IET Wireless Sensor Systems

Research Article

Maximising network lifetime for target coverage problem in wireless sensor networks

ISSN 2043-6386
Received on 23rd June 2015
Revised on 25th May 2016
Accepted on 5th June 2016
doi: 10.1049/iet-wss.2015.0094
www.ietdl.org

Manju[1] ✉, Satish Chand[2], Bijender Kumar[1]

[1]*Department of Computer Engineering, Netaji Subhash Institute of Technology, New Delhi 110078, India*
[2]*School of Computer and System Sciences, Jawaharlal Nehru University, New Delhi, 110067, India*
✉ *E-mail: manju.nunia@gmail.com*

Abstract: Target coverage problem is one of the important problems in wireless sensor network in which each target should be covered by at least one sensor. All the sensors are organised into different groups called sensor cover in such a way that each cover can monitor all the targets for a fixed duration. By keeping one sensor cover active at a time while others in sleep mode, sensors batteries can be utilised in efficient way. This helps in prolonging the network lifetime. In this study, the authors propose a new energy-efficient heuristic to schedule the sensors in different non-disjoint sensor covers which helps to maximise network lifetime. At first, the authors' heuristic identify all the *critical targets* (least covered) and the *critical sensors* (covering critical targets). The *critical targets*, covered by minimum number of sensors, will be the targets that become uncovered first. Utilising *critical sensors* efficiently will help to increase the network lifetime. In their method, they try to select minimum number of *critical sensors* in each sensor cover so that the *critical targets* can be covered for longer period. Simulation results shows that the proposed heuristic outperforms many existing approaches for solving target coverage problem.

1 Introduction

Wireless sensor networks (WSNs) have gained a lot of interest by the research community in the past few years. As the name suggests, a WSN consists of a large number of small, light-weight, highly energy-constrained and inexpensive devises called sensors. Sensors are equipped with batteries, sensing hardware, processing and storage resources [1]. WSNs design is highly influenced by many factors which includes fault-tolerance, scalability, network topology, hardware constrains, transmission media and power consumption [2]. These factors are important because they serve as guidelines to design a protocol/algorithm for various purposes of the WSN. When compared with traditional *ad hoc* networks, the WSN have some limitations such as limited battery power, computational capacity and memory. Sensor's energy resources are generally irreplaceable or may be deployed with non-rechargeable batteries. Due to the limited battery power, the sensor nodes die one after another during the networks operations. Therefore, all the network requirements must be met with minimum power consumption due to battery limitations. In most of the applications, it is impossible to replenish energy resources. Thus, the efficient use of available energy resources directly impacts the lifetime of WSN.

Coverage requirement in WSNs is to provide continuous monitoring over specified area/targets for longest possible time with limited energy resources. Coverage problems are basically classified as area and target coverage problem [3–5]. In this paper, we consider only target coverage problem where objective is to monitor a set of fixed targets for longest possible time. In order to provide coverage, a sensor can be in two states; *active* or *sleep*. In *active* state, it can cover targets by consuming a certain amount of its battery power. In *sleep* state, sensor basically stays idle and does not deplete energy. For better coverage, the sensor nodes are densely deployed to cover these target points. Dense deployment not only helps to improve a sensor network's reliability [6] but also extends its longevity. In order to keep all the targets fully covered for maximum possible time, a common approach is not to activate all the sensors at once, instead, activate a subset of sensors which covers all the targets for a fixed activation period. These subsets are known as *sensor covers* and their activation duration is called *working time*. The basic idea is to activate these sensor covers consecutively for a fixed working time to monitor all the targets. The sum of all the covers activation time is called *network lifetime*.

There has been lots of research work [3–16] done on target coverage problem and its variant. The work done in [5, 13] is an improvement over work done in [4] for target coverage problem, yet, they have not given enough attention to those targets which are least covered in the WSNs. In this paper, we propose a new energy-efficient heuristic for the target coverage problem where we try to improve over the total gain in network lifetime time achieved in [5, 13]. We first identify all *critical targets* (those are covered by least number of sensors) and all the *critical sensors* (those covering critical targets). We maximise the network lifetime by giving priority to those sensors which has more remaining energy, covers maximum number of uncovered target and in each sensor cover we make sure that there should be minimum *critical sensor*. Our proposed rules will help to keep alive the *critical sensors* for longer time that will in turn increase the network lifetime.

The rest of this paper is organised as follows: Section 2 defines the target coverage problem addressed in this paper and discusses some earlier work on the same problem. Section 3 includes various definitions required in the subsequent part of paper. In Section 4, we describe the proposed heuristic in detail and explain its functionality. Simulation results and discussions on the performance are presented in the Section 5. Section 6 put a conclusion remark and gives guidelines for future scope.

