . That is,  $\mathcal{L}^1 = (\delta_1^1, \delta_2^1, \delta_3^1)$ . Consequently,  $r_1^1 = 1$  and  $r_{max}^1 = 3$ . With regard to  $u^2$ , since  $s_2$  and  $s_3$  are not equipped with  $u^2$ , therefore,  $r_1^2 = 1$  and  $r_{max}^2 = 1$ .

Let  $\rho_n$  be the priority of  $s_n$ , which takes the priorities of all sensing units equipped on  $s_n$  into consideration.  $\rho_n$  is set as:

$$\rho_n = \sum_{l|_{\nu_n^l=1}} \frac{r_n^l}{r_{max}^l}.$$

For the above example,  $\rho_1 = \frac{1}{3} + \frac{1}{1} = \frac{4}{3}$ .

Because the number of sensing units equipped on sensors is different, the coverage contribution of  $s_n$ , denoted  $\bar{\rho}_n$ , is defined as the average priority of the sensing units equipped on  $s_n$  and represented as follows.

$$\bar{\rho}_n = \frac{\rho_n}{\sum_{l|_{\nu_n^l=1}} \nu_n^l} = \frac{\sum_{l|_{\nu_n^l=1}} \frac{r_n^l}{r_{max}^l}}{\sum_{l|_{\nu_n^l=1}} \nu_n^l}. \quad (1)$$

Notice that  $0 < \bar{\rho}_n \leq 1$ . With regard to the example in Figure 1,  $\bar{\rho}_1 = \frac{4}{3}/2 = \frac{2}{3}$ .

### Connectivity contribution

The connectivity contribution of a sensor is defined as the probability of being others' relay. Let  $\sigma_n$  be the connectivity contribution of  $s_n$ .  $\sigma_n$  is determined as follows.

It is hard for a sensor to find the best route to the sink merely by its local information, such as the one-hop neighbor information. Like REFS, the neighbor with the shortest distance to the sink is the best candidate to relay the sensed data for the sensor. Therefore, a sensor closest to the sink has a higher probability to serve as others' relay. Based on the concept, the location of the relay for sensor  $s_n$  can only be located in the *forwarding zone* of  $s_n$ . The definition of the forwarding zone of  $s_n$  is defined as follows, where  $C(s, R)$  denotes a circle centered at  $s$  with the radius  $R$ . Figure 3 (a) also illustrates the forwarding zone of  $s_n$ .

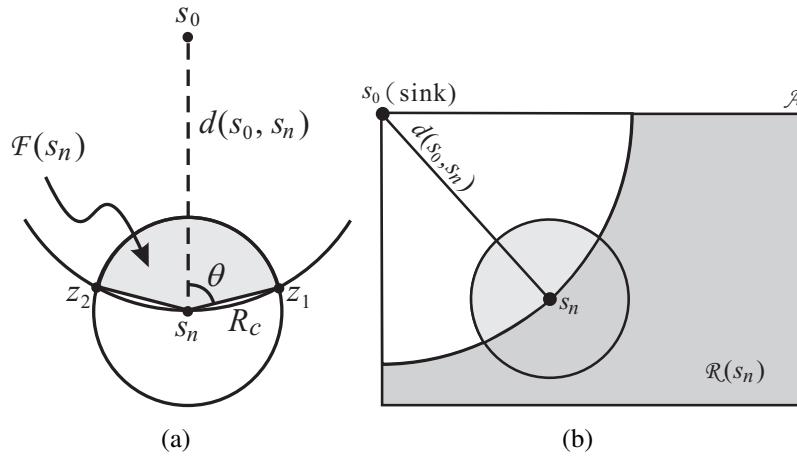
**Definition 3 (Forwarding Zone)** *Let the intersection points of two circles  $C(s_0, d(s_0, s_n))$  and  $C(s_n, R_c)$  be  $z_1$  and  $z_2$ . The forwarding zone of  $s_n$ , denoted  $\mathcal{F}(s_n)$ , is defined as the circular sector formed by two radii  $\overline{s_n z_1}$  and  $\overline{s_n z_2}$ , and the arc  $\widehat{z_1 z_2}$ .*

Compared with  $s_n$ , the sensor located in the forwarding zone of  $s_n$  has a shorter to-sink-distance, which is defined as the distance from the sensor to the sink. On the contrary, the sensor located in the area  $C(s_n, R_c) - \mathcal{F}(s_n)$  has a longer to-sink-distance than  $s_n$ . It implies that the sensor located in  $C(s_n, R_c) - \mathcal{F}(s_n)$  may choose  $s_n$  as its relay. Consequently, the more the sensors are located in  $C(s_n, R_c) - \mathcal{F}(s_n)$ , the more likely  $s_n$  become others' relay. Therefore,

$$\sigma_n = \frac{|\{s_{n'} | s_{n'} \in (C(s_n, R_c) - \mathcal{F}(s_n))\}|}{|\mathbb{N}(n)|}.$$

The higher the  $\sigma_n$  is, the more likely  $s_n$  become others' relay. Similarly, consider the same example in Figure 1. Because  $s_2$  and the sink, regarded as a neighbor of  $s_1$ , are located at  $\mathcal{F}(s_1)$ ,  $\sigma_1 = \frac{3-2}{3} = \frac{1}{3}$ .

**Figure 3.** (a) The forwarding zone of  $s_n$ ,  $\mathcal{F}(s_n)$ . (b) The relaying zone of  $s_n$ ,  $\mathcal{R}(s_n)$ .



The setting of  $W_n$

A sensor with a higher coverage contribution can more efficiently cover the targets, that is, consume less energy but cover more targets. Therefore, the sensor with a higher coverage contribution shall make its decision sooner, implying a shorter  $W_n$ . On the other hand, a sensor with a higher connectivity contribution has a higher probability to relay. If the sensor is selected as a relay, a longer waiting time can let it make its decision later, to be likely to turn on its sensing units to cover the targets, and balance its energy consumption. Consequently, the sensor with a higher connectivity contribution shall wait for a longer time to make its decision.

As a result, a sensor with a higher coverage contribution and a lower connectivity contribution will have a shorter  $W_n$  so that the sensor can determine whether it should turn on its sensing and/or communication units.  $W_n$  is set as follows.

$$W_n = 1 - (\alpha_n * (1 - \sigma_n) + (1 - \alpha_n) * \bar{\rho}_n) * \mathcal{D}_{SAR}, \quad (2)$$

where  $\alpha_n$  is the *contribution tendency* of  $s_n$  on coverage contribution or connectivity contribution. Basically, a sensor near the sink may have a higher probability to forward others the sensed data, because all the sensed data should be forwarded to the sink. The sensor near the sink shall have a higher connectivity tendency to relay the sensed data in order to keep the network connected. On the other hand, a sensor covering more targets shall have a higher coverage contribution tendency to sense the targets. Therefore, each sensor in the sensing field shall have different  $\alpha_n$  to make the best use of its contribution tendency. In other words, in addition to the status of neighbors, the hardware difference and the location of a sensor should be considered to determine  $W_n$ . How to determine each sensors'  $\alpha_n$  will be shown in detail in next subsection.

