Improving network lifetime using sensors with adjustable sensing ranges

Mihaela Cardei,* Jie Wu and Mingming Lu

Department of Computer Science and Engineering,
Florida Atlantic University,
777 Glades Road,
Boca Raton, FL 33431, USA
Fax: +1-561-297-2800
E-mail: mihaela@cse.fau.edu
E-mail: jie@cse.fau.edu   E-mail: mlu2@fau.edu
*Corresponding author

Abstract: This paper addresses the target coverage problem in wireless sensor networks with adjustable sensing range. Communication and sensing consume energy, therefore efficient power management can extend network lifetime. In this paper, we consider a large number of sensors with adjustable sensing range that are randomly deployed to monitor a number of targets. Since targets are redundantly covered by multiple sensors, to conserve energy resources, sensors can be organised in sets, activated successively. In this paper, we address the Adjustable Range Set Covers (AR-SC) problem that has as its objective finding a maximum number of set covers and the ranges associated with each sensor, such that each sensor set covers all the targets. A sensor can participate in multiple sensor sets, but the sum of the energy spent in each set is constrained by the initial energy resources. In this paper, we mathematically model solutions to this problem and design heuristics that efficiently compute the sets. Simulation results are presented to verify our approaches.

Biographical notes: Mihaela Cardei is an Assistant Professor in the Department of Computer Science and Engineering at Florida Atlantic University and the Director of the NSF-funded Wireless and Sensor Network Laboratory. She received a PhD and an MS in Computer Science from the University of Minnesota, Twin Cities, in 2003 and 1999, respectively. Her research interests include wireless networking, wireless sensor networks, network protocol and algorithm design and resource management in computer networks. She is a Member of IEEE and ACM.

Jie Wu is a Professor in the Department of Computer Science and Engineering, Florida Atlantic University. He has published over 300 papers in various journals and conference proceedings. His research interests are in the areas of wireless networks and mobile computing, routing protocols, fault-tolerant computing and interconnection networks. He served on many conference organisation committees and editorial boards. He was also the recipient of the 1996-1997 and 2001-2002 Researcher of the Year Award at Florida Atlantic University. He has also served as an IEEE Computer Society Distinguished Visitor. He is a Member of ACM and a Senior Member of IEEE.

Mingming Lu is a PhD candidate in the Department of Computer Science and Engineering, Florida Atlantic University. He received a BS and an MS in Computer Science from Tongji University and Fudan University, Shanghai, China in 2000 and 2003, respectively. His current research focuses on the algorithm design in the wireless ad hoc and sensor networks.

1 Introduction

Wireless Sensor Networks (WSNs) constitute the foundation of a broad range of applications related to national security, surveillance, military, healthcare and environmental monitoring. One important class of WSNs is wireless ad hoc sensor networks, characterised by an ad hoc or *random* sensor deployment method (Megerian and Potkonjak, 2002), where the sensor location is not known a priori. This feature is required when individual sensor placement is infeasible, such as battlefield or disaster areas. Generally, more sensors are deployed than required (compared with the optimal

placement) to perform the proposed task; this compensates for the lack of exact positioning and improves fault tolerance. The characteristics of a sensor network (Akyildiz et al., 2002) include limited resources, large and dense networks and a dynamic topology.

An important issue in sensor networks is power scarcity, driven in part by battery size and weight limitations. Mechanisms that optimise sensor energy utilisation have a great influence on prolonging the network lifetime. Power saving techniques can generally be classified into two categories:

1   scheduling the sensor nodes to alternate between active and sleep mode and

2   adjusting the transmission or sensing range of the wireless nodes.

In this paper, we deal with both methods. We design a scheduling mechanism in which only some of the sensors are active, while all other sensors are in the sleep mode. Also, for each sensor in the set, the goal is to have a minimum sensing range while meeting the application requirements.

In this paper, we address the *target coverage* problem. The goal is to maximise the network lifetime of a power-constrained wireless sensor network, deployed for monitoring a set of targets with known locations. We consider a large number of sensors, deployed randomly in close proximity to a set of targets, that send the sensed information to a central node for processing. The method used to extend the network's lifetime is to divide the sensors into a number of sets. Using the property that sensors have adjustable sensing ranges, the goal is to set up minimum sensing ranges for the active sensors, while satisfying the coverage requirements. Besides reducing the energy consumed, this method lowers the density of active nodes, thus reducing interference at the MAC layer.

The contributions of this paper are the following:

1   introduce the Adjustable Range Set Covers (AR-SC) problem and the mathematical model

2   design efficient heuristics (both centralised and distributed) to solve the AR-SC problem, using linear programming and greedy techniques and

3   analyse the performance of our approaches through simulations.

The rest of this paper is organised as follows. In Section 2 we present related works on sensor coverage problems. Section 3 defines AR-SC problem and Section 4 presents our heuristic contributions. In Section 5 we present the simulation results and Section 6 concludes this paper.

2   State-of-the art

In this paper, we address the sensor coverage problem. As pointed out in Meguerdichian et al. (2001), the coverage concept is a measure of the Quality of Service (QoS) of the sensing function and is subject to a wide range of interpretations due to a large variety of sensors and applications. The goal is to have each location in the physical space of interest within the sensing range of at least one sensor.

A survey on coverage problems in wireless sensor networks is presented in Cardei and Wu (2006). The coverage problems can be classified in the following types (Cardei and Wu, 2006):

1   area coverage (Carle and Simplot, 2004; Tian and Georganas, 2002; Wang et al., 2003; Wu and Yang, 2004; Zhang and Hou, 2004), where the objective is to cover an area

2   point coverage (Cardei and Du, 2005; Cardei et al., 2005; Cheng et al., 2005), where the objective is to cover a set of targets and

3   coverage problems that have the objective to determine the maximal support/breach path that traverses a sensor field (Meguerdichian et al., 2001).