2 Target coverage problem

In this section, we first define the target coverage problem of finding the maximum number of non-disjoint sensor covers in WSNs. Moreover, then, we formulate the problem as a linear programming problem. In the later part of the section, we present some related work on the target coverage problem.

2.1 Problem statement

Suppose we have an area of size $M \times M$ in which there is a target set $T = \{t_1,\ldots,t_n\}$ which consists of n number of target which are randomly placed in the above specified area. To monitor these targets for maximum possible time, we randomly deploy a set of sensor $S = \{s_1,\ldots,s_m\}$ with fix sensing range R_s. Every sensor s_i is assigned an initial battery life b_i. A target t_j, $1 \le j \le n$, is said to be covered if it is falling within the sensing range of at least one sensor s_i, $1 \le i \le m$. A sensor can be in either *active* state or *sleep* state. In the *active* state, the sensor can sense information of covered targets and, in the *sleep* mode it conserves its energy and does not monitor the targets. We try to find out the maximum number of non-disjoint sets of the sensors (called sensor cover), which are of course upper bounded by $C_{\max}$, so that, at a given point of time only one sensor cover is active. Let C be the set of the entire sensor covers, i.e. $C = \{C_1,\ldots, C_{\max}\}$. Each sensor cover C_k ($C_k \in C$) is enough to cover all the targets in the network. The life time of a sensor cover C_k, $X(C_k)$, cannot exceed the remaining energy of a sensor in C_k which has the minimum battery lifetime, i.e. $X(C_k) = \mathrm{Min}_{si\epsilon Ck}(b_i)$. The objective of the target coverage problem is to generate maximum number of sensor covers to prolong network lifetime.

Energy-efficient target coverage problem [4] is to maximise the sum of all sensor covers lifetime (i.e. sum of $X(C_k)$) (we shall refer to x_k) with the constraints that none of the sensor (s_i) can be used longer than assigned initial battery life (b_i). Hence, the target coverage problem is to find the complete family of sensor covers which has maximum aggregated network lifetime among all the families of sensor covers. The linear programming (LP) problem of energy-efficient target coverage problem is as follows:

$$\begin{aligned} &\text{Maximise} \quad \sum_p x_p \\ &\text{Subject to} \quad \sum_p B_{ip} x_p \le b_i \quad \text{for all sensors } s_i \\ &\qquad\qquad\quad x_p \ge 0 \quad \text{for all sensor covers } C_p. \end{aligned}$$

where $\boldsymbol{B}$ is the constraint matrix, which is defined as

$$B_{ij} = \begin{cases} 1 & \text{if sensor } s_i \text{ is in sensor} - \text{cover } C_j \\ 0 & \text{otherwise} \end{cases}$$

If the complete set (C) of all sensor covers is known in advance then, the constraint matrix $\boldsymbol{B}$ is explicitly known. However, as discussed earlier, most of the time the sensor network is densely deployed, which results in more number of columns of matrix $\boldsymbol{B}$ (sensor covers). Therefore, none of the regular algorithm for solving the given linear programming can be applied as $\boldsymbol{B}$ is not known in advance. The other method is to generate sensor covers (columns of $\boldsymbol{B}$) as and when necessary using column generation method applied for linear programming. All the known methods [3–8, 13, 17, 18] discussed in the following section for energy-efficient target coverage problem follow this column generation approach.

Slijepcevic and Potkonjak [3] solved the target coverage problem heuristically which initially finds poorly covered targets and then start selecting sensor in a sensor cover such that for each such target, there is only one selected sensor. If more than one sensor is covering the same target, then, the heuristic ascertains to exclude rest of the sensors from the same cover. The sensors covers generated so far are disjoint covers where each sensor can participate only in one sensor cover. Due to this limited participation of sensors, the network lifetime achieved was significantly less.

Further, to increase the network lifetime, Cardei *et al.* [4] presented a greedy heuristic which generate non-disjoint sensor covers where a sensor can participate in more than one cover. In order to select sensor in a sensor cover, this heuristic gives priority to that sensor which covers maximum number of uncovered targets. Cardei *et al.* [4] were first to represent the target coverage problem as maximum network lifetime problem (MLP) and which is further proved to be NP-complete, and proposed an approximation algorithm to solve it.

To solve the above said MLP, column generation methodology is best suitable as suggested in [5, 6]. Manju [5] proposed a high-energy-efficient heuristic which generate non-disjoint covers by selecting only highest remaining energy sensors and then minimise the generated sensors cover to eliminate coverage redundancy. Later, a hybrid approach is proposed by Deschinkel [6] which solve the MLP using column generation technique by either heuristically or optimally both using integer linear programming (ILP) formulation.

