

Satisfying coverage and connectivity in bandwidth constrained sensor networks

Dimitrios Zorbas and Christos Douligeris

`jim@students.cs.unipi.gr`

Abstract

One of the main problems in wireless sensor networks is the successful coverage of a number of targets lying on a field and at the same time the maximization of the lifetime of the network. Many algorithms have been proposed in the literature in order to find the maximum number of disjoint or non-disjoint sets of sensors (cover sets), where one set can be active at any one time. The complexity of the problem increases if we consider that the channel bandwidth of the base station, where the sensor nodes transmit their monitoring data, is limited. The limited channel bandwidth leads to cover sets that cannot monitor all the available targets resulting in a coverage breach.

In this paper, we propose an algorithm that computes cover sets that minimize the overall coverage breach. This algorithm takes into account several problem parameters, such as the connectivity requirement between the sensors and the base station, the coverage status of the nodes and their remaining battery life. Our experimental results show that our algorithm outperforms existing approaches in various performance metrics and that the connectivity constraint sets an upper bound on the algorithm's output.

1 Introduction

Wireless sensor networks (WSNs) can be effectively used in various applications. One of them is the surveillance of a number of discrete targets lying on a field, where in most cases, the sensor nodes and the targets are randomly distributed on the field. The sensors collect the monitoring data and transmit them to a wired node that is located outside of the WSN, which is called the base station (BS).

Due to the limited battery lifetime of the sensor nodes, efficient techniques and protocols have been proposed in the literature, aiming at prolonging the total monitoring time of the targets [7, 8, 2, 3, 12]. Most of these algorithms divide the sensor nodes into a number of disjoint or non-disjoint sets, called *cover sets*. Only one cover set is active at each moment of time. By maximizing the number of cover sets, the network lifetime will be also maximized.

However, in applications where there is a need of continuous monitoring, such as the military or the detection applications [1, 6], the base station must receive the data at well defined time instances. A delayed information delivery may result to a loss of information. Moreover, a sufficient number of available time slots (or channels) is required, to ensure that the application receives a high data rate. Existing protocols such as TDMA, can split the interval between two successive deliveries, in a certain number of time slots. Each time slot refers to a maximum number of data bits that can transmit each active node to the BS in a single shared channel. Thus, the nodes of a cover set cannot transmit a higher number of bits than what the time division protocol assigns, since it could cause a delay to the information delivery. In other words, the amount of data that can be transmitted from an active cover set is fixed. The authors of [5] call this restriction “the bandwidth constraint”.

If we assume that each target generates a constant data rate, the bandwidth constraint can lead to uncovered targets in the cover sets. The targets that remain uncovered at the end of the construction of a cover set, are called “breached” [5]. The objective of a target coverage algorithm is to minimize the total number of breached targets and, at the same time, to maximize the network lifetime of the remaining targets. At this point it should be mentioned that this problem becomes more acute if we consider that no data aggregation can be done by the sensors. In this case a node that covers two targets must forward twice the packets of a node that covers one target, minimizing this way the number of available slots W .

In this paper we propose an algorithm that tries to address the above problem. The algorithm computes the cover sets, while it tries to minimize the breached targets. It, also, takes into account the critical requirement of connectivity and its communication cost, a constraint that is not mentioned in the corresponding literature. We propose a solution that computes the cover sets, while it can alternate the roles among the nodes using a weighted connected graph to avoid a quick node energy exhaustion. In order to compare the performance of our algorithm with other previously proposed approaches, we present simulation results that include the connectivity constraint and others that do not include it.

2 Related Work

The work of Cheng et al. [5] introduces the idea of the coverage breach problem. The authors formulate the minimum breach problem and they prove that it is NP-Complete. They transform the problem to an integer programming problem and present two heuristics to solve it. The authors consider the generated sets as disjoint only.

In [4] the authors present an extension to [5]. They analyze three instances of the problem; the minimum breach, the minimum individual breach time and the minimum maximal breach. The objective of the first instance is to find a user-given number of cover sets when the cardinality of each cover set

must be smaller than $W + 1$ and the total breach minimized. The other two instances consider a maximum allowed breach time and a maximum number of cover sets that must be computed. Two algorithms are proposed to solve the above problems. The first one is a greedy algorithm based on the coverage status of the sensor candidates, while the second one is a three-step heuristic algorithm that uses Integer Programming (IP) and Linear Programming (LP) techniques to compute the cover sets. The greedy algorithm outperforms the second approach called “Relaxation”.