In order to demonstrate how the coverage and connectivity contributions affects  $W_n$ , assume that the  $\alpha_n$  of the sensors in the example in Figure 1 are the same and set to 0.5. That is, the locations and the energy model of the sensors in Figure 1 is assumed similar. According to Equation (2),  $W_1 = [1 - (\alpha_1 * (1 - \sigma_1) + (1 - \alpha_1) * \bar{\rho}_1)] * \mathcal{D}_{SAR} = [1 - (0.5 * (1 - \frac{1}{3}) + (1 - 0.5) * \frac{2}{3})] * \mathcal{D}_{SAR} = \frac{1}{3} \mathcal{D}_{SAR}$ . Similarly,  $W_2$  can be calculated as follows. Assume  $E_4^r = 7$ ,  $\bar{\rho}_2 = \frac{8}{3}/2 = \frac{3}{4}$ . On the other hand, because only the sink

is located in  $\mathcal{F}(s_2)$ ,  $\sigma_2 = \frac{4-1}{4} = \frac{3}{4}$ . Therefore,  $W_2 = [1 - (0.5 * (1 - \frac{3}{4}) + (1 - 0.5) * \frac{3}{4})] * \mathcal{D}_{SAR} = \frac{1}{2} \mathcal{D}_{SAR}$ . Clearly,  $s_1$  will make its decision first, because  $W_1 < W_2$ . In Figure 1, both  $s_1$  and  $s_2$  are equipped with two kinds of sensing units that can sense the corresponding sensing attributes at  $t_1$ . Therefore,  $s_1$  and  $s_2$  have approximately the same coverage contribution. However, because the number of  $s_2$ 's neighbors is more than that of  $s_1$ ,  $s_2$  should make its decision later to ensure the network connectivity.

The setting of  $\alpha_n$

The tendency of a sensor toward coverage or connectivity is determined according to the locations of the targets, which are known by all sensors in advance. If the amount of sensed data to be relayed by a sensor is large, the sensor will take a higher priority in relaying the sensed data, instead of sensing the targets. Therefore, a *relaying zone* of a sensor is defined to estimate the amount of sensed data to be relayed by the sensor. Let  $\mathcal{R}(s_n)$  denote the *relaying zone* of  $s_n$ . Definition 4 and Figure 3(b) give the formal definition and an illustration of  $\mathcal{R}(s_n)$ , respectively.

**Definition 4 (Relaying Zone)** *Let  $\mathcal{R}(s_n)$  denote the relaying zone of  $s_n$ .*

$$\mathcal{R}(s_n) = (\mathcal{A} - C(s_0, d(s_0, s_n))) \cup C(s_n, R_c).$$

It is possible that the sensed data from the targets located in  $\mathcal{R}(s_n)$  may need  $s_n$  to relay to the sink. Therefore, the number of targets located in  $\mathcal{R}(s_n)$  can be regarded as the amount of sensed data which needs  $s_n$  to relay to the sink. As a result, a sensor with a larger number of targets in  $\mathcal{R}(s_n)$  shall have a higher connectivity contribution tendency. Formally,  $\alpha_n$  of  $s_n$  is set as follows.

$$\alpha_n = \epsilon * \frac{|\{t_m | t_m \in \mathcal{R}(s_n)\}|}{M},$$

where  $\epsilon$  is a system adjustable parameter to reflect the hardware difference. If the energy cost of the communication unit is much higher than that of the sensing unit, even if the sensor has a higher priority in coverage contribution, the sensor should still pay more attention on the connectivity contribution. In contrast, if the energy cost of the sensing unit is higher than that of the communication unit, the sensor should put a higher weight on coverage contribution, regardless of the distance from the sensor to the sink. Therefore,  $\epsilon$  is set high when the energy cost of the sensing unit is lower than that of the communication unit. Otherwise,  $\epsilon$  is set low accordingly. Basically,  $\epsilon$  is set between 0 and 1.

Notice that  $\alpha_n$  is based on the hardware characteristic, e.g., the energy consumption models of sensing and communication units, and the environment characteristic, e.g., the locations of sensors. These characteristics have an influence on whether a sensor can spend its energy efficiently on sensing or communication. However, these characteristics of a sensor are unchanged after the sensor is deployed. Therefore, it is not sufficient for the sensor to efficiently use its energy. When determining  $W_n$ , a sensor also has to locally take the coverage and connectivity contributions among its neighbors and itself, as well as the remaining energy into consideration to meet the sensing requirements, to ensure the network connectivity, and to prolong the network lifetime.

Countdown  $W_n$

The processes to be performed here are the same as those in Step 2 of REFS.

### Remove incapable or redundant sensing responsibilities

Upon the expiration of  $W_n$ , the remaining  $\Gamma_n$  is the sensing responsibility of  $s_n$  in this round. However, it is still possible for  $s_n$  to alleviate its burden via pruning out redundant sensing responsibilities. For example, for some neighbor of  $s_n$ , say  $s_{n'}$ , if  $\Theta_{n'} \neq \emptyset$ ,  $s_{n'}$  has the responsibility to cover the sensing responsibilities indicated in  $\Theta_{n'}$ . Suppose  $s_{n'}$  turns on  $u^l$ , for some  $l$ , to cover  $t_m^l$  ( $\in \Theta_{n'}$ ), for some  $m$ . If turning on the sensing unit  $u^l$  also covers the other targets, say  $t_{m'}^l$ , for some  $m'$  (that is,  $t_{m'}^l \in \delta_{n'}^l$ ) and  $t_{m'}^l \in \Gamma_n$ , therefore,  $t_{m'}^l$  can be pruned away from  $\Gamma_n$ . As a result,  $\Gamma_n$  can be further improved.