Lu *et al.* [8] addresses another heuristic with the constraint that a sensor can cover only one target at a time. Further, a LP technique is proposed by Berman *et al.* [7]. Their approach is slightly different in the sense that a series of sensor covers are generated first to deduce the optimal working time for each sensor cover. They also proposed an $(1+\varepsilon)f$ approximation based on Garg and Konemann approximate scheme [18]. Mini *et al.* [13] proposed a weight based heuristic where sensor covers are first generated by selecting high remaining energy sensors and then optimise the sensor cover. By doing so, if theoretical UB on the network lifetime is not achieved, then, they assign pre defined weights to each sensor based on the coverage of sensors and those target which are covered by a sensor. Moreover, then, recalculate the network lifetime by considering these weights to decide whether a sensor can be part of a sensor cover or not. Kim *et al.* [17] addressed a branch and bound algorithm which increases the network lifetime by utilising repeated activation of these sensor covers for small fragments of working time. Zorbas *et al.* [9] proposed an algorithm based on pre calculated weights on sensors in which all the sensor nodes are categorised into four classes namely *Best, Good, Ok* and *Poor* using their coverage information. Then, it first select sensors from the *Best* class to generate sensor cover and if there are not sufficient sensors in *Best* class, then, it chooses sensors from *Good*, *Ok* and *Poor* class onwards. Zorbas *et al.* [12] again shown the detailed methodology adopted by a greedy target coverage algorithm to works. In their proposed heuristic, they maximise the number of sensor covers by efficiently managing the individual sensor's coverage and their association with the poorly covered targets.

Most of the above discussed methods have considered all the targets almost equally. As discussed earlier, there are the targets in the network which are covered by least number of sensors called *critical target*. These targets need special attention so that they can be monitored for longer period. In other words, the sensors covering these targets should be kept alive for possible longest time. In this paper, we propose a new method which tries to look after these critically covered targets so that network lifetime can be further improved over earlier works [4, 5, 13].

3 Motivation

As we know that any linear programming problem is solvable in polynomial time, but, it does not mean that the target coverage problem is also polynomial time solvable. This is because the number of sensor covers generated in densely deployed networks may be exponentially large. Thus, to maximise the network lifetime, here we propose an energy-efficient heuristic for the given target coverage problem. For better understanding of the remaining paper, we provide some required definitions and assumptions.

Definition 1: Sensor–target coverage relationship matrix
The sensor-target coverage relationship matrix $\boldsymbol{R}_{m\times n}$ is defined as follows:

$$\boldsymbol{R}_{ij} = \begin{cases} 1 & \text{if sensor } s_i \text{ covers target } t_j \\ 0 & \text{otherwise} \end{cases}$$

Definition 2: Critical target set (T_{critical})
A target t_j is *critical target* if sum of the entire sensor's battery lifetime covering it is minimum among all the targets in T

$$j = \arg\min \sum_i \boldsymbol{R}_{ij} b_i.$$

Definition 3: Critical sensor set (S_{critical})
A sensor s_i is said to be *critical sensor* if it covers at least one *critical target*.

Definition 4: Sensor cover
Given R, a sensor cover C_k, is a set of rows of R, such that, for every column j, there is a row i in C_k such that $R_{ij} = 1$. A sensor cover C_k is a minimal cover if for any cover C_k', C_k'$\subseteq C_k$ if and only if C_k' $= C_k$.

Definition 5: Lifetime of a sensor cover
The maximum lifetime which can be assigned to a sensor cover C_k, is the smallest available battery lifetime of sensors which are part of it. Thus

$$X(c_k) = \operatorname*{Min}_{S_i \in C_k} (b_i)$$

Definition 6: Network lifetime
The total network lifetime (L) can be defined as the sum of all the generated sensor covers lifetime

$$L = \sum_{k=1}^{|C|} X(C_k)$$

Definition 7: Upper bound (UB)
The maximum number of non-disjoint sensor covers cannot exceed the maximum number of full sensor cover sets that satisfy the coverage constraint. Therefore, the maximum number of full cover sets ($C_{\max}$) can be used as the upper limit of the number of non-disjoint sensor covers. Finding the maximum number of full sensor cover subsets is an NP-complete problem [4], but, we can compute the UB on the number of full sensor cover sets with the following method:

As discussed earlier, sensors are densely deployed in WSNs. In case of target coverage problem, network is assumed to be alive till each target is covered by at least one sensor. Since, *critical targets* are covered by the minimum number of sensors; therefore, the first target going to be uncovered is among *critical target*(*s*). Thus, the numbers of *critical sensors* can be used to estimate the UB on the number of full cover subsets as follows:

Considering a network of $\boldsymbol{m}$ sensors $\{s_1, s_2, \ldots, s_m\}$ with their respective energies as $\{b_i, b_2, \ldots, b_m\}$and $\boldsymbol{n}$ targets $\{t_1, t_2, \ldots, t_n\}$. The maximum coverage time for target t_j denoted by U_j, is given by the following equation

$$U_j = \sum_{i=1}^{m} \boldsymbol{R}_{ij} b_i$$

where $\boldsymbol{R}_{ij}$ is the sensor-target coverage relationship matrix. Now, the upper bound on the network lifetime, denoted by UB, is given by the following equation

$$\text{UB} = \min\{U_1, U_2, \ldots, U_n\}$$

We illustrate the estimation on UB on the network lifetime in WSNs. Let R be a sensor-target coverage relationship matrix which has three targets (t_1, t_2 and t_3) and four sensors (s_1, s_2, s_3 and s_4) as follows:

s/t	t_1	t_2	t_3
s_1	0	1	1
s_2	1	1	0
s_3	1	0	1
s_4	1	0	1

We assume all the sensors have same sensing range and initially loaded with 1 unit battery life (b_i). Each sensor which covers targets set is as follows, $s_1 = \{t_2, t_3\}$, $s_2 = \{t_1, t_2\}$, $s_3 = \{t_1, t_3\}$ and $s_4 = \{t_1, t_3\}$. According to Definition 2, t_2 is *critical target* which is covered by only two sensors whose total battery life is 2 units. Similarly, the corresponding sensors covering t_2, which are s_1 and s_2 become *critical sensors*. The UB on the network lifetime can be calculated by summing up all the battery lifetimes available to cover *critical target* (t_2), which is equal to 2 units.

As we discussed earlier, the network is said to be alive till all the targets are covered by at least one sensor each. We also know that the first target(s) which is going to be uncovered is *critical target*(*s*). Therefore, we propose to design an energy-efficient heuristic in which we try to avoid selecting more sensors which are covering *critical target*. In our proposal, we select only minimal number of *critical sensors* which are required to cover all the *critical target* (*s*). Moreover, then, we avoid selecting any more sensors from *critical sensor* set, so that the network will be alive for longer period. For rest of the uncovered targets, we give priority to those sensors which has more remaining energy and covers maximum uncovered targets. The works in [4, 5, 13] have provided methods to improve the network lifetime; however, they have not put enough attention on *critical targets*. Attending *critical targets* in efficient way, the network lifetime can be further improved over [4, 5, 13]. In the next section, we discuss the proposed heuristic to maximise the network lifetime.

4 Proposed heuristic

The network lifetime can be maximised by scheduling sensors in energy-efficient way to be part of a sensor cover. Paper by Manju [5] considered the remaining energy only for generating the sensor cover. The problem in this work is that the highest remaining energy sensor will defuse first, creating early energy holes in the network. The work [13] considered two parameters, but only one at a time, i.e. either the remaining energy or the target coverage to construct the sensor covers. First, it calculates the network lifetime by selecting sensors with highest remaining energy and compares it with the UB (Definition 7). If the obtained network lifetime is less than the UB then, the lifetime is recalculated using the target coverage. Depending on the parameter used, it has the problem of energy holes or the coverage holes.

There are many other parameters also in the energy constrained sensor networks which affects the network lifetime. Few of them include *critical target*'*s* coverage, individual target coverage in the networks and the number of sensors covering *critical targets*. The *critical target* coverage is the crucial parameter because *critical targets* will be left uncovered first in the networks which in turn exhaust the network at early stage. If sensors covering these *critical targets* (*critical sensors*) are wisely used, then the network lifetime can be enhanced.

To follow this observation, our proposed heuristic smartly limits the usages of *critical sensors* which in turn results in improved network lifetime than that of [5, 13]. As observed earlier, *critical targets* and sensors covering these targets plays an important role in improving the network lifetime. Our heuristic first identify all the *critical targets* and *critical sensors* and use them in constructing the sensor cover. Once the *critical targets* have been identified, we make the sensor covers as follows:

(a) select the minimum number of *critical sensors* covering all the *critical targets*;
(b) for the remaining uncovered targets, find highest remaining energy sensor from the remaining sensor set;
(c) select the sensors obtained in above step that have maximum target coverage.

Our heuristic keep selecting sensors in this manner until all the targets have been covered. The generated sensor cover may have redundant target coverage. Since the sensors have limited energy, we should remove those sensors whose coverage is already covered by other sensors in the sensor cover. We discarded extra sensors using *minimise* function (*discussed later*). Once the sensor cover has been minimised, we use this sensor cover for monitoring the targets. Assuming the sensors in the above constructed sensor cover to be active for the $X(C_k)$ unit, the remaining energy of all the sensors in current sensor cover is reduced by $X(C_k)$ unit. In order to avoid early energy holes or the coverage holes, we construct the new sensor covers using the above given procedure until network is not exhausted. We present our heuristic in algorithmic form as follows Figs. 1 and 2.