Finally, in [10] the authors present two equivalent instances of the coverage breach problem with those of [4]. The objective in the first one is to achieve a maximum amount of total lifetime by minimizing the total breach time, while in the second one a maximum value of the breach time (or breach rate) is allowed, while the total lifetime must be maximized. The authors allow a sensor node to be a member of multiple cover sets. In order to solve the above instances of the problem they propose an LP-based algorithm and a greedy heuristic called “greedy-MSc”.

The above approaches consider that the sensors are found in one-hop distance from the BS, a non-practical situation since in practical WSNs, only a few nodes can communicate directly with the BS. Moreover, they do not take into account the communication cost, while they consider that a sensing node that covers a large number of targets holds only one slot of the time division protocol. Using a practical energy model, the transmission distances among sensors would rise, which eventually could lead to an inefficient energy usage in sensors, since energy usage increases exponentially with distance.

3 Problem description

In this section we formulate the connected coverage breach problem and present the radio and sensing model used in our solution.

3.1 Problem parameters

The sensor coverage model consists of a set of sensors $S_0 = \{s_1, s_2, \dots, s_n\}$ and a set of targets $T_0 = \{t_1, t_2, \dots, t_k\}$. Each target in T_0 is covered by at least one sensor node in S_0 . Also, the set S_R consists of sensors that can not monitor any target in T_0 .

The coverage algorithm produces a collection $C = \{C_1, \dots, C_m\}$ of m connected cover sets. Each cover set C_p is a subset of the available sensors ($C_p \subseteq S_0 \cup S_R$). Depending on the data rate that the targets produce, each sensor holds one or more slots of the W available in the time division protocol. The sum of the hold slots must not exceed W .

Finally, we use the value “total coverage breach rate” (*i.e.* tcb_r) to measure the number of the breached targets [10]:

$$tcb_r = \frac{\sum_{p=1}^{|C|} tcb_p}{m \cdot |T_0|}, \quad (1)$$

where tcb_p is the number of the targets of the p -th element of C that have remained uncovered due to the bandwidth constraint.

Next, we define two instances of the coverage breach problem. In the first instance, called Maximum Network Lifetime (**MNLT**), the objective is to minimize tcb_r , while at the same time making sure that total network lifetime is at least max_nl ($max_nl > 0$). The objective of the second instance, called Minimum Coverage Breach Rate (**MCBR**), is to maximize the network lifetime and achieve a breach rate that is not higher than max_tcb_r ($0 \leq max_tcb_r \leq 1$).

3.2 Radio and sensing model

The energy consumption model of the sensors takes into account the energy consumed by the sensing operation and the communication operation. We assume that a node spends a constant amount of energy in order to sense a data bit, called α_3 . Concerning the communication cost, we follow the radio model presented in [9]. According to [9], a node spends E_{tx} energy in order to transmit a bit to distance d and E_{rx} energy in order to receive a bit, where:

$$E_{tx} = \alpha_{11} + \alpha_2 d^f, E_{rx} = \alpha_{12}. \quad (2)$$

α_{11} is the energy/bit consumed by the transmitter electronics, α_2 accounts for energy dissipated in the transmit op-amp and α_{12} is the energy consumed by the receiver electronics.

4 The proposed solution

In this section, we present an efficient algorithm that computes the cover sets under the bandwidth constraint. It is called Coverage Breach Heuristic (CBH) (see Algorithm 1) and it uses several mechanisms in order to minimize the number of breached targets and maximize the network lifetime at the same time. CBH takes into account the connectivity constraint using a weighted connected graph. CBH solves the MNLT problem, but it can be easily transformed in order to solve the MCBR problem too (see Section 4.1).

CBH requires a number of elements in order to operate: (i) the set T_0 , (ii) the set S_0 , (iii) the set S_R , (iv) the sets N_i that contain the sensors that cover the target t_i , $\forall t_i \in T_0$, (v) the sets P_j that contain the targets that a sensor s_j can cover, $\forall s_j \in S_0$, (vi) the connected graph G that contains the sensors in S_0 and S_R , as well as the BS (vertices), (vii) the battery capacity of a sensor l , (viii) the desired time duration of a cover set τ , (ix) the desired network lifetime max_nl , (x) the bandwidth W , and (xi) the data rate DR_{t_i} , $\forall t_i \in T_0$.