In addition, it is possible to improve by pruning the inefficient sensing responsibilities of  $s_n$  if it is better to leave these responsibilities to the neighbors with higher sensing efficiency. As defined above,  $\delta_n^l$  is the sensing capability of  $u_n^l$ . The more  $|\delta_n^l|$  is, the more targets  $u_n^l$  can cover at a time. Therefore,  $|\delta_n^l|$  can be regarded as the *benefit* of  $u_n^l$  if  $u_n^l$  is turned on. On the contrary,  $\frac{e^l}{E_n^r}$  can be regarded as the *cost* of  $u_n^l$ , where  $e^l$  and  $E_n^r$  are the energy consumption of  $u^l$  in sensing for a time unit and the remaining energy of  $s_n$ , respectively. The cost considers not only the energy consumption of  $u^l$ , but also takes the remaining energy of  $s_n$  into account in order to reflect the effect of the energy consumption of  $u^l$  on the remaining energy of  $s_n$ . Consequently,  $\frac{|\delta_n^l|}{e^l/E_n^r}$  can be regarded as the *benefit-cost ratio* (BCR) of  $u_n^l$ . In addition to the BCR of  $u_n^l$ , the sensing efficiency of  $u_n^l$  on  $s_n$  should take the ratio of the remaining energy of  $s_n$  to the initial energy into consideration as well. Therefore, let  $BCR(u_n^l)$  and  $Eff(s_n, u^l)$  denote the BCR of  $u_n^l$  and the sensing efficiency of  $u^l$  on  $s_n$ , respectively. Accordingly, the sensing efficiency of  $u^l$  on  $s_n$ ,  $Eff(s_n, u^l)$ , is defined as below.

$$Eff(s_n, u^l) = (1 - \alpha_n) * BCR(u_n^l) * \frac{E_n^r}{E}, \text{ where } BCR(u_n^l) = \frac{|\delta_n^l|}{\frac{e^l}{E_n^r}}. \quad (3)$$

In designing  $Eff(s_n, u^l)$ , the contribution tendency is also considered to differentiate the sensors nearby or far away from the sink with the same sensing unit to increase the sensing efficiency. To do so can further alleviate the sensing responsibility of  $s_n$ . As a result, if there exists a sensor, say  $s_{n'}$ , who has not sent out the **DecAnn** packet and whose sensing efficiency of  $u^l$  is better than that of  $u^l$  on  $s_n$ , then  $s_n$  will leave the sensing responsibilities covered by  $u^l$  to  $s_{n'}$ .

### Select the relay

In EEFS, in addition to the neighbor's to-sink-distance, the remaining energy of the neighbor is also taken into account for  $s_n$  to select its relay. Basically, the neighbor with more remaining energy and shorter to-sink-distance will be selected as a relay. However, it is possible that the neighbor with a longer to-sink-distance may be selected. This will result in a longer path to the sink. Therefore, in EEFS, the neighbors located in  $\mathcal{F}(s_n)$  are considered as relay candidates, instead of all the neighbors. Formally,  $nextRelay(n)$  is set as below.

$$nextRelay(n) = \arg \min_{n' \in \mathcal{F}(s_n)} \frac{d(s_{n'}, s_0)}{\max_{j \in \mathcal{F}(s_n)} d(s_j, s_0)} * (1 - \frac{E_{n'}^r}{E}),$$

where  $\frac{d(s_{n'}, s_0)}{\max_{j \in \mathcal{F}(s_n)} d(s_j, s_0)}$  and  $\frac{E_{n'}^r}{E}$  are to normalize the to-sink-distance and the remaining energy of  $s_{n'}$ , respectively. Notice that the case of  $|\mathcal{F}(s_n)| = 0$  is regarded as the local minimum problem and is not addressed in the paper.

Broadcast my decision

$s_n$  announces its decision by sending out a **DecAnn** packet.

Summary

As mentioned above, REFS has a significant drawback that a sensor may turn on too many redundant sensing units. It is because a sensor in REFS only considers its remaining energy and its neighbors' decisions. However, EEFS can alleviate such a situation and make the best use of the sensing and communication units equipped on a sensor. In addition, the coverage and connectivity contributions are introduced in designing  $W_n$  of  $s_n$ . To do so can let the sensor make its decision sooner if it has a higher sensing capability and a lower probability to relay for others. The order of making a decision has a significant impact on the performance of both REFS and EEFS. Moreover, in EEFS, both the coverage and the connectivity contributions are taken into account. As a result, a sensor with a better coverage or connectivity contribution can make its best decision either in covering the target or in relaying the sensed data. Consequently, the network lifetime can be prolonged efficiently.

The complete REFS and EEFS algorithms are omitted due to the space limitation.

## 5. Performance Evaluations

In this section, scheduling multiple sensing units on each sensor to sense a given number of targets and relay the sensed data to the sink in a WHSN are simulated extensively. The simulation setting for the MU-CTC problem is summarized in Table 2. The numbers of sensors and targets are specified in each simulation. The number and types of sensing units on sensors are also specified in each simulation, whereas the number and types of sensing units on each sensor are randomly selected. Similarly, the number and types of attributes required to be sensed at each target are all randomly selected as well. Moreover, targets and sensors are randomly deployed in the sensing field. The locations of sensors and targets are fixed during the whole simulation. The sensing range of each sensing unit is the same and is set as 50 m. The communication range of each sensor is twice of the sensing range. A reliable communication channel is assumed in these simulations. All measurements are averaged over 10 runs, if not otherwise specified.

**Table 2.** Simulation setting for MU-CTC problem.

Sensing field	300 m × 300 m
Sensing range	50 m
Communication range	100 m
Number of sensors	100 .. 600
Number of targets	5 .. 40
Number of attributes	2 .. 7

Since the ILP solution is a centralized scheme, no control and computation overheads are counted. Therefore, two different scenarios are considered. In the first scenario, REFS and EEFS are compared with the ILP solution to show the efficiency of the proposed schemes without the control and computation overheads, where the ILP solution is implemented by ILOG CPLEX [24] optimization library. In the second scenario, REFS and EEFS are evaluated when the control and computation overheads are considered. In order to show the effectiveness of the proposed algorithms, a straightforward scheme, named *m*-SU-CTC, is compared in this scenario. As mentioned above, there existed several schemes in the literature considering the CTC problem on a wireless homogeneous sensor network, where each sensor is equipped with only one sensing unit and only one attribute is required to be sensed at each target. Therefore, the *m*-SU-CTC will apply this kind of scheme multiple times, each for the required attribute to be sensed at the targets, so that all the attributes required to be sensed by every target are sensed by the sensors with those specific sensing units on sensors.

Energy consumption and network lifetime are mainly evaluated to verify the performance of the proposed schemes. As mentioned above, the network lifetime is defined as the time interval from the beginning to the time that either the attributes required to be sensed at any target can not be sensed anymore or the sensed data can not be delivered to the sink. On the other hand, the energy consumption models for sensing, communication, and computation are different in scenarios 1 and 2. In scenario 1, the initial energy of each sensor is 50 units. The energy consumption of each type of sensing unit is assumed linearly proportional to the type of the sensing unit. That is, the first type of the sensing unit is assumed to consume one unit of energy in a round, the second type of the sensing unit consumes two units of energy, and likewise. Additionally, the communication module of a sensor consumes one unit of energy to send or receive a unit of the sensed data.