As discussed earlier, our proposed heuristic collectively takes three parameters to schedule sensors in each sensor cover whereas the works in [5, 13] consider only single parameter. We show the efficacy of our proposed method over the works in [5, 13] by illustrating with the following example. We consider three targets t_1, t_2, t_3 and four sensors s_1, s_2, s_3, s_4 with 1 unit battery lifetime. The sensor-target coverage relationship matrix $\boldsymbol{R}$ is given below. The UB of this network is 2 units (refer matrix $\boldsymbol{R}$).

s/t	t_1	t_2	t_3
s_1	0	1	1
s_2	1	1	0
s_3	1	0	1
s_4	1	0	1

Heuristic in [5] generates only one sensor cover, $C_1 = \{s_1, s_2\}$ that provides network lifetime of 1 unit.

The work [13] also generate same sensor cover $C_1 = \{s_1, s_2\}$ (choosing highest energy sensor first) and results in same network lifetime as in [5]. Since the achieved network lifetime is less than the UB (2 units for above given R), it recalculates the network lifetime by changing the priority of sensors based on the weight (1) as given below:

$$wi = \sum_{j=1}^{n} \frac{Rij}{\sum_{i=1}^{m} Rij} \tag{1}$$

where n, m and R_{ij}, $1 \le i \le m$, $1 \le j \le n$, refer to number of targets, number of sensors and sensor-target coverage relationship matrix.

We obtained the sensors' weights using (1) as follows: $W(s_1) = 0.83$, $W(s_2) = 0.83$, $W(s_3) = 0.66$ and $W(s_4) = 0.66$. We select the sensors having the highest weight for constructing the sensor covers. On the basis of these weights, we get same sensor cover $C_1 = \{s_1, s_2\}$ which has 1 unit of network lifetime. Now, we apply our proposed method on the same example. There is only one *critical target* t_2 and there are two sensors s_1, s_2 that cover t_2 and hence s_1 and s_2 are *critical sensors*. We select only one *critical sensor*, say s_1, to construct the sensor cover, say C_1 (using rule ***a***). There are two remaining uncovered targets t_1 and t_3 and two non-*critical sensors* s_3 and s_4 have the same energy, covering both the targets. So, we select either sensor, say s_3 to construct the sensor cover. Thus, we get sensor cover, say, $C_1 = \{s_1, s_3\}$. We can generate one more sensor cover, say, $C_2 = \{s_2, s_4\}$. The network lifetime can be obtained by activating C_1 followed by C_2 or C_2 followed by C_1; thus providing 2 units of total network lifetime (i.e. equal to UB). From this example, we have illustrated that our method provides network lifetime better than that of [5, 13]. In the next section, we discuss the simulation results for different scenarios to show the effectiveness of the proposed heuristic.

```
Proposed Heuristic: for Target Coverage Problem
Input:   S: set of all Sensors
         T: set of all targets
         R: sensor-target coverage relationship matrix
         bi: battery life of all si∈S
Output:  L: Network-Lifetime
         C: set of all sensor covers
initialise   L= Ø, C= Ø
Calculate Scritical, Tcritical, UB (using definition 2, 3 & 7)
    //Tcritical: set of critical targets
    //Scritical: set of sensors covering Tcritical
    //UB: theoretical upper bound on the network
  // Generate sensor cover Ck
for k=1toUB do
    Calculate Scritical and Tcritical
    while Tcritical≠ Ø
        select highest energy sensor si ∈ Scritical which covers maximum uncovered targets in Tcritical
        Ck=Ck∪{si}
        for all target t covered by si do
            T critical=T critical-{t}
        end for
    end while
    T=T-{Tcritical}
    while T≠ Ø
      select highest remaining energy sensor si∈(S-Scritical) which covers  maximum uncovered targets in T
      Ck=Ck∪{si}
      for all target t covered by si do
        T=T-{t}
      end for
    end while
  Minimise Sensor Cover Ck (using Minimise () )
    X(Ck) = Min si∈Ck (bi)
    for all si in minimal_Ck do
        bi=bi-X(Ck)
    end for
     L=L+X(Ck)
   C=C∪{minimal_Ck}
end for
return C, L
```