The algorithm operates in iterations. At the end of each iteration it produces a cover set that satisfy the bandwidth and the connectivity constraints. Each iteration can be separated to a number of steps: During the ‘‘Bandwidth Applicability’’ step the algorithm builds a cover set C_p that can cover all or part of the available targets depending on the bandwidth constraint. The ‘‘Bandwidth Applicability’’ step includes the ‘‘Sensor Applicability’’ step, that gradually fills

Algorithm 1: Coverage Breach Heuristic

```

require:  $S_0 \neq \emptyset, S_R \neq \emptyset, N_i \neq \emptyset, \forall t_i \in T_0, P_j \neq \emptyset, \forall s_j \in S_0, T_0 \neq \emptyset, max\_nl > 0,$ 
 $\tau > 0, W > 0, DR_{t_i} > 0, \forall t_i \in T_0, l > 0, G$ 
1  $C = \emptyset$ 
2  $S_{avail} = S_0$ 
3  $T_{avail} = T_0$ 
4 foreach  $s_j \in S_0 \cup S_R$  do  $lifetime_j = l$ 
5 foreach edge  $u, v$  do  $w_{u,v} = (\alpha_{11} + \alpha_2 \cdot d_{uv}^f)\tau$ 
6 while  $\sum_{p=1}^{|C|} \tau_p < max\_nl$  do
    // Bandwidth Applicability Step
    7  $p := p + 1$ 
    8  $S_{cur} = S_{avail}$ 
    9  $T_{cur} = T_{avail}$ 
    10  $C_p = \emptyset$ 
    11  $\tau_p = 0$ 
    12  $W_t := 0$ 
    13 update the  $max\_neighbors$  value using  $G$ 
    14 while  $(T_{cur} \neq \emptyset)$  and  $(W_t < W)$  do
        15  $selected := none$ 
        16  $max\_CF := 0$ 
        // Sensor Applicability step
        17 select sensor  $selected$  with the highest  $CF$  value among all sensors in  $S_{cur}$ 
        18 if  $selected = none$  then  $W_t = W$ 
        19 foreach  $t_i \in P_{selected}$  do
            20 if  $\frac{DR_{t_i}W}{max\_data} + W_t \leq W$  then
                21  $T_{selected} = T_{selected} \cup \{t_i\}$ 
                22  $T_{cur} = T_{cur} - \{t_i\}$ 
                23  $W_t = W_t + \frac{DR_{t_i}W}{max\_data}$ 
            24  $S_{cur} = S_{cur} - \{selected\}$ 
            25  $C_p = C_p \cup \{selected\}$ 
            26 foreach  $s_j \in neighbors_{selected}$  do
                27  $w_{selected,s_j} = w_{selected,s_j} + \frac{l}{lifetime_{selected}}$ 
            28  $W_t := \lceil W_t \rceil$ 
        // Connection Applicability Step
        29 compute the SPT
        30 foreach  $s_j \in C_p$  do
            31  $C_p = C_p \cup \{\text{sensors on path from } s_j \text{ to BS}\}$ 
        32 compute  $t_i$  of the cover set  $C_p$ 
        33 foreach  $s_j \in C_p$  do
            34 update  $lifetime_j$ 
            35 if  $lifetime_j < 0$  then
                36 delete vertex  $s_j$  from  $G$ 
                37 foreach  $t_i \in P_j$  do
                    38  $N_i = N_i - \{s_j\}$ 
                    39 if  $|N_i| = 0$  then  $T_{avail} = T_{avail} - \{t_i\}$ 
                    40 update  $P_{j'}$  for each sensor  $s_{j'} \neq s_j$  covers  $t_i$ 
                41  $S_{avail} = S_{avail} - \{s_j\}$ 
        42 foreach vertex  $s_j$  of  $G$  do
            43 if  $s_j$  is not connected with the BS then delete vertex  $s_j$  from  $G$ 
            44  $S_{avail} = S_{avail} - \{s_j\}$ 
        45  $C = C \cup \{C_p\}$ 
        46 if  $|S_{avail}| = 0$  or  $|T_{avail}| = 0$  then return  $C$ 
    47 return  $C$ 

```

the C_p with sensors. Finally, during the ‘‘Connection Applicability’’ step all

nodes in C_p are connected with the BS through nodes from S_R .