An important method for extending the network lifetime for the area coverage problem is to design a distributed and localised protocol that organises the sensor nodes in sets. The network activity is organised in rounds, with sensors in the active set performing the area coverage, while all other sensors are in the sleep mode. Set formation is done based on the problem requirements such as energy-efficiency, area monitoring, connectivity, etc. Different techniques have been proposed in the literature (Carle and Simplot, 2004; Tian and Georganas, 2002; Wang et al., 2003; Wu and Yang, 2004; Zhang and Hou, 2004) for determining the eligibility rule, that is, to select which sensors will be active in the next round. Wu and Yang (2004) addressed area coverage when sensors can adjust their sensing ranges.

For applications that require more stringent fault-tolerance or for positioning applications, k-coverage might be a requirement. In Huang and Tseng (2003), the goal is to determine whether a given area satisfies the k-coverage requirement, when each point in the area of interest is covered by at least k sensors. Both uniform and non-uniform sensing ranges are considered, and the k-coverage property is reduced to the k perimeter coverage of each sensor in the network.

A different coverage formulation is given in Meguerdichian et al. (2001). A path has the worst (best) coverage if it has the property that for any point on the path, the distance to the closest sensor is maximised (minimised). Given the initial and final locations of an agent and a field instrumented with sensors, Meguerdichian et al. (2001) proposed centralised solutions to the worst (best) coverage based on the observation that worst coverage path lies on the Voronoi diagram lines and best coverage path lies on Delaunay triangulation lines.

The works most relevant to our approaches are (Cardei and Du, 2005) and (Cardei et al., 2005). Cardei and Du (2005) introduces the target coverage problem, where disjoint sensor sets are modelled as disjoint set covers, such that every cover completely monitors all the target points. The disjoint set coverage problem is proved to be NP-complete and a lower bound of two for any polynomial-time approximation algorithm is indicated. The disjoint set cover problem (Cardei and Du, 2005) is reduced to a maximum flow problem, which is then modelled as mixed integer programming. This problem is further extended in Cardei et al. (2005), where sensors are not restricted to participation in only disjoint sets, that is, a sensor can be active in more than one set.

The coverage breach problem is introduced in Cheng et al. (2005), addressing the case when sensor networks have limited bandwidth. The objective of the problem is to organise the sensors in disjoint sets, such that each set has a given bounded number of sensors and the overall breach is minimised. The overall breach is measured as the number of targets uncovered by the sensor sets.

Our paper is an extension of the maximum set covers problem addressed in Cardei et al. (2005), for the case when sensor nodes can adjust their sensing range. Our goal is to reduce the sensing range of the active sensors, while maintaining the coverage requirements. This method has a double impact: first it reduces energy consumption and second it reduces interference at the MAC layer. Sensors with adjustable sensing ranges are available commercially (Photoelectric Sensors, 2000; Wu and Yang, 2004).

Compared with Cardei et al. (2005), in this paper we are also concerned with designing a distributed and localised algorithm (see Section 4.2.2) for the AR-SC problem. Distribution and localisation are important properties of a node scheduling mechanism, as it adapts better to a scalable and dynamic topology.

3 Problem definition

Let us assume that N sensors $s_1, s_2,..., s_N$ are randomly deployed to cover M targets $t_1, t_2,..., t_M$. Each sensor has an initial energy E and has the capability to adjust its sensing range. Sensing range options are $r_1, r_2,..., r_P$, corresponding to energy consumptions of $e_1, e_2,..., e_P$.

We assume a Base Station (BS) located within the communication range of each sensor. One method to compute the sensor - target coverage relationship is to consider that a sensor covers a target if the Euclidean distance between the sensor and target is no greater than a predefined sensing range. The formal problem definition is given below:

Definition 1: *Target Coverage Problem (Cardei et al., 2005) Given M targets with known location and an energy-constrained WSN with N sensors randomly deployed in the targets' vicinity, schedule the sensor nodes' activity such that all targets are continuously observed and the network lifetime is maximised.*

The approach we used in this paper is to organise the sensors in sets, such that only one set is responsible for monitoring the targets and all other sensors are in the sleep mode. Besides determining the set covers, we are also concerned with setting the sensing range of each active sensor. The goal is to use a minimum sensing range to minimise the energy consumption, while meeting the target coverage requirement.

Next we formally define the Adjustable Range Set Covers (AR-SC) problem, used to solve the target coverage problem.

Definition 2: *AR-SC Problem Given a set of targets and a set of sensors with adjustable sensing ranges, find a family of set covers $c_1, c_2, \ldots, c_K$ and determine the sensing range of each sensor in each set, such that:*

1 *K is maximised*

2 *each sensor set monitors all targets and*

3 *each sensor appearing in the sets $c_1, c_2, \ldots, c_K$ consumes at most E energy.*

In the AR-SC definition, the requirement to maximise K is equivalent with maximising the network lifetime. The sensing range of a sensor determines the energy consumed by the sensor when that set is activated. If a sensor participates in more than one set, then the sum of energy spent has to be at most E.

The AR-SC problem is NP-complete, as proved by the restriction method (Garey and Johnson, 1979). Maximum Set Covers (Cardei et al., 2005) is a special case of AR-SC problem when the number of sensing ranges $P = 1$ and when the time a sensor is active is considered to be the energy consumed.