Fig. 1 *Target coverage problem*

```
Minimise(): Minimise Sensor Cover C_k
  Input:   S: set of all Sensors
           C_k: Sensor Cover
  Output:  minimal_C_k
  initialise minimal_C_k = Ø,
  for i=length(C_k) down to 1 do
        // s_i.C_k is i^th sensor in C_k
      if C_k-(s_i.C_k) still a sensor cover then
            C_k=C_k- s_i.C_k        // ignore s_i.C_k
      else
            minimal_C_k=minimal_C_k ∪s_i.C_k
      end if
  end for
  return minimal_C_k
```

Fig. 2 *Minimise sensor cover C_k*

5 Simulation results

In this section, we show how good the proposed heuristic through various simulation outcomes. For simulation, we take a square area of size 200×200 M where targets (n) and sensors (m) are randomly deployed. Sensor's coordinates are randomly generated by pseudo-random number generator function. Each value in simulation is an average of ten instances on randomly generated test cases. We assume that the sensors are initially loaded with 1 unit battery lifetime and for each sensor cover, we take $X(C_k) = 0.25$ unit. Pre calculation of theoretical UB helps us to analyse how far the proposed heuristic is successful. To show the superiority of the proposed heuristic, we compare its performance with greedy heuristic (named G-MSC**)** by Cardei *et al.* [4], high-energy-first (named HEF) heuristic in [5] and heuristic in [13]. We performed all the simulations in Matlab (R2009) on a core i3 processor with 2.10 GHz processor and 4 GB RAM system. Complete simulation outcomes are shown through four experiments (1 to 4) as given below.

5.1 Experiment 1

In this section, we compare the network lifetime obtained by the proposed heuristic with respect to the theoretical UB. Here, we take fixed targets (25) and assume homogenous sensors (in terms of energy and sensing range) ranging from 50 to 250 with sensing range $R_s = 70$M. Table 1 shows the network lifetime achieved by the proposed heuristic under specified simulation environment for simple coverage where a target is said to be covered if and only if it is covered by at least one sensor.

As depicted in Table 1, we can observe that proposed heuristic achieves network lifetime almost equal to the UB calculated. This happens due to the selection of minimal required sensors from *critical sensor* set. By avoiding sensor selection from *critical sensor* set, we can keep alive them for longer time period which results in longer network lifetime.

5.2 Experiment 2

Target Q-coverage [10, 11, 13] is a variant of target coverage problem where each target in $T = \{t_1,\ldots,t_n\}$ should be covered by at least $\boldsymbol{Q} = \{q_1,\ldots,q_n\}$ number of sensor nodes. Similarly, target K-coverage problem [13] is another variant where a target is said to be covered if and only if it is covered by at least K-number of sensors. Here, we can say that target K-coverage problem is a special case of target Q-coverage problem where $q_j = K$ for all $1 \le j \le n$. To support this feature, we too simulated the proposed heuristic for $K = 2$ (each target covered by at least two sensors), and $K = 3$ (where each target is covered by at least three sensors). To do so, we need to recalculate theoretical UB on the given

Table 1 Network lifetime achieved by the proposed heuristic when compared with UB

Sensors	Proposed heuristic	Upper bound	Percentage gain, %
50	07.50	07.80	96.15
100	19.35	19.60	98.73
150	24.45	25.00	97.80
200	35.65	36.00	99.02
250	42.30	42.40	99.76

network which is different from target coverage problem. In case of Q-coverage, we assume the same network with $\boldsymbol{n}$ number of target set $T = \{t_1,\ldots,t_n\}$ and $\boldsymbol{m}$ number of sensor set $S = \{s_1,\ldots,s_m\}$ with initial battery life b_i. A sensor s_i, $1 \le i \le m$, can cover a target t_j, $1 \le j \le n$, only if it is in the sensing range (R_s) of the sensor. The UB is calculated for target Q-coverage as follows:

$$\text{UB} = \min j \left[\frac{Rij \times bi}{qj} \right] \tag{2}$$

To show the performance of the proposed heuristic under various K-coverage requirements, we calculated network lifetime for fix targets (25) and sensor fixed to 150. The sensing range (70 M) is same for all the sensors. We considered homogenous network in terms of battery lifetime (1 unit) and sensing range (70 M) assigned to each sensor. We take $X(C_k) = 0.25$ units for each sensor cover. The network lifetime achieved by our heuristic under various K-coverage requirements is shown in Fig. 3.

It is clearly shown in Fig. 3 that the network lifetime achieved by the proposed heuristic is always near to optimal UB under various K values. While increasing K, the network lifetime decreases because more number of sensors is needed to cover same set of targets which results in less network lifetime.