Before starting to build cover sets, CBH initializes the following elements (lines 1-5): a) the set C , b) the set S_{avail} that initially contains all the sensors found in S_0 , c) the set T_{avail} that initially contains all the targets found in T_0 , d) the lifetime of each node in S_0 and S_R , and e) the weights of the edges of the graph G using the following formula:

$$weight_{u,v} = \alpha_{11} + \alpha_2 \cdot d_{uv}^f. \quad (3)$$

The assignment of the aforementioned weights, can assist in the selection of relay nodes that are found on the shortest possible path to the base station in “Connection Applicability” step.

Each iteration starts with the “Bandwidth Applicability” step that is responsible to produce a cover set C_p that meets the bandwidth constraint. To achieve this it uses the value W_t that checks whenever the bandwidth exceeds its maximum value W . Furthermore, it uses the sets S_{cur} and T_{cur} that contain the sensors and the targets, respectively, that will participate in the “Node selection” step. Their cardinality gradually decreases while nodes are selected for inclusion in C_p . During the “Sensor Applicability” step, each node is evaluated according to a cost function. The cost function takes into account several parameters of the problem that are related to a sensor candidate and the top scored node is then selected. The cost function depends on (a) the coverage status of a sensor, (b) the number of its neighbors, and (c) its remaining battery lifetime. Using parameter (a) we minimize the probability of double-covering a target and gradually decrease the network lifetime [12]. Parameters (b) and (c) are used to decrease the probability of producing cover set that could operate for a short time (and thus minimize the network lifetime).

The cost function CF is given by the following formula and is computed for each sensor $s_j \in S_{cur}$:

$$CF(T_{cur}, T_{avail}, P_j, neighbors_j, lifetime_j) = \frac{uncovered}{(covered + 1)} + \frac{neighbors_j}{max_neighbors} + \frac{lifetime_j}{l}.$$

The *uncovered* and the *covered* values denote the number of targets that the sensor candidate covers and the number of targets covered by previously selected sensors, respectively. To test the coverage status of a sensor and compute these values, the utility function $freq(P_j, T)$ is used. $freq(P_j, T)$ counts the targets in T that sensor s_j covers, *i.e.* $freq(P_j, T) = |P_j \cap T|$. The value *max_neighbors* denotes the maximum number of neighbors a node has among all sensors and it is computed at the beginning of each iteration.

The value W_t increases depending on the data rate of the selected targets and the maximum amount of data that we allow to transmit a cover set (*max_data*). The selected sensor is removed from S_{cur} and is added to the current cover set C_p . The weights of the graph between the selected sensor and its neighbors are updated according to formula (4) in order to avoid selecting the same node as

a relay node during the ‘‘Connection Applicability’’ step.

$$weight_{selected,v} = weight_{selected,v} + \frac{l}{lifetime_{selected}} \quad (4)$$

The ‘‘Bandwidth Applicability’’ step follows the ‘‘Connection Applicability’’ step, where the algorithm tries to connect the sensors of C_p to the BS. During this step, a *Shortest Path Tree (SPT)*, rooted at BS, is computed and all the intermediary nodes on the path from each sensing sensor to the BS are added to C_p . The weights of the graph between the members of C_p and their neighbors are further updated according to formula (4), avoiding the selection of the same paths in future iterations. The time duration of the cover set is then computed according to formula 5 [11], as one or more nodes of C_p may not be able to operate for the desired time duration τ (they may not have enough battery lifetime).

$$\tau_p = \min(\tau, \min(\frac{lifetime_j}{EC_j} \tau)), \forall s_j \in C_p, \quad (5)$$

where $lifetime_j$ is the energy of the sensor before the operation of the cover set and EC_j the energy consumed by the sensor during this cover set. This means that the cover set C_p will be active until one active node has depleted its battery.

The remaining lifetime of each sensor s_j of C_p is updated according to formulas (2), (5) and the functionality of a sensor in the cover set that is described in Table 1. SN and RL contain the sensors that are used for sensing and relay, respectively, while P_j contains the targets that a sensing node covers in C_p . PD_j contains all the targets that the descendant sensing nodes (in SPT) cover. If the examined sensor has no descendant sensors, then $PD_j = \emptyset$. A relay sensor must forward the data to the BS. The amount of the forward data r_j depends on the data rate DR of the targets that the sensor s_j or/and the descendant sensors cover ($r_j = \sum_{i=1}^{|P_j \cup PD_j|} DR_{t_i} \tau_p, t_i \in P_j \cup PD_j, s_j \in C_p$).