In scenario 2, the energy consumption caused by control and computation overheads are considered. The energy consumption model of MICA2 [4] is adopted in the simulation. In addition, the initial energy of each sensor is assumed 2000 J. A sensor in active mode consumes 10.9 mA. Similar to scenario 1, the energy consumption of each type of the sensing unit is also assumed linearly proportional to the type of the sensing unit. However, the unit of the energy consumption of the sensing units is J/min. If a sensor decides to turn on any sensing unit, the sensor is assumed to reply the sensed data to the sink every 10 minutes. Each round is 100 minutes. Table 3 summaries the energy consumption model used in scenario 2.

Since there is a system parameter  $\epsilon$  involved in the design of the EEFS in order to obtain an appropriate value of  $\epsilon$  for scenarios 1 and 2, the experiment contains two parts. The first part is to observe the impact of  $\epsilon$  on the performance of EEFS for scenarios 1 and 2. According to the results obtained from the first part, the second part of the experiment evaluates the performance of the proposed schemes. As mentioned above, scenario 1 compares the performance of the proposed schemes, REFS and EEFS, against the ILP solution, where the control and computation overheads are not taken into account. In scenario 2, the proposed schemes are compared against a heuristic scheme, where the control and computation overheads are considered.

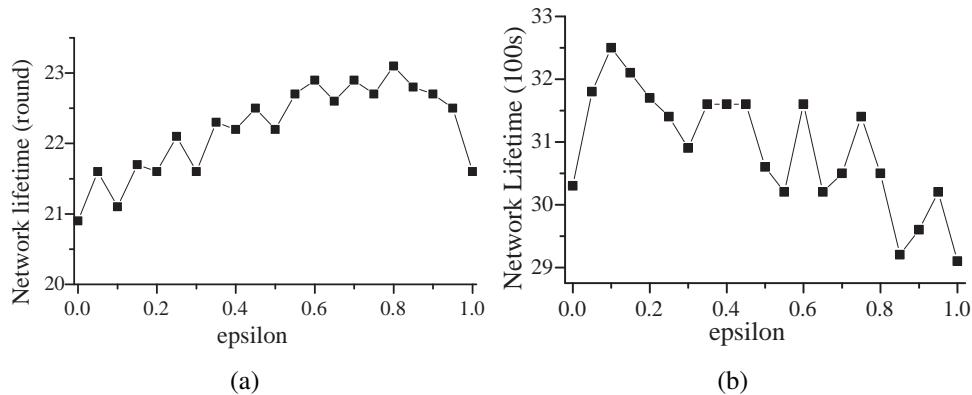
**Table 3.** Energy consumption model for MU-CTC problem when control and computation overheads are considered.

Active	10.9 mA
Transmit (+4 dBm)	11.6 mA
Receive	7.0 mA
Sensing unit	$u^1 \quad u^2 \quad u^3 \quad \dots \quad u^l$
Energy consumption ( $J/min$ )	1    2    3 $\dots$ $l$

### 5.1. The Impact of $\epsilon$

In EEFS,  $\epsilon$  is a system parameter considering the difference of the energy consumption between the communication unit and the sensing units. The use of the  $\epsilon$  is to make the best use of the sensing and communication units and increase the network lifetime. Therefore, this part of experiment focuses on the impacts of  $\epsilon$  on the network lifetime for scenarios 1 and 2. In this experiment, three types of attributes are required to be sensed and there are 25 targets and 300 sensors randomly deployed in the 300 m \* 300 m sensing field.

**Figure 4.** The impact of  $\epsilon$  on the network lifetime for (a) scenario 1 and (b) scenario 2.



Figures 4(a) and 4(b) show the results for scenarios 1 and 2, respectively. According to Figure 4(a), basically, the network lifetime increases with the increase of  $\epsilon$  until  $\epsilon = 0.8$ . The best performance is observed when  $\epsilon = 0.8$ . In scenario 1, the energy spent in sensing is close to that spent in communication, which implies that the cost of the communication unit is higher than that of the sensing unit. As mentioned above, the sensor should pay more attention on the connectivity contribution. Consequently,  $\epsilon$  should be set high, which is also coincident with the simulation results shown in Figure 4(a). Note that, in scenario 1, the control overhead is not taken into account. However, if the control overhead is taken into consideration,  $\epsilon$  should be larger than 0.8.

On the other hand, in scenario 2, the energy cost of the sensing unit is much higher than that of the communication unit. Therefore, the sensor should have a higher tendency toward the coverage contribution. As a result,  $\epsilon$  should be set low to have the sensor toward the coverage tendency. According to Figure 4(b), the network lifetime decreases with the increase of  $\epsilon$ . The network obtains the

longest lifetime when  $\epsilon = 0.1$ . Consequently, for the following simulations,  $\epsilon$  are set to 0.8 and 0.1 for scenarios 1 and 2, respectively.

## 5.2. Performance Evaluations of the Proposed Schemes

### Energy consumption taken from a snapshot

Figure 5 shows the snapshots of the remaining energy of each sensor in the last round of REFS and EEFS, respectively. The circles are the targets to be sensed and the gray squares represent the remaining energy of the corresponding sensor. The darker the gray color of the sensor is, the more remaining energy the sensor has. Figure 5 shows the snapshots of the remaining energy of the sensors in the last round of REFS and the corresponding round of EEFS both in scenarios 1 and 2. On average, the gray squares in Figures 5(b) and 5(d) are darker than those in Figures 5(a) and 5(c), which means that the remaining energy of the sensors in EEFS is higher than that in REFS. Moreover, in addition to the remaining energy, the number of remaining sensors (gray squares) in EEFS is also more than that in REFS, especially in scenario 2. These all imply that EEFS performs more efficiently than REFS.