5.3 Experiment 3

In this simulation scenario, we compare the network lifetime achieved by the proposed heuristic, G-MSC [4], HEF heuristic in [5] and heuristic in [13]. In G-MSC [4], sensor covers are generated by selecting sensors with more remaining energy that covers maximum uncovered targets. HEF heuristic [5] always selects sensors with highest remaining energy first to form a sensor cover. Moreover, then, minimises the generated cover to remove redundant sensors. Initially, heuristic in [13] generates sensor cover in the way same as HEF and check whether the theoretical UB is achieved. By doing so, if obtained network lifetime is less than the UB calculated, then, the network lifetime is recalculated by choosing sensors with highest weight W (1). In order to calculate the total network lifetime, we simulated for fix targets (25) and sensors ranging from 50 to 250. We consider homogenous network in terms of energy (1 unit) and sensing range (70M) assigned to each sensor. We take $X(C_k) = 0.25$ units for each sensor cover. Fig. 4 shows the network lifetime achieved by the proposed heuristic, G-MSC [4], HEF [5] and heuristic in [13].

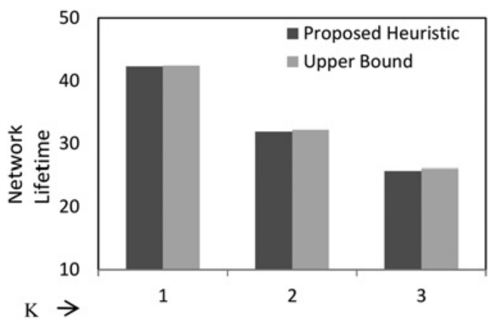

Fig. 3 *Network lifetime achieved by the proposed heuristic and UB on the network for different coverage requirements (K = 1, K = 2 and K = 3)*

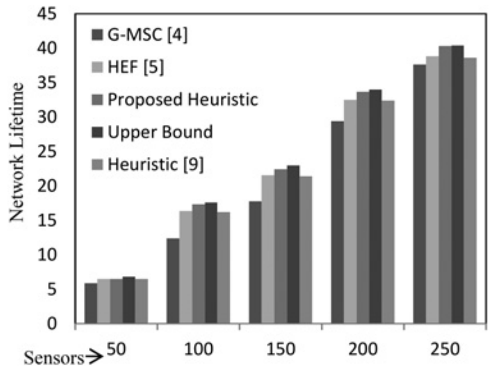

Fig. 4 *Comparison of network lifetime achieved by the proposed heuristic, G-MSC [4], HEF [5] and heuristic in [13] with theoretical UB on the network given*

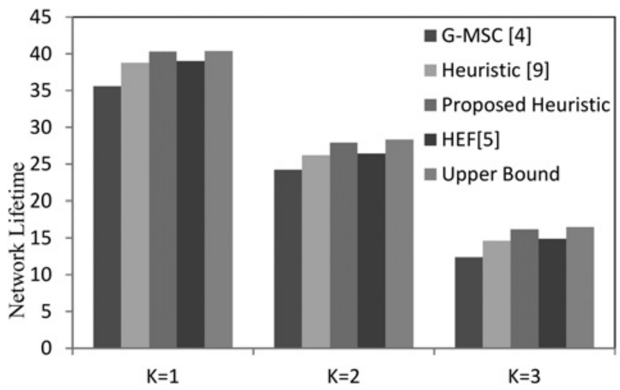

Fig. 5 *Comparison of network lifetime achieved by the proposed heuristic, G-MSC [4], HEF [5] and heuristic in [13] with theoretical UB on the network given*

As shown in Fig. 4, the proposed heuristic outperforms the G-MSC [4], HEF [5] and heuristic in [13] in terms of obtained network lifetime. It is also depicted in Fig. 4, that the proposed heuristic always achieves network lifetime nearly equal to the theoretical UB. By increasing number of sensors in the same area, network lifetime achieved also increases. This happens so, because, with more sensors, the same number of targets will be covered by multiple sensors which results in more number of sensor covers. As discussed in Section 1, that total network lifetime is the sum of working time of all the covers, therefore, with more sensor covers, the network lifetime also increases.

5.4 Experiment 4

Here, we compare the performance of the proposed heuristic with the G-MSC [4] and heuristic in [13] under different coverage requirements ($K = 1$, $K = 2$ and $K = 3$) for fix targets (25) and fixed sensors (250) in the same environment. We consider homogenous sensors with sensing range 70 M and unit battery lifetime. The UB for different coverage requirement ($K = 1$, $K = 2$ and $K = 3$) is calculated using (2) to evaluate the performance of all three methods as shown in Fig. 5.

It is clearly depicted in Fig. 5 that, the proposed heuristic always outperforms G-MSC [4], HEF [5] and heuristic in [13] under various K values. The total network lifetime decreases when there is increase in K value. This happens so because with more K, each target needs more sensors to be part of sensor cover which results in less number of total sensors covers which in turn decreases the network lifetime.