Table 1: Energy consumption of a sensor s_j in a cover set

Functionality of sensor s_j	Energy Consumed (EC_j)
$s_j \in SN, s_j \notin RL$	$\alpha_3 \cdot r_j + E_{tx} \cdot r_j$
$s_j \in SN, s_j \in RL$	$\alpha_3 \cdot r_j + E_{rx} \cdot r'_j + E_{tx} \cdot (r_j + r'_j)$
$s_j \notin SN, s_j \in RL$	$E_{rx} \cdot r_j + E_{tx} \cdot r_j$
$s_j \notin SN, s_j \notin RL$	0

A node s_j with no available lifetime is deleted from the available sensors set S_{avail} , from the graph G and from each set N_i that it is member of ($t_i \in P_j$). A target that may not be covered by any sensor anymore, is deleted from T_{avail} . Finally, all the sensor-vertices that are not connected with the BS are deleted from G and S_{avail} , while the current cover set C_p is added to the collection C .

The algorithm terminates whenever it reaches the desired network lifetime max_nl or whenever it runs out of targets or sensors. Its running time mainly depends on the complexity of the “Bandwidth Applicability” step and the complexity of the SPT computation. If all the available sensing nodes are used, the “Bandwidth Applicability” step is upper bounded by $\frac{l}{(\alpha_3 + E_{tx})\tau DR} \nu |S_0|^2 W$, where ν is the number of sensors that cover the most densely covered target. The complexity of the SPT computation is $O(|E| + (|S_0| + |S_R|) \log(|S_0| + |S_R|))$, where $|E|$ is the number of edges of G if Dijkstra’s algorithm is used. Hence, the total running time of the algorithm is $O(\frac{l}{(\alpha_3 + E_{tx})\tau DR} \nu |S_0|^2 W + \frac{l}{(\alpha_3 + E_{tx})\tau DR} \nu (|E| + (|S_0| + |S_R|) \log(|S_0| + |S_R|)))$.

4.1 Solving MCBR instance using CBH

CBH can be slightly transformed to solve the second instance of the examined problem (see Algorithm 2). It first generates cover sets that cover a large number of uncovered targets, different from the last generated sets. Thus, the breach rate shows a gradual upward trend from the beginning to the termination of the algorithm. When the total breach rate exceeds a user-given maximum value max_tcbr , the algorithm will have generated $|C|$ cover sets. By excluding the last generated cover set from C , the algorithm will have generated the accurate result within the bound of breach rate that the user has defined.

Algorithm 2: Coverage Breach Heuristic (MCBR)

```

require:  $S_0 \neq \emptyset, S_R \neq \emptyset, N \neq \emptyset, P \neq \emptyset, T_0 \neq \emptyset, max\_tcbr > 0, \tau > 0, DR_{t_i} > 0, \forall t_i \in T_0,$ 
            $W > 0, l > 0, G$ 
...
 $C = \emptyset$ 
 $C\_prev = \emptyset$ 
 $tcbr := 0$ 
// Breach Rate Check Step
while  $tcbr < max\_tcbr$  do
    ...
    while  $(T_{cur} \neq \emptyset)$  and  $(W_t < W)$  do
        // Bandwidth Applicability Step
        ...
        // Sensor Applicability Step
        foreach  $s \in S_{cur}$  do
            ...
        // Connection Applicability Step
        ...
         $C\_prev = C$ 
         $C = C \cup \{C_{cur}\}$ 
        update  $tcbr$ 
    if  $tcbr > max\_tcbr$  then
         $C = C\_prev$ 
return  $C$ 

```

5 Evaluation and Discussion of Results

In this section we simulate the proposed algorithm¹ and compare it to previously proposed approaches. The evaluation of our algorithm can be divided into two parts. In the first part we have transformed CBH in order to be able to compare it to the other approaches, as they consider that all the nodes are in one-hop distance and the energy consumption model is based on the number of occurrences of a node in the output cover sets. The objective for the MNLT instance, is to minimize the total coverage breach, while satisfying a number of produced cover sets equal to $\lceil \frac{|S_0|}{W} \rceil$ [10]. The objective for the MCBR instance, is to achieve a maximum number of cover sets, while at the same time satisfying a coverage breach rate not higher than a user-given value. In this part, the network lifetime is measured as the average number of the uniquely covered targets in the cover sets, called *covers*². We call “Greedy” the greedy algorithm presented in [4] and “Greedy-MS” the heuristic one presented in [10]. Because the authors of [4] do not present a case of the problem similar to MCBR, we use “Greedy” only for the MNLT instance.