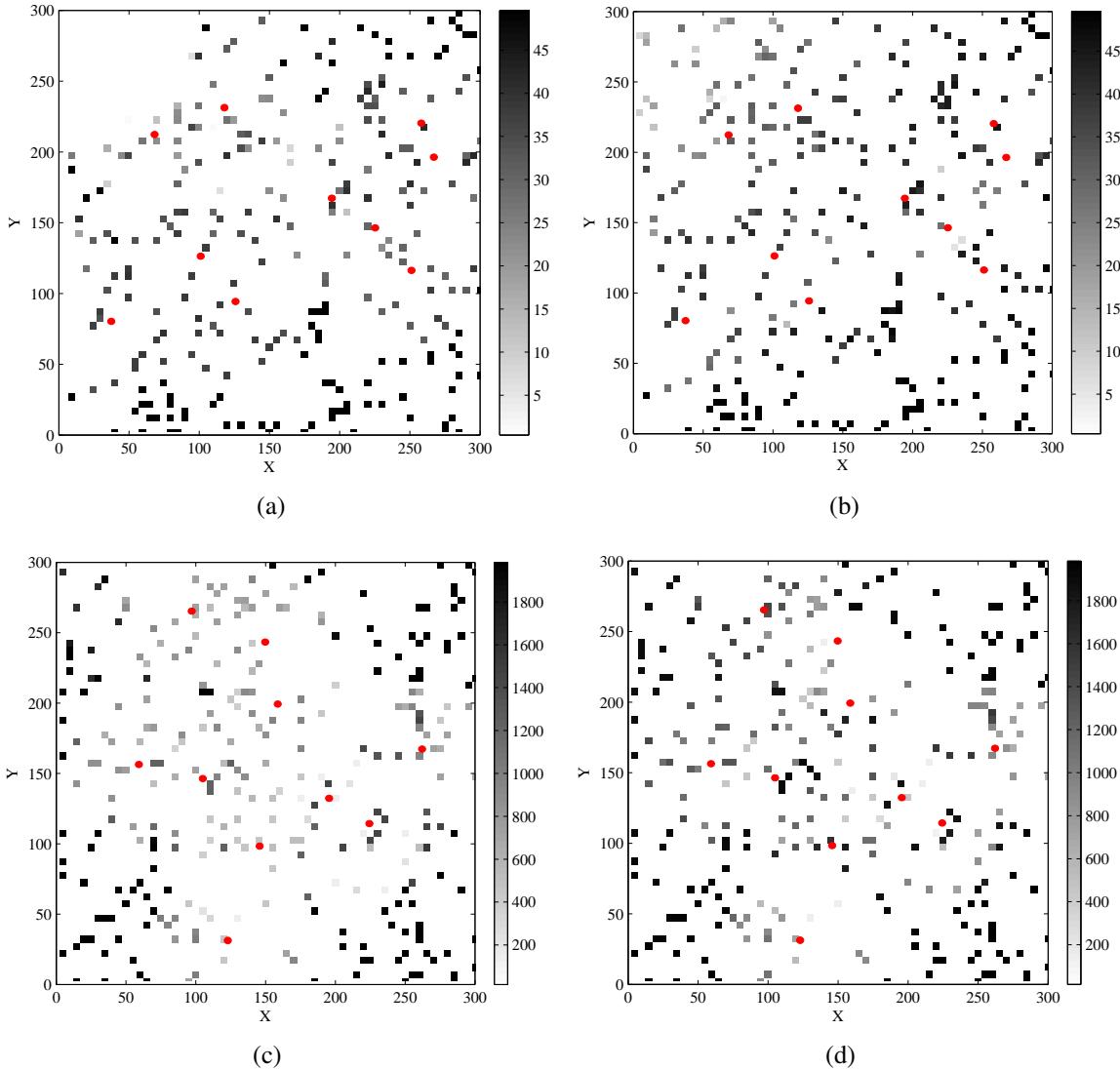
### Energy consumption and remaining energy in a run

To look into the detailed performance of REFS, EEFS, as well as the ILP solution, Figure 6 illustrates the energy consumption and the remaining energy in each round, which is randomly taken from some run, rather than the average of 10 runs. In addition to the settings of the scenarios 1 and 2 described above, the other settings are as follows. The numbers of sensors, targets, and attributes are 300, 10, and 3, respectively.

Figure 6(a) shows the energy cost of REFS, EEFS, and the ILP solution consumed in each round for some run in scenario 1. Obviously, the energy consumptions of REFS and EEFS fluctuate with time, while that of the ILP solution stays stable. This is because the sensing redundancy problem in the ILP solution is slight. Therefore, the energy consumption of the ILP solution in each round is very similar. On the contrary, REFS and EEFS are heuristic. The energy consumptions of REFS and EEFS in each round depend heavily on the number of sensing and communication units selected. Therefore, the energy consumptions of REFS and EEFS have a large variation among rounds. Moreover, since the sensing redundancy problem is much more serious in REFS, the energy consumption of REFS is always the worst and the network lifetime of REFS is the shortest. In addition, in EEFS, sensors consume less energy to perform the sensing tasks at the beginning because many redundant sensing units are removed. However, when the number of failed sensors increases, some sensing responsibilities of sensors may become the critical sensing responsibilities. Therefore, the sensing redundancy problem will be getting worse, which will result in a higher energy consumption of the sensors in a round.

On the other hand, although the energy consumption of EEFS is higher than that of the ILP solution in each round, the network lifetime of EEFS is still very close to that of the ILP solution, whose difference is smaller than eight rounds, as shown in Figure 6(b). Figures 6(c) and 6(d) illustrate the energy consumption and the remaining energy of REFS and EEFS in each round when the control and computation overheads are considered. The results are similar to those shown in Figures 6(a) and 6(b).

**Figure 5.** The snapshots of the remaining energy of sensors by (a) REFS and (b) EEFS in scenario 1 and (c) REFS and (d) EEFS in scenario 2, where the snapshots are taken from the end of the 27th round in scenario 1 and are taken from the 3800th minute in scenario 2.