6 Conclusion

In this paper, we have proposed a new heuristic for the target coverage problem. The proposed heuristic basically tries to keep alive *critical sensors* in the network to maximise network lifetime. It is done so, because, we know that the network will get exhausted when even a single target becomes uncovered. It is also to note that the *critical targets* are the ones which will be left uncovered first in the networks. Therefore, one can maximise the network lifetime only by limiting the usages of those sensors covering *critical targets*. Our heuristic performs better than the existing works [4, 5, 13]. It achieves the network lifetime near to theoretical UB on the network for target coverage problem. We have also discussed the Q-coverage and K-coverage variant of target coverage problem. In both the above cases, the proposed algorithm outperforms the existing ones. In future, we can extend this work for partial coverage and connected coverage variant of target coverage problem.

7 References

1 Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., *et al.*: 'A survey on sensor networks', *IEEE Commun. Mag.*, 2002, **40**, (8), pp. 102–114
2 Jin, R., Che, Z., Wang, Z., *et al.*: 'Battery optimal scheduling based on energy balance in wireless sensor networks', *IET Wirel. Sens. Syst.*, 2015, **5**, (6), pp. 277–282
3 Slijepcevic, S., Potkonjak, M.: 'Power efficient organization of wireless sensor networks'. Proc. IEEE Int. Conf. on Communications, 2001, pp. 472–476
4 Cardei, M., Thai, M.T., Li, Y., *et al.*: 'Energy-efficient target coverage in wireless sensor networks'. Proc. IEEE Infocom, 2005
5 Manju, A.K.P.: 'High-energy-first (HEF) heuristic for energy efficient target coverage problem', *Int. J. Ad Hoc Sens. Ubiquit. Comput.*, 2011, **2**, (1), pp. 45–58
6 Deschinkel, k.: 'A column generation based heuristic for maximum lifetime coverage in wireless sensor networks'. SENSOR COMM11, Fifth Int. Conf. on Sensor Technologies and Applications, 2011, vol. 4, pp. 209–214
7 Berman, P., Galinescu, G., Shan, C., *et al.*: 'Power efficient monitoring management in sensor networks'. Proc. IEEE Wireless Communications and Networking Conf., 2004, pp. 2329–2334
8 Lu, M., Wu, J., Cardei, M., *et al.*: 'Energy-efficient connected coverage of discrete targets in wireless sensor networks'. Proc. Third Int. Conf. on Networking and Mobile Computing, 2005, vol. 3619, pp. 43–52
9 Zorbas, D., Glynos, D., Kotzanikolaou, P., *et al.*: 'B{GOP}:an adaptive algorithm for coverage problems in wireless sensor networks'. Thirteenth European Wireless Conf., EW, 2007
10 Gu, Y., Liu, H., Zhao, B.: 'Target coverage with QoS requirements in wireless sensor networks'. IPC, 2007, pp. 35–38
11 Chaudhary, M., Pujari, A.K.: 'Q-coverage problem in wireless sensor networks'. Proc. Int. Conf. on Distributed Computing and Networking, 2009, pp. 325–330
12 Zorbas, D., Glynos, D., Kotzanikolaou, P., *et al.*: 'Solving coverage problems in wireless sensor networks using cover sets', *Ad Hoc Netw.*, 2010, **8**, pp. 400–415
13 Mini, S., Udgata, S.K., Sabat, S.L.: 'Sensor deployment and scheduling for target coverage problem in wireless sensor networks', *IEEE Sens. J.*, 2014, **14**, (3), pp. 636–644
14 Kumar, D.: 'Performance analysis of energy efficient clustering protocol for maximizing lifetime of wireless sensor networks', *IET Wirel. Sens. Syst.*, 2014, **4**, pp. 9–16
15 Harb, H., Makhoul, A., Tawil, R., *et al.*: 'Energy-efficient data aggregation and transfer in periodic sensor networks', *IET Wirel. Sens. Syst.*, 2014, **4**, pp. 149–158
16 Pujari, A.K., Mini, S., Padhi, T., *et al.*: 'Polyhedral approach for lifetime maximization of target coverage problem'. ICDCN 2015, pp. 14:1–14:8
17 Kim, Y.-H., lee, H.-J., han, Y.-H., *et al.*: 'Branch and bound algorithm for extending lifetime of WSN'. Proc. IEEE Vehicular Technology Conf., 2009, pp. 1–5
18 Garg, N., Könemann, J.: 'Faster and simpler algorithms for multi-commodity flow and other fractional packing problems'. Proc. 39th Annual Symp. on Foundations of Computer Science (FOCS), 1998, pp. 300–309