In the second part, we compare CBH through an algorithm that considers that all the sensors are BS-connected, in order to assess the impact of the connectivity constraint and communication cost on the algorithm output. We call this algorithm “CBH 1-hop”. The radio and sensing model described in Section 3.2 is used here.

The simulated scenario involves 300 sensors and 40 targets lying on a square terrain. We run each simulation scenario 50 times, with random and uniform target and sensor deployment. We compute the average results of these runs and we vary the size of the terrain, keeping W constant. In this case, we assess the impact of the node and target deployment density on the algorithms’ output. In order to show the impact of the bandwidth on the network lifetime and on the breach rate, we vary the size of the bandwidth keeping a fixed terrain size. The error bars are also shown in the figures. The base station is always located at $(0,0)$, the communication range of the nodes is $50m$ and their corresponding sensing range is $10m$. Finally, for the second part of the evaluation, concerning the energy consumption model, we use the following parameters: $\alpha_3 = 100nJ/bit$, $\alpha_{11} = 50nJ/bit$, $\alpha_{12} = 100nJ/bit$, $\alpha_2 = 100pJ/bit/m^2$, $l = 20J$, $\tau = 1000sec$ and each target generates constant traffic with data rate from 1 to 20Kbps.

5.1 Comparing with the existing approaches

In Figure 1 we present the simulated results of the compared algorithms, where all the participating sensors can be found in a distance up to R_c from the base station. Despite the fact that all the algorithms produce the same number

¹The software is written in Perl language and can be found at <http://rainbow.cs.unipi.gr/projects/sensors>

²an algorithm can produce a large number of cover sets, but the total number of covers may remain low, because it may monitor only a small part of the available targets.

of cover sets (*i.e.* $\lceil \frac{|S_0|}{W} \rceil$), CBH produces a slightly higher network lifetime, while at the same time minimizing the total coverage breach. As the terrain becomes larger the number of covers decreases. In the contrary, the coverage breach increases, because a sensor can only cover a small number of targets in deployments with a large terrain size.

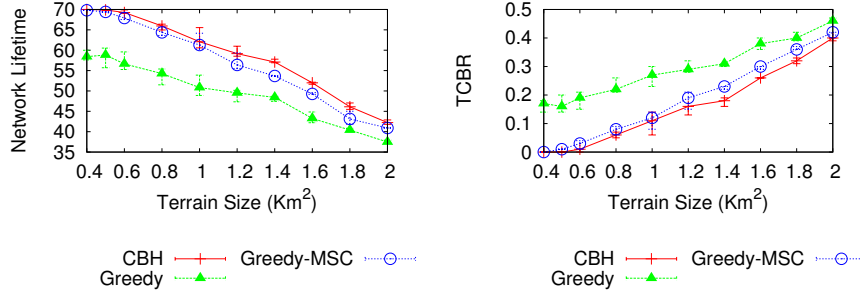


Figure 1: A comparison of 1-hop algorithms (variable terrain size, $W = 5$, $max_sets = \lceil \frac{|S_0|}{W} \rceil$)

In the next experiment, we vary the size of the bandwidth keeping a fixed terrain size of $1150m^2$. As it is expected, the breach rate decreases as the bandwidth increases (Figure 2 right). The network lifetime also decreases since the increased size of the denominator W decreases the maximum number of cover sets (*i.e.* $max_sets = \lceil \frac{|S_0|}{W} \rceil$).