Figure 1(a) shows an example with four sensors s_1, s_2, s_3, s_4 and three targets t_1, t_2, t_3. Each sensor has two sensing ranges r_1, r_2 with $r_1 < r_2$. In this example, we assume a node's sensing area is the disk centred at the sensor, with a radius equal to the sensing range. We use a solid line to denote range r_1 and a dotted line for range r_2. The coverage relationships between sensors and targets are also illustrated in Figure 1(b): $(s_1, r_1) = \{t_3\}$, $(s_1, r_2) = \{t_1, t_3\}$, $(s_2, r_1) = \{t_1\}$, $(s_2, r_2) = \{t_1, t_2\}$, $(s_3, r_1) = \{t_2\}$, $(s_3, r_2) = \{t_2, t_3\}$, $(s_4, r_1) = \{t_1, t_3\}$ and $(s_4, r_2) = \{t_1, t_2, t_3\}$. The dotted lines in Figure 1(b) show the additional targets covered by increasing the sensing range from r_1 to r_2. Note that a circular sensing area is not a requirement for our solution; we are just concerned with identifying which sensors cover each target.

Figure 1 Example with three targets $T = \{t_1, t_2, t_3\}$ and four sensors $S = \{s_1, s_2, s_3, s_4\}$
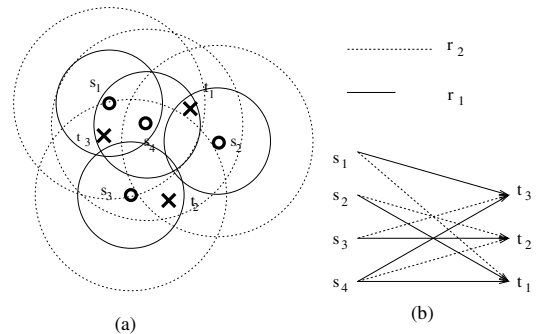
In this paper, a sensor can be part of more than one cover set. Let us consider for this example $E = 2$, $e_1 = 0.5$ and $e_2 = 1$. Each set cover is active for a unit time of 1. One solution for the AR-SC problem uses the set covers illustrated in Figure 2. This solution has five different set covers and maximum lifetime 6, obtained for example with the following sequence of set covers: C_1, C_2, C_3, C_4, C_5 and C_4. After this sequence, the residual energy of each sensor becomes zero.

If sensor nodes do not have adjustable sensing ranges, then we obtain a lifetime five for a sensing range equal to r_2. When sensing range is r_2, we can use the following set covers $\{s_1, s_2\}$, $\{s_1, s_3\}$, $\{s_2, s_3\}$, $\{s_4\}$ and the last set can be activated twice. On the other hand, when the sensing range is r_1 we have a maximum lifetime of four, since the target t_2 is covered only by the sensor s_3 and a sensor can participate in at most four set covers. Therefore, this example shows a 16%–33% lifetime increase when using adjustable sensing ranges.

Figure 2   Five set covers: $C_1 = \{(s_1, r_1), (s_2, r_2)\}$,
$C_2 = \{(s_1, r_2), (s_3, r_1)\}$, $C_3 = \{(s_2, r_1), (s_3, r_2)\}$,
$C_4 = \{(s_4, r_2)\}$, and $C_5 = \{(s_1, r_1), (s_2, r_1), (s_3, r_1)\}$

(a)   (b)   (c)

(d)   (e)
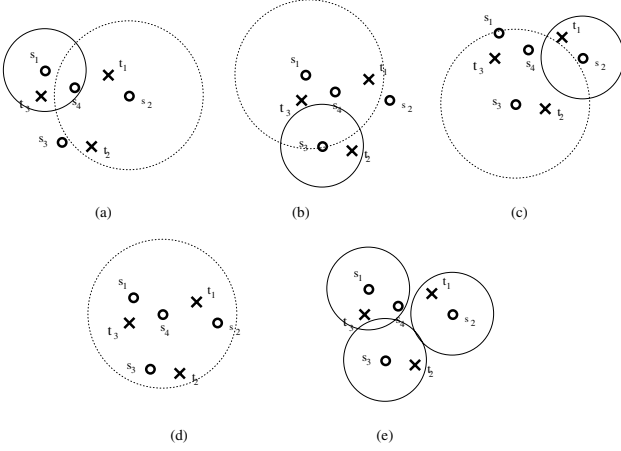
4   Solutions for the AR-SC problem

In this section, we present three heuristics for solving the AR-SC problem. In Section 4.1, we formulate the problem using integer programming and then solve it using *relaxation* and *rounding* techniques. In Section 4.2, we propose a greedy heuristic, where both centralised and distributed (localised) solutions are given for computing the set covers.

The centralised heuristics are executed at the BS. Once the sensors are deployed, they send their coordinates to the BS. The BS computes and broadcasts back the sensor schedules. In the distributed and localised algorithm, each sensor node determines its schedule based on communication with one-hop neighbours.

In this paper, we do not consider the energy cost caused by transmission of the sensing data. The sensing data can be collected by mobile robots, such as the data mules proposed in Chakrabarti et al. (2005) and Shah et al. (2003) and thus no energy is consumed by the sensors for the transmission of the sensed data.

4.1   Integer programming-based heuristic

In this subsection, we first formulate the AR-SC problem using integer programming in Section 4.1.1 and then present the LP-based heuristic in Section 4.1.2.

4.1.1   Integer programming formulation of the AR-SC problem

Given:

- N sensor nodes $s_1, \ldots, s_N$

- M targets $t_1, t_2, \ldots, t_M$

- P sensing ranges $r_1, r_2, \ldots, r_P$ and the corresponding energy consumption $e_1, e_2, \ldots, e_P$

- initial sensor energy E

- the coefficients showing the relationship between sensor, radius and target: $a_{ipj} = 1$ if sensor s_i with radius r_p covers the target t_j.

For simplicity, we use the following notations:
- i: ith sensor, when used as index
- j: jth target, when used as index
- p: pth sensing range, when used as index
- k: kth cover, when used as index.