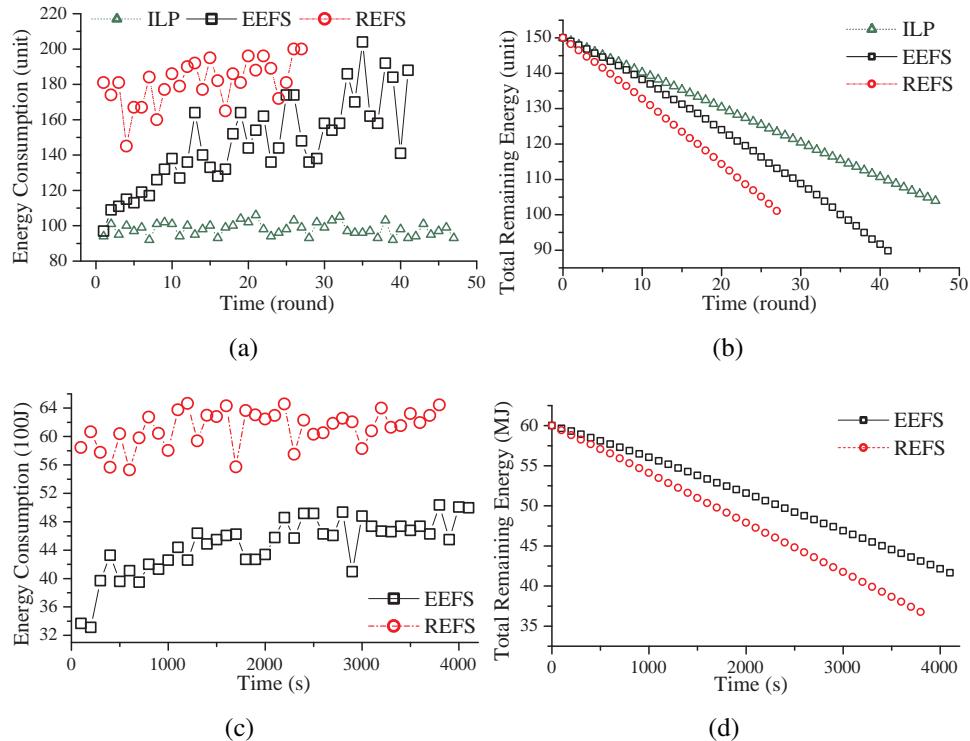


#### The impacts of the numbers of sensors, targets, and attributes on network lifetime

The following experiments are to observe the impacts of *the number of sensors*, in which the number of targets and the number of attributes are fixed at 10 and 3, *the number of targets*, in which the number of sensors and the number of attributes are fixed at 300 and 3, and *the number of attributes*, in which the number of sensors and the number of targets are fixed at 300 and 10, on the network lifetime with regard to REFS, EEFS, and the ILP solution. The results for scenarios 1 and 2 are shown in Figure 7.

Figures 7(a), 7(b) and 7(c) illustrate the simulation results of scenario 1. Since the ILP solution becomes extraordinarily complicated when the numbers of attributes, targets or sensors increase, some results of the ILP solution are unavailable. Obviously, the network lifetime increases with the increase of the number of sensors. On the contrary, when the number of targets or attributes increases, the network lifetime decreases accordingly. However, the ILP solution always has the longest network lifetime, if

**Figure 6.** The energy consumption and the remaining energy in each round for the some run. (a) The energy consumption and (b) the remaining energy in each round for some run in scenario 1. (c) The energy consumption and (d) the remaining energy in each round for some run in scenario 2.



available, whereas REFS is the worst in all cases. Nevertheless, the performance of EEFS is close to that of the ILP solution. From the practical viewpoint, EEFS is a promising alternative solution.

Figures 7(d), 7(e) and 7(f) respectively show the impacts of the number of sensors, targets, and attributes on network lifetime when the control and computation overheads are taken into consideration in scenario 2. Here, the communication overhead takes the messages exchanged among sensors into account, such as the sensor's capabilities, decision announcement, etc.

Basically, Figures 7(d), 7(e) and 7(f) are similar to Figures 7(a), 7(b) and 7(c), respectively. As indicated in Figure 7(d), EEFS, REFS, and  $m$ -SU-CTC extend the network lifetime when the number of sensors increases. Among them, EEFS performs the best when a large number of sensors are deployed in the area of interest, even though EEFS needs extra control and computation overheads.

**Figure 7.** The impacts of the numbers of (a) sensors, (b) targets, and (c) attributes in scenario1 as well as (d) sensors, (e) targets, and (f) attributes in scenario 2 on the network lifetime.

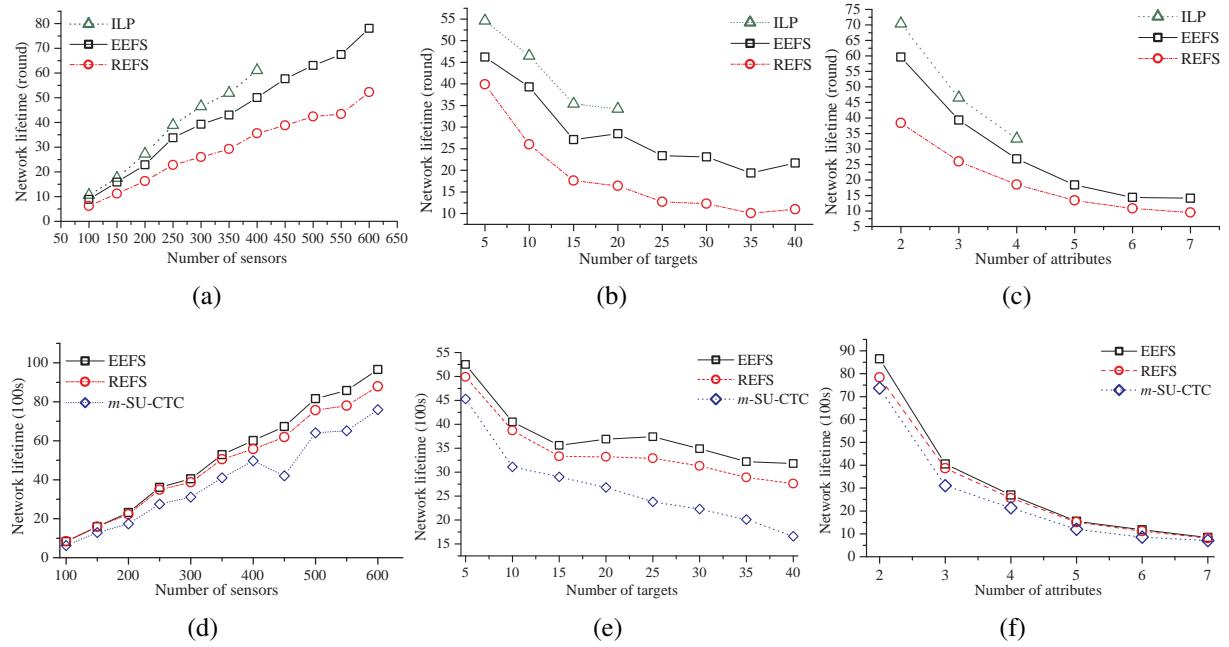


Figure 7(e) illustrates the impact of the number of targets on the network lifetime. It can be found that the network lifetime of *m*-SU-CTC rapidly decreases when the number of targets increases. However, the network lifetimes of EEFS and REFS decrease smoothly. That is because, with the aid of the proposed algorithms, the increase of the number of targets does not activate too many sensing units on sensors. The simulation result also indicates that *m*-SU-CTC is unsuitable for WHSNs when a large number of targets needs to be sensed.