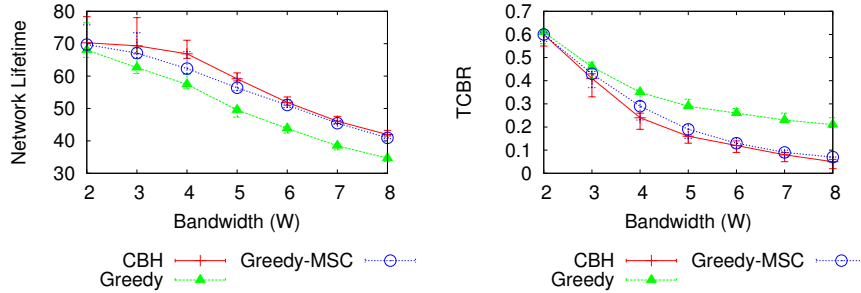


Figure 2: A comparison of 1-hop algorithms (terrain size $1150m^2$, variable bandwidth size, $max_sets = \lceil \frac{|S_0|}{W} \rceil$)

In the second instance, the simulated algorithms must not exceed a total breach rate equal to 0.3, while we measure the generated number of cover sets and the lifetime of the network. In this experiment, we test the impact of a relatively low breach rate and a variable terrain size on the output of the algorithms. As we can see from Figure 3, Greedy-MSC produces a slightly higher number of cover sets (for most cases), but CBH exhibits the same or

higher lifetime as it is capable of covering more targets in each cover set.

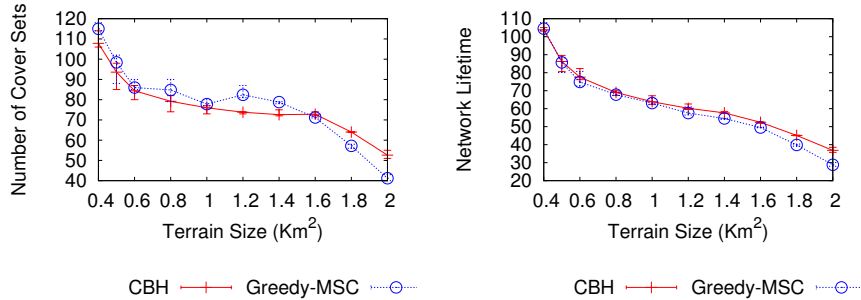


Figure 3: A comparison of 1-hop algorithms (variable terrain size, $W = 5$, $max_tcsr = 0.3$)

In the next experiment, we test the impact of the bandwidth on the algorithms' output, while the total breach rate (max_tcsr) and the terrain size are fixed and equal to 0.3 and $1150m^2$, respectively. The results (see Figure 4), show that the very low bandwidth size and low breach rate do not allow the algorithms to produce any cover sets. Moreover, in cases where the bandwidth is larger than 6, the output remains constant, because the total breach rate is kept much lower than 0.3.

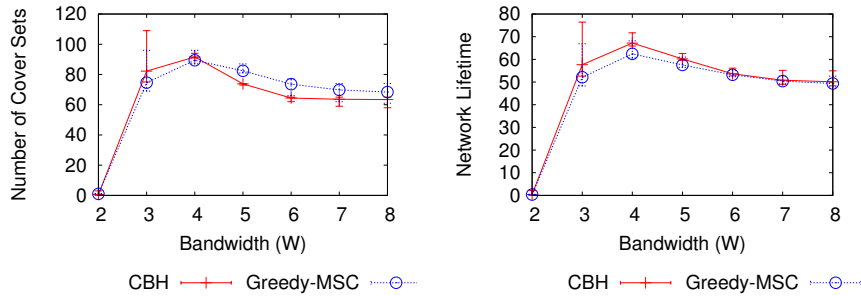


Figure 4: A comparison of 1-hop algorithms (terrain size $1150m^2$, variable bandwidth size, $max_tcsr = 0.3$)

5.2 Considering the communication cost

Figure 5 illustrates the performance of CBH in comparison to the “CBH 1-hop”, an algorithm that does not take into account the connectivity requirement and the additional energy cost. It is obvious that the connectivity constraint limits the maximum lifetime for multi-hop sensor deployments. In sparse sensor deployments CBH can only produce the 50% of the lifetime produced by the

“CBH 1-hop” algorithm. We must point out though that the breach rate can be improved, despite the fact that the results fluctuate from scenario to scenario.