Variables:

- c_k, boolean variable, for $k = 1 \ldots, K$; $c_k = 1$ if this subset is a set cover, otherwise $c_k = 0$

- x_{ikp}, boolean variable, for $i = 1, \ldots N, k = 1, \ldots K$, $p = 1, \ldots P$; $x_{ikp} = 1$ if sensor i with range r_p is in cover k, otherwise $x_{ikp} = 0$.

Maximise $c_1 + \cdots + c_K$
s.t.

$$\sum_{k=1}^{K} \left(\sum_{p=1}^{P} x_{ikp} e_p \right) \leq E \qquad \text{for all } i = 1, \ldots N$$

$$\sum_{p=1}^{P} x_{ikp} \leq c_k \qquad \text{for all } i = 1, \ldots N, \\ k = 1, \ldots K$$

$$\sum_{i=1}^{N} \left(\sum_{p=1}^{P} x_{ikp} \times a_{ipj} \right) \geq c_k \quad \text{for all } k = 1, \ldots K, \\ j = 1, \ldots M$$

$$x_{ikp} \in \{0, 1\} \text{ and } c_k \in \{0, 1\}$$

Remarks:

1   *K represents an upper bound for the number of covers*

2   *The first constraint, $\sum_{j=1}^{K} \left(\sum_{p=1}^{P} x_{ikp} e_p \right) \leq E$ for any $i = 1, \ldots, N$, guarantees that the energy consumed by each sensor i is less than or equal to E, which is the starting energy of each sensor.*

3   *The second constraint, $\sum_{p=1}^{P} x_{ikp} \leq c_k$ for any $i = 1, \ldots, N$ and $k = 1, \ldots, K$, assures that, if sensor i is part of the cover k then exactly one of its P sensing ranges are set.*

4   *The third constraint, $\sum_{i=1}^{N} \left(\sum_{p=1}^{P} x_{ikp} \times a_{ipj} \right) \geq c_k$ for any $k = 1, \ldots, K$ and $j = 1, \ldots, M$, guarantees that each target t_j is covered by each set c_k.*

4.1.2   LP-based heuristic

In this subsection we propose a heuristic to solve the AR-SC problem. In Section 4.1.1, we presented the Integer Programming-(IP-) based formulation. Since IP is NP-hard, we propose to use a relaxation and rounding mechanism. We first *relax* the IP to Linear Programming (LP), solve the LP in polynomial time and then *round* the solutions in order to get a feasible solution for the IP.
Relaxed Linear Programming:

Maximise $c_1 + \cdots + c_K$
s.t.

$$\sum_{k=1}^{K} \left(\sum_{p=1}^{P} x_{ikp} e_p \right) \leq E \qquad \text{for all } i = 1, \ldots, N$$

$$\sum_{p=1}^{P} x_{ikp} \leq c_k \qquad \text{for all } i = 1, \ldots, N, \\ k = 1, \ldots, K$$

$$\sum_{i=1}^{N} \left(\sum_{p=1}^{P} x_{ikp} \times a_{ipj} \right) \geq c_k \quad \text{for all } k = 1, \ldots, K, \\ j = 1, \ldots, M$$

$$0 \leq x_{ikp} \leq 1 \qquad \text{for all } i = 1, \ldots, N, \\ k = 1, \ldots, K, \\ \text{and } p = 1, \ldots, P$$

$$0 \leq c_k \leq 1 \qquad \text{for all } k = 1, \ldots, K$$

LP-based heuristic

1: solve the LP and get the optimal solution $\bar{x}_{ikp}$ and $\bar{c}_k$
2: set $\bar{x}'_{ikp} = 0$ and $\bar{c}'_k = 0$ for all $i = 1, \ldots, N$, $k = 1, \ldots, K$, $p = 1, \ldots, P$
3: sort $\bar{c}_k$ in non-increasing order $\bar{c}_1, \bar{c}_2, \ldots, \bar{c}_K$
4: **for all** variable $\bar{c}_k$ taken from the list in non-increasing order **do**
5:   **if** $\bar{c}_k > 0$ **then**
6:    /∗ try to build a set cover if $\bar{c}_k > 0$ ∗/
7:     sort $\bar{x}_{ikp}$, $i = 1, \ldots, N$, $p = 1, \ldots, P$ in non-increasing order
8:    **for all** $\bar{x}_{ikp}$ **do**
9:    **if** $\bar{x}_{ikp}$ covers new targets and sensor i has at least e_p energy at the beginning of setting up the cover $\bar{c}'_k$ **then**
10:     set up the range of sensor i to r_p, $\bar{x}'_{ikp} = 1$
11:    **else**
12:     $\bar{x}'_{ikp} = 0$
13:    **end if**
14:   **end for**
15:   **if** all targets are covered by $\bar{x}'_{ikp}$ having value 1 **then**
16:    /∗ we formed a valid set cover ∗/
17:    set $\bar{c}'_k = 1$
18:     update residual energy of any sensor i with range r_p in $\bar{c}'_k$: $E_i = E_i - e_p$
19:   **else**
20:    set $\bar{c}'_k = 0$ and reset $\bar{x}'_{ikp} = 0$ for any $i = 1, \ldots, N$ and $p = 1, \ldots, P$
21:   **end if**
22: **end if**
23: **end for**
24: return the total number of set covers $\sum_{k=1}^{K} \bar{c}'_k$

The heuristic starts in line 1 by solving the relaxed LP that outputs the optimal solution $\bar{x}_{ikp}$ and $\bar{c}_k$. We round this solution to get a feasible solution $\bar{x}'_{ikp}$ and $\bar{c}'_k$ for the IP. We use a greedy approach, by giving priority to the set covers with a larger $\bar{c}_k$. When adding sensors to a cover $\bar{c}'_k$, priority is given to the sensors with larger $\bar{x}_{ikp}$. We sort values $\bar{c}_k$ in the non-increasing order. In lines 8,…,14, we add sensors to the current set cover k, by adding first the sensors with higher $\bar{x}'_{ikp}$ values. If, later, the same sensor with a larger range is encountered, the new range setting is used if new targets are covered and if the sensor has sufficient energy resources for this setting. If all the targets are covered by the selected sensors in this set cover, then we set $\bar{c}'_k = 1$. Otherwise, forming the current set cover was unsuccessful, $\bar{c}'_k = 0$, and all of set k's members are removed ($\bar{x}'_{ikp} = 0$ for any $i = 1, \ldots, N$ and $p = 1, \ldots, P$).