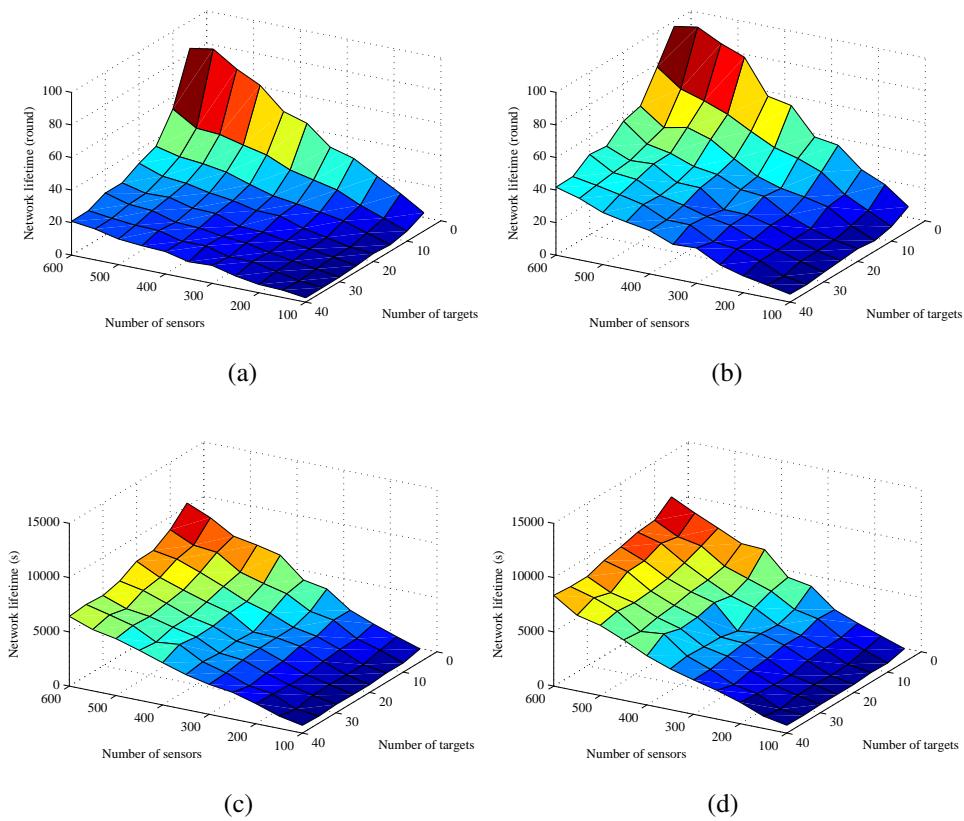
According to Figure 7(f), the impact of the number of attributes on the network lifetime is not significant when the number of attributes is larger than 5. Basically, EEFS performs better than REFS and *m*-SU-CTC. However, the differences of the network lifetime among EEFS, REFS and *m*-SU-CTC are not significant, especially when the number of attributes is larger than 5. Since only 300 sensors are deployed in the area of interest and many kinds of attributes of the targets are needed to be sensed, a lot of sensors have to be activated to sense the required attributes. As a result, the effect of the policies to remove redundant or inefficient sensing responsibilities in the EEFS scheme is not obvious. Thus, the differences of the network lifetime between EEFS and REFS are not significant. The result is consistent with that in Figure 7(d), which indicates that the number of sensors to be deployed should be increased when the number of attributes increases.

It is worth noticing that *m*-SU-CTC performs the worst in all cases. It is because that *m*-SU-CTC is originally designed for the sensor equipped with only one sensing unit, which is unsuitable to be applied for WHSNs. Without good coordination among sensing units or sensors, *m*-SU-CTC will turn on many redundant sensing units or sensors, and thus, result in higher energy consumption.

## Comprehensive comparisons of REFS and EEFS

Figure 8 shows the performance of REFS and EEFS in a comprehensive view combining the numbers of sensors and targets to observe their impact on the network lifetime. In the simulation, the number of sensors is varied from 100 to 600 with a step of 100, the number of targets is varied from 5 to 40 with a step of 5, and the number of attributes is fixed at 3.

**Figure 8.** Comprehensive comparisons of REFS and EEFS. The impact of the numbers of sensors and targets on the network lifetime in terms of (a) REFS and (b) EEFS in scenario 1 as well as (c) REFS and (d) EEFS in scenario 2.



In scenario 1, both REFS and EEFS have similar inclinations that the network lifetime increases with the increase of the number of sensors, but decreases with the increase of the number of targets, as shown in Figures 8(a) and 8(b). However, EEFS still shows a better performance than REFS. When the control and computation overheads are taken into consideration, both REFS and EEFS also have similar inclination that the network lifetime increases with the increase of the number of sensors, but decreases with the increase of the number of targets, as shown in Figures 8(c) and 8(d). Also, EEFS has better performance than REFS. In addition, when control and computation overheads are considered, the impact of the number of targets on the network lifetime is not significant. That is because when the number of targets increases, the energy consumption in sensing the required attributes does not increase substantially. It is shown again that, in practical view, when multiple attributes are required to be sensed, deploying a WHSN with multiple sensing units is a promising solution.

### 5.3. Summary

The simulation results can be summarized as follows.

- (1) The network lifetime increases with the increase of the number of sensors.
- (2) The network lifetime decreases with the increase of the number of targets.
- (3) The network lifetime decreases with the increase of the number of attributes.
- (4) The performance of EEFS is very close to that of the ILP solution, which is an optimal solution.
- (5) According to the simulation results, even though the control overhead of EEFS is higher than that of REFS, the performance of EEFS is still better than that of REFS. However, since the computation cost of EEFS is higher than that of REFS, if the computation capability of a sensor is not good enough, REFS may be a better choice.
- (6) The modified algorithm *m*-SU-CTC originally designed for the sensor with one sensing unit may not work well in WHSNs because *m*-SU-CTC does not jointly take the sensing abilities of the equipped sensing units into consideration.
- (7) Even though the ILP solution performs the best, REFS and EEFS are still the practical solutions to WHSNs.

## 6. Conclusions

Coverage and connectivity are important measurements for the quality of surveillance that a sensor network can provide. Therefore, the paper emphasizes on the connected target coverage problem in wireless heterogeneous sensor networks with multiple sensing units, termed the MU-CTC (**M**ultiple **s**ensing **U**nits for **C**onnected **T**arget **C**overage) problem. The problem is to schedule the activity of each sensing unit on each sensor to completely cover the targets of interest, to make sensors relay the sensed data to the sink, and, subject to the energy constraint of each sensor, to maximize the network lifetime. The problem is further reduced to a connected set cover problem, called the MU-CSC (**M**ultiple **s**ensing **U**nits for **C**onnected **S**et **C**over) problem. According to the MU-CSC problem, several ILP constraints are proposed. In addition, two distributed schemes, REFS and EEFS, are proposed to solve the MU-CTC problem. These two schemes are executed by each sensor in the initial phase of each round. In REFS, each sensor enables its sensing units by its remaining energy and neighbors' decisions. However, in EEFS, the coverage and connectivity contributions are also taken into account. Simulation results show that REFS and EEFS can prolong the network lifetime effectively. The performances of the two schemes are close to that of the ILP solution. However, the ILP solution is a centralized and computationally intensive scheme, so the proposed distributed schemes are much more practical. In addition, the schemes can be easily implemented in a real network.