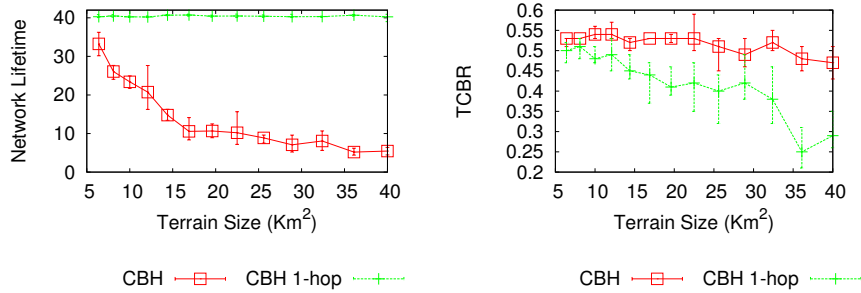


Figure 5: CBH 1-hop vs. CBH (variable terrain size, $W = 20$, $max_nl = 40 \cdot \tau$)

In the next experiment, we assess the performance of the algorithms using a scenario with a fixed terrain size ($10Km^2$) and a variable bandwidth size. Our goal is to produce a network lifetime equal to 40τ . The results presented in Figure 6 show that CBH is not able to achieve the above goal, because the consumed energy increases exponentially with the distance, resulting to a quick battery exhaustion of the nodes that carry out a high rate of information. Additionally, more targets can be covered as the bandwidth grows up, increasing even more the producing data rate and the consumed energy. On the other hand, the increased number of covered targets can lead to a lower breach rate (Figure 6 right).

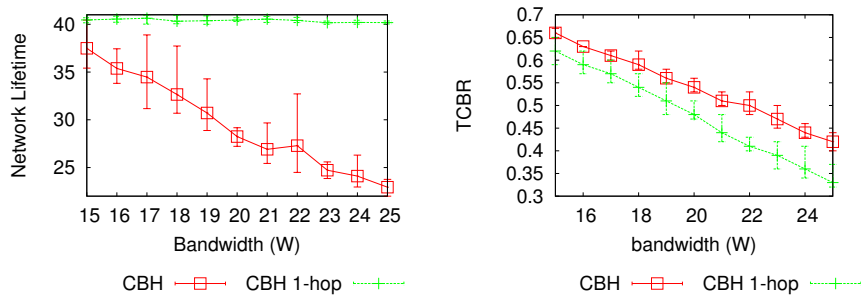


Figure 6: CBH 1-hop vs. CBH (terrain size $10Km^2$, variable bandwidth size, $max_nl = 40 \cdot \tau$)

Concerning the MCBR instance, we distinguish two experiments. In the first one, we vary the terrain size, while we keep a constant bandwidth size, and in the second one, we test the impact of the bandwidth on the lifetime, assuming a fixed terrain and a variable bandwidth size. We can conclude from Figure 7 that despite the fact that the coverage breach rate cannot exceed a certain value, CBH produces lower network lifetime than the 1-hop algorithm.

Furthermore, when W is low, the algorithms cannot produce any result, since the breach rate is higher than max_tcsr from the first cover set. Despite the fact that CBH tries to use all the available energy of all the possible paths to the BS, the 1-hop algorithm is still ahead, since the sensor nodes communicate directly with the BS.

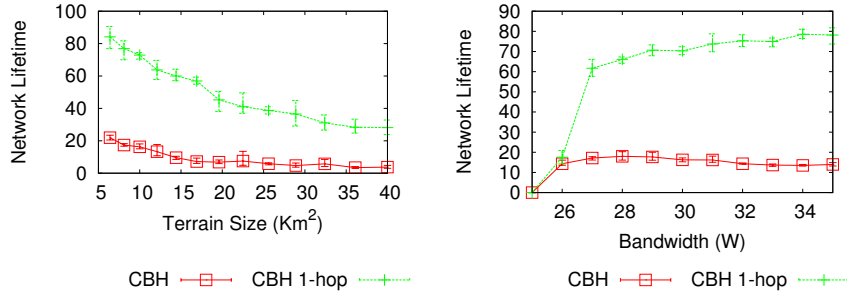


Figure 7: CBH 1-hop vs. CBH (left: variable terrain size, $W = 30$, $max_tcsr = 0.4$ – right: $10Km^2$, variable bandwidth size, $max_tcsr = 0.4$)

6 Conclusions

In this paper we dealt with the connected coverage breach problem in sensor networks. We proposed an algorithm that takes into account the critical requirement of connectivity and a practical energy consumption model. The algorithm tries to distribute the traffic to different paths by using a weighted connected graph. We showed through simulation that our algorithm can maximize the network lifetime and minimize the total coverage breach rate, outperforming the existing approaches for 1-hop networks. Moreover, in multi-hop networks, we have shown that the connectivity constraint and the increased data rate limit