The complexity of this algorithm is dominated by the linear programming solver. The best performance is $O(n^3)$ using Ye's algorithm (1991), where n is the number of variables. In our case $n = K(1 + NP)$, where P is usually a small number.

4.2 Greedy-based heuristics

In this subsection we propose two greedy solutions for the AR-SC problem. The centralised solution is given in Subsection 4.2.1 followed by a distributed and localised solution in Subsection 4.2.2.

4.2.1 Centralised greedy heuristic

In this subsection we present a centralised greedy heuristic. We use the following notations:

- T_{ip}: the set of uncovered targets within the sensing range r_p of sensor i.

- B_{ip}: the contribution of sensor i with range r_p. $B_{ip} = |T_{ip}|/e_p$.

- ΔB_{ip}: the incremental contribution of the sensor i when its sensing range is increased to r_p. $\Delta B_{ip} = \Delta T_{ip}/\Delta e_p$, where $\Delta T_{ip} = |T_{ip}| - |T_{iq}|$ and $\Delta e_p = e_p - e_q$. The range r_q is the current sensing range of the sensor i, thus $r_p > r_q$. Initially, all the sensors have assigned a sensing range $r_0 = 0$ and the corresponding energy is $e_0 = 0$.

- C_k: the set of sensors in the kth cover.

- $\overline{T}_{C_k}$: the set of targets uncovered by the set C_k.

The algorithm selects sensors in a greedy fashion, based on their contribution values. A contribution parameter B_{ip} is associated with each (sensor, range) pair. For brevity, in cases of no ambiguity, we write (i, p) instead of (s_i, r_p). Intuitively, a sensor that covers more targets per unit of energy should have higher priority in being selected in a sensor cover. We are using the incremental contribution parameter ΔB_{ip}, defined at the beginning of this subsection, as the selection decision parameter.

In our algorithm, we are concerned not only with selecting the sensors of each set cover, but also with determining their sensing ranges. Intuitively, a smaller sensing range is preferable as long as the target coverage objective is met, since energy resources are conserved, allowing the sensor to be operational longer.

Our algorithm repeatedly constructs set covers, as long as each target is covered by at least one sensor with enough energy resources. In forming a set cover, sensors are selected repeatedly, giving priority to the sensors with the highest contribution. We assume that initially all the sensors have been assigned the range $r_0 = 0$. If a sensor i is selected based on its contribution ΔB_{ip}, its sensing range is increased to r_p. Once the set cover is formed (e.g. all targets are covered by the selected set of sensors), the sensors with a sensing range greater than zero form the set of active sensors, while all other sensors with sensing range r_0 will be in the sleep mode.

Assume that a sensor (i, p) with the highest contribution ΔB_{il} is selected to be added to the current set cover. Then the sensor i updates its sensing range from r_p to r_l. For each sensor node s_x that covers at least one target in T_{il}, we update $T_{xu} = T_{xu} - T_{il}$ and ΔB_{xu} for any range r_u greater than the current sensing range of s_x. Note that although there are P sensing ranges for each sensor, we maintain contribution values only for those sensing ranges for which sufficient residual energy is available. For example, if the residual energy E_x of the sensor s_x satisfies the relation $e_q \le E_x < e_{q+1}$, then we consider only the contributions ΔB_{xu} for $u \le q$.

We present next the *Centralised Greedy Algorithm* that repeatedly constructs set covers as long as each target is covered by at least one sensor node with sufficient residual energy.

Centralised Greedy Algorithm

1: set the residual energy of each sensor s_i to E, $E_i = E$
2: assign to each sensor s_i a range $r_0 = 0$ having the corresponding energy $e_0 = 0$
3: $k = 0$
4: **while** each target is covered by at least on sensor (i, p) and $E_i > e_p$
5:      /* a new set cover will be formed */
6:      $k = k + 1$
7:      $\overline{T}_{C_k} = \{t_j | j = 1 \ldots, m\}$
8:          for each sensor s_i compute ΔB_{ip} and T_{ip}, for all sensing ranges that can be set up with the current residual energy
9:      **while** $\overline{T}_{C_k} \neq \emptyset$ **do**
10:     /* more targets have to be covered */
11:     select the sensor (i, p) with the highest contribution value ΔB_{il}
12:     increase sensor's s_i sensing range from r_p to r_l
13: $\overline{T}_{C_k} = \overline{T}_{C_k} - T_{il}$
14:     **for all** (x, u) such that $T_{xu} \cap T_{il} \neq \emptyset$ **do**
15:     /* update the uncovered target set and the incremental contribution */
16:         update $T_{xu} = T_{xu} - T_{il}$
17:         update $\Delta B_{xu} = \Delta T_{xu}/\Delta e_u$
18:     **end for**
19: **end while**
20: **for all** $(i, p) \in C_k$ **do**
21:     update the residual energy of sensor s_i, $E_i = E_i - e_p$
22: **end for**
23: **end while**
24: output the number of set covers k

The complexity of Centralised Greedy Algorithm is $O(MN^2PE/e_1)$. The number of iterations of the while loop (lines 4,…,23) is upper-bounded by $N\frac{E}{e_1}$, corresponding to the case when all the targets are covered by all sensors with range r_1. The complexity of the inner while loop (lines 9, …, 19) is upper-bounded by MNP.