In the future, different levels of target coverage requirements (e.g., *k*-coverage) will also be considered to meet the sensing requirements of different applications. From the practical viewpoint, the sensing units with different sensing ranges will also be taken into consideration in the future.

## Acknowledgements

The work was partially supported by the National Science Council of the Republic of China under Grants NSC 97-3114-E-119-001 and NSC 98-2218-E-254-001.

## References and Notes

1. Akyildiz, I.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. Wireless Sensor Networks: A Survey. *Comput. Netw.* **2002**, *38*, 393–422.
2. Sanli, H.O.; Poornachandran, R.; Çam, H. Collaborative Two - Level Task Scheduling for Wireless Sensor Nodes with Multiple Sensing Units. In *Proceedings of the IEEE International Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, Santa Clara, CA, USA, 2005; pp. 350–361.
3. Shih, K.-P.; Wang, S.-S.; Yang, P.-H.; Chang, C.-C. Collect: Collaborative Event Detection and Tracking in Wireless Heterogeneous Sensor Networks. In *Proceedings of the IEEE International Symposium on Computers and Communications (ISCC)*, Pula-Cagliari, Sardinia, Italy, 2006; pp. 935–940.
4. Crossbow Technology, Inc. Available Online: <http://www.xbow.com> (accessed on 8 June 2009).
5. Kar, K.; Banerjee, S. Node Placement for Connected Coverage in Sensor Networks. In *Proceedings of the First Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, Sophia Antipolis, France, 2003.
6. Wang, Y.-C.; Hu, C.-C.; Tseng, Y.-C. Efficient Placement and Dispatch of Sensors in a Wireless Sensor Network. *IEEE Trans. Mob. Comput.* **2008**, *7*, 262–274.
7. Kumar, S.; Lai, T. H.; Balogh, J. On k-Coverage in a Mostly Sleeping Sensor Network. In *Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM)*, San Francisco, CA, USA, 2004; pp. 144–158.
8. Huang, C.-F.; Tseng, Y.-C. The Coverage Problem in a Wireless Sensor Network. *Mobile Netw. Appl.* **2005**, *10*, 519–528.
9. Tian, D.; Georganas, N.D. A Coverage-Preserving Node Scheduling Scheme for Large Wireless Sensor Networks. In *Proceedings of the ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*, Atlanta, GA, USA, 2002; pp. 32–41.
10. Yan, T.; He, T.; Stankovic, J.A. Differentiated Surveillance for Sensor Networks. In *Proceedings of the ACM International Conference on Embedded Networked Sensor Systems (SENSYS)*, Los Angeles, CA, USA, 2003; pp. 51–62.
11. Zhang, H.; Hou, J. Maintaining Sensing Coverage and Connectivity in Large Sensor Networks. *Ad Hoc Sens. Wirel. Netw.* **2005**, *1*, 89–124.
12. Shih, K.-P.; Chen, Y.-D.; Chiang, C.-W.; Liu, B.-J. A Distributed Active Sensor Selection Scheme for Wireless Sensor Networks. In *Proceedings of the IEEE International Symposium on Computers and Communications (ISCC)*, Pula-Cagliari, Sardinia, Italy, 2006; pp. 923–928.

13. Cardei, M.; Thai, M.T.; Li, Y.; Wu, W. Energy-Efficient Target Coverage in Wireless Sensor Networks. In *Proceedings of the IEEE INFOCOM, the Annual Joint Conference of the IEEE Computer and Communications Societies*, Miami, FL, USA, 2005; Vol. 3, pp. 1976–1984.
14. Cardei, M.; Wu, J.; Lu, M.; Pervaiz, M.O. Maximum Network Lifetime in Wireless Sensor Networks with Adjustable Sensing Ranges. In *Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Montreal, Canada, 2005; Vol. 3, pp. 438–445.
15. Liu, H.; Wan, P.; Yi, C.-W.; Jia, X.; Makki, S.; Pissinou, N. Maximal Lifetime Scheduling in Sensor Surveillance Networks. In *Proceedings of the IEEE INFOCOM, the Annual Joint Conference of the IEEE Computer and Communications Societies*, Miami, FL, USA, 2005; Vol. 4, pp. 2482–2491.
16. Liu, H.; Wan, P.; Jia, X. Maximal Lifetime Scheduling for Sensor Surveillance Systems with K Sensors to One Target. *IEEE Trans. Parallel Distrib. Syst.* **2006**, *17*, 1–11.
17. Liu, H.; Jia, X.; Wan, P.-J.; Yi, C.-W.; Makki, S.K.; Pissinou, N. Maximizing Lifetime of Sensor Surveillance Systems. *IEEE-ACM Trans. Netw.* **2007**, *15*, 334–345.
18. Zhao, Q.; Gurusamy, M. Maximizing Network Lifetime for Connected Target Coverage in Wireless Sensor Networks. In *Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Montreal, Canada, 2006; pp. 94–101.
19. Yang, S.; Dai, F.; Cardei, M.; Wu, J. On Multiple Point Coverage in Wireless Sensor Networks. In *Proceedings of the IEEE International Conference on Mobile Ad - hoc and Sensor Systems (MASS)*, Washington, DC, USA, 2005; pp. 757–764.
20. Cardei, I. Energy-Efficient Target Coverage in Heterogeneous Wireless Sensor Networks. In *Proceedings of the IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS)*, Vancouver, Canada, 2006; pp. 397–406.
21. Shih, K.-P.; Wang, S.-Y.; Chen, H.-C.; Liu, B.-J. On Target Coverage in Wireless Heterogeneous Sensor Networks with Multiple Sensing Units. In *Proceedings of the IEEE International Symposium on Computers and Communications (ISCC)*, Aveiro, Portugal, 2007; pp. 185–190.
22. Karp, B.; Kung, H.T. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In *Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM)*, Boston, MA, USA, 2000; pp. 243–254.
23. Seada, K.; Helmy, A.; Govindan, R. Modeling and Analyzing the Correctness of Geographic Face Routing under Realistic Conditions. *Ad Hoc Netw.* **2007**, *5*, 855–871.
24. ILOG CPLEX. Available Online: <http://www.ilog.com> (accessed on 8 June 2009).