4.2.2 *Distributed and localised heuristic*

In this subsection, we extend the algorithm introduced in Subsection 4.2.1 to a distributed and localised version. We use the notations introduced in the previous subsection. By 'distributed and localised' we refer to a decision process at each node that only makes use of information from a neighbourhood within a constant number of hops. A distributed and localised algorithm is desirable in wireless sensor networks since it adapts better to dynamic and large topologies.

The distributed greedy algorithm runs in rounds. Each round begins with an initialisation phase, where sensors decide whether they will be in an active or sleep mode during the current round. The initialisation phase takes W time, where W is far less than the duration of a round. Each sensor maintains a waiting time, after which it decides its status (sleep or active) and its sensing range, and then it broadcasts the list of targets it covers to its one-hop neighbours. The waiting time of each sensor s_i depends on s_i's contribution and is set up initially to $W_i = (1 - B_{iP}/B_{max}) \times W$ where B_{max}

is the largest possible contribution, defined as $B_{max} = M/e_1$, where M is the number of targets.

The waiting time can change during the initialisation phase, when broadcast messages are received from neighbours. If a sensor s_i receives a broadcast message from one of its neighbours, then s_i updates the set of uncovered targets T_{iP} and sets up its sensing range to the smallest value r_u needed to cover this set of targets. The sensor contribution value is also updated to B_{iu}. If all s_i's targets are already covered by its neighbours, then s_i sets up its sensing range to $r_0 = 0$. The waiting time W_i of the sensor s_i is also updated to $(1 - B_{iu}/B_{max}) \times W$. At the end of its waiting time, a sensor broadcasts its status (active or sleep) as well as the list of targets it covers. If its sensing range is r_0, then this sensor node will be in the sleep mode, otherwise it will be active during this round.

As different sensors have different waiting times, this serialises the sensors' broadcasts in their local neighbourhood and gives priority to the sensors with higher contribution. These sensors decide their status and broadcast their target coverage information first. In this algorithm we use a discrete time window, where d is the length of the time slot. Thus, the time window W has W/d time units. If the waiting times of two sensor s_i and s_j are too close, that is, $|W_i - W_j| < d$, then the sensors that are neighbours to both s_i and s_j cannot tell from whom the message was received, thus they will not update their uncovered target set.

We assume sensor nodes are synchronised and the protocol starts by having the BS broadcast a start message. If, after the initialisation phase, a sensor s_i cannot cover one of the targets in the set T_{iP} and its waiting time reached the value 0, then s_i sends this failure information to BS. In our algorithm, we measure the network lifetime as the time until BS detects the first failure.

Next we present the *Distributed Greedy Initialisation*, that is run by each sensor $s_i, i = 1, \ldots, N$ during the initialisation phase:

Distributed Greedy Initialisation (s_i)

1: compute the waiting time W_i and start timer t
2: **while** $t \leq W_i$ and $T_{iP} \neq \emptyset$ **do**
3:      **if** message from neighbour sensor is received **then**
4:          update T_{iP} and set-up the sensing range to the smallest value r_u needed to cover T_{iP}
5:          **if** $T_{iP} == 0$ **then**
6:              set s_i's sensing range to r_0
7:              break
8:          **end if**
9:          update s_i's contribution to B_{iu}
10:         update the waiting time W_i to $(1 - \frac{B_{iu}}{B_{max}}) \times W$
11:     **end if**
12: **end while**
13: /* assume s_i's sensing range was set up to r_u */
14: **if** $r_u == r_0$ **then**
15:     s_i broadcasts its sleep state decision

16:     return

17:     **end if**

18:     **if** $E_i < e_u$ **then**

19:         s_i reports failure to BS, indicating the targets it cannot cover due to the energy constraints

20:     **end if**

21:     s_i broadcasts information about the set of targets T_{iu} it will monitor during this round

22:     return

The complexity of the Distributed Greedy Initialisation procedure is $O(\frac{W}{d}NMP)$. This corresponds to the case when s_i receives messages from N neighbours, each d time. The updates for each message take $O(MP)$.

5   Simulation results

In this section, we evaluate the performance of LP-based and greedy-based heuristics. We simulate a stationary network with sensor nodes and targets randomly located in a $100\,m \times 100\,m$ area. We assume sensors are homogeneous and initially have the same energy. In the simulation, we consider the following tunable parameters:

- N the number of sensor nodes. In our experiments we vary N between 25 and 200.

- M the number of targets to be covered. It varies between 10 to 50.

- P sensing ranges $r_1, r_2,..., r_P$. We vary P between 1 and 6, and the sensing range values between $5\,m$ and $30\,m$.

- Energy consumption model $e_p(r_p)$. We evaluate the network lifetime under linear ($e_p = \Theta(r_p)$) and quadratic ($e_p = \Theta(r_p^2)$) energy consumption models.

- Time slot d in the distributed greedy heuristic. d shows the impact of the transfer delay on the performance of the distributed greedy heuristic. We vary d between 0.25 and 1 with the increment of 0.25.

In the first experiment in Figure 3, we compare the network lifetime computed by LP-based, centralised greedy and distributed greedy heuristics when we vary the number of sensors. We consider 10 targets randomly deployed, and we vary the number of sensors between 25 and 100 with an increment of 5. Each sensor has two adjustable sensing ranges, $15\,m$ and $30\,m$. The energy consumption model is linear.

Network lifetime results returned by the heuristics are close and they increase with sensor density. When more sensors are deployed, each target is covered by more sensors, thus more set covers can be formed.

In the second experiment in Figure 4, we adopt the centralised greedy heuristic to evaluate the impact of the number of adjustable sensing ranges on the network lifetime in a network with 40 targets randomly distributed. In the experiment, the number of sensors varied between 50 and 150 with an increment of 10. We compare 5 adjustable sensing

range sets, whose sizes are 1, 2, 3, 5 and 6, respectively. In each set with size P, the adjustable sensing ranges are $r_1 = \frac{30}{P}, r_2 = \frac{30}{P} \times 2, \dots, r_{P-1} = \frac{30}{P} \times (P-1), r_P = 30$. For example, in case of $P = 6$, the adjustable sensing range set is $\{r_1 = 5m, r_2 = 10m, r_3 = 15m, r_4 = 20m, r_5 = 25m, r_6 = 30m\}$. Note that the maximum sensing range in all the sets is $30\,m$ and $P = 1$ is the case when all sensor nodes have a fixed sensing range with value $30\,m$.

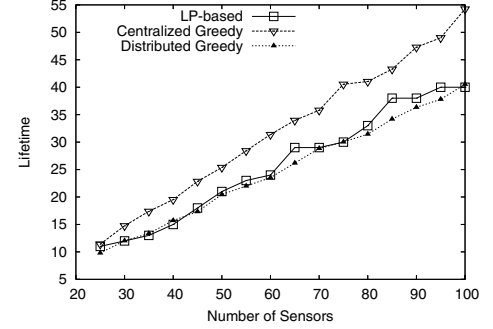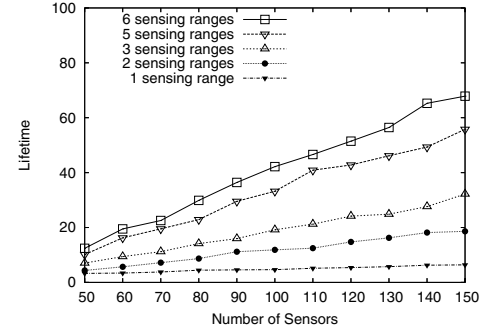**Figure 3** Network lifetime with number of sensors

Figure 4 Network lifetime for different sensing range values

Simulation results indicate that adjustable sensing ranges have great impact on the network lifetime. This simulation results also justify the contribution of this paper, showing that adjustable sensing ranges can greatly contribute to increasing the network lifetime.

In Figure 5, we compare the network lifetime produced by centralised and distributed greedy algorithms. We measure the network lifetime when the number of sensors varies between 50 and 150 with an increment of 10 and the number of targets is 50. Each sensor has six sensing ranges with value $5\,m$, $10\,m$, $15\,m$, $20\,m$, $25\,m$ and $30\,m$. The energy consumption model is linear. We change the length of the time slot d in the distributed greedy algorithm to $d = 0.25$, 0.5, 0.75 and 1. In Figure 5, we use the curve labelled as $d = 0$ to denote the centralised greedy heuristic.

Network lifetime produced by the centralised algorithm is longer than that produced by the distributed algorithm. This happens because the centralised greedy heuristic has global information and can always select the sensor with the greatest contribution. Also, if there is a tie between the contribution of different sensors, the centralised greedy heuristic can break the tie arbitrarily, without any additional cost.

**Figure 5**   Network lifetime for different values of the time
slot d
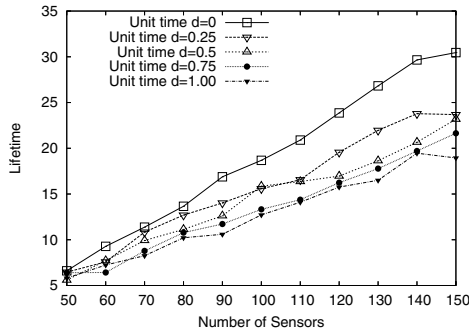
In the distributed heuristic, breaking a tie happens at the expense of backoff time, and there is also no guarantee of no conflict. A conflict occurs when sensors broadcast at the same time based on their contributions. Then, there might be sensors that work instead of going to the sleep state, even if the targets within their sensing range are already covered. As illustrated in Figure 5, the transfer delay also affects the network lifetime. The longer the transfer delay is, the smaller the lifetime.

In Figure 6, we study the impact of two energy models on the network lifetime computed by the distributed greedy heuristic when we vary the number of sensors between 40 and 200 and the number of targets is 25 or 50. Each sensor has $P = 3$ sensing ranges with values $10, 20$ and 30 m. The two energy models are the linear model $e_p = c_1 \times r_p$, and quadratic model $e_p = c_2 \times r_p^2$. In this experiment we defined constants $c_1 = 0.01$ and $c_2 = 0.05$, where $E = 10$ is the sensor starting energy. For both energy models, the simulation results are consistent and indicate that the network lifetime increases with the number of sensors and decreases as more targets have to be monitored.

Figure 6   Linear and quadratic energy models
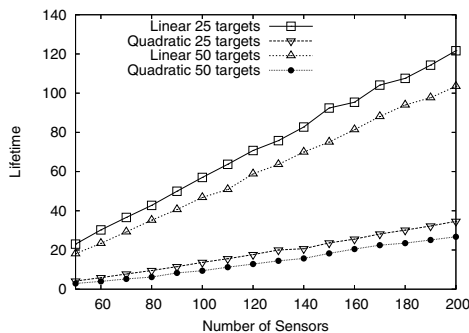
We give an example of coverage produced by centralised and distributed heuristics in Figure 7, where 'o' and '•' are inactive (sleeping) and active sensors, respectively and '+' are targets. The upper left subgraph is the sensors and targets deployment. The upper right subgraph is a set cover produced by the centralised greedy heuristic. The bottom subgraph is a set cover produced by the distributed greedy heuristic. We assume a 100 m $\times\, 100$ m area, with 40 sensors and 20 targets. Each sensor has $P = 3$ sensing ranges with values $10, 20$ and 30 m. We use solid lines to represent $r_1 = 10$ m, dashed lines

for $r_2 = 20$ m and dotted lines for $r_3 = 30$ m. We used a linear energy model. The first subgraph represents the sensors' and targets' random deployment. The second and third subgraph show set covers produced by the centralised and distributed greedy heuristics. The active sensors are blackened and the line type indicates the sensing range value.

Figure 7   Set covers example
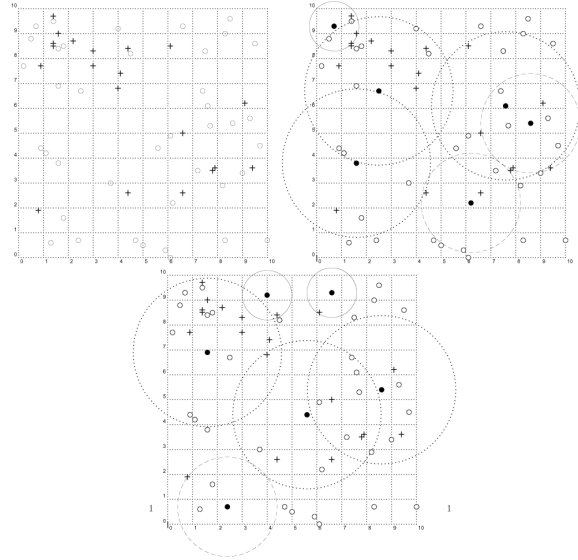
The simulation results can be summarised as follows:

- Given the number of targets and the sensing range values, the network lifetime output by our heuristics increases with the number of sensors deployed.

- Network lifetime increases with the number of adjustable sensing ranges.

- Even if the two centralised solutions perform better than the distributed solution (longer network lifetime), using a distributed and localised heuristic is an important characteristic for a solution in wireless sensor networks environment.

- Transfer delay used for internode communication in the distributed greedy heuristic affects the network lifetime. Smaller transfer delays result in longer network lifetime.

- For both linear and quadratic energy models, the network lifetime increases with the number of sensors and decreases as more targets have to be covered.

6   Conclusions

In this paper, we proposed scheduling models for the target coverage problem for wireless sensor networks with adjustable sensing range. The problem addressed in this paper is to determine maximum network lifetime when all targets are covered and sensor energy resources are constrained.

In this paper, we introduced the mathematical model, proposed efficient heuristics (both centralised and distributed and localised) using integer programming formulation and greedy approaches and verified our approaches through simulation.

In our future work, we will integrate the sensor network connectivity requirement. Maintaining connectivity among the selected sensors has an advantage in facilitating the exchange of information between sensors and the base station.

Acknowledgement

References

Akyildiz, I.F., Su, W., Sankarasubramaniam, Y. and Cayirci, E. (2002) 'A survey on sensor networks', *IEEE Communications Magazine*, pp.102–114, August.

Cardei, M. and Du, D-Z. (2005) 'Improving wireless sensor network lifetime through power aware organization', *ACM Wireless Networks*, Vol. 11, No. 3, May.

Cardei, M., Thai, M., Li, Y. and Wu, W. (2005) 'Energy-efficient target coverage in wireless sensor networks', *IEEE INFOCOM 2005*, March.

Cardei, M. and Wu, J. (2006) 'Energy-efficient coverage problems in wireless ad hoc sensor networks', *Computer Communications*, Vol. 29, No. 4, pp.413–420.

Carle, J. and Simplot, D. (2004) 'Energy efficient area monitoring by sensor networks', *IEEE Computer*, Vol. 37, No. 2, pp.40–46.

Chakrabarti, A., Sabharwal, A. and Aazhang, B. (2005) 'Data collection by a mobile observer in a single-hop sensor network', *ACM Transactions on Sensor Networks*.

Cheng, M., Ruan, L. and Wu, W. (2005) 'Achieving minimum coverage breach under bandwidth constraints in wireless sensor networks', *IEEE INFOCOM 2005*, March.

Garey, M.R. and Johnson, D.S. (1979) *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman.

Huang, C-F. and Tseng, Y-C. (2003) 'The coverage problem in a wireless sensor network', *ACM MobiCom'03*, pp.115–121, September.

Megerian, S., and Potkonjak, M. (2002) 'Wireless sensor networks', J. G. Proakis (Ed). *Wiley Encyclopedia of Telecommunications*, December.

Meguerdichian, S., Koushanfar, F., Potkonjak, M. and Srivastava, M. (2001) 'Coverage problems in wireless ad-hoc sensor networks', *IEEE Infocom 3*, pp.1380–1387.

Photoelectric Sensors, (2000) Available at: http://www. schneider-electric.ca/www/en/products/sensors2000/html/ osiris.htm.

Shah, R., Roy, S., Jain, S. and Brunette, W. (2003) 'Data MULEs: modeling a three-tier architecture for sparse sensor networks', *IEEE SNPA Workshop*, May.

Tian, D. and Georganas, N.D. (2002) 'A coverage-preserving node scheduling scheme for large wireless sensor networks', *Proceedings of the First ACM Workshop on Wireless Sensor Networks and Applications*.

Wang, X., Xing, G., Zhang, Y., Lu, C., Pless, R. and Gill, C.D. (2003) 'Integrated coverage and connectivity configuration in wireless sensor networks', *First ACM Conference on Embedded Networked Sensor Systems*.

Wu, J. and Yang, S. (2004) 'Coverage and connectivity in sensor networks with adjustable ranges', *International Workshop on Mobile and Wireless Networking (MWN)*, August.

Ye, Y. (1991) 'An $o(n^3l)$ potential reduction algorithm for linear programming', *Mathematical Programming*, Vol. 50, pp.239–258.

Zhang, H. and Hou, J.C. (2004) 'Maintaining sensing coverage and connectivity in large sensor networks', *NSF International Workshop on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless and Peer-to-Peer Networks*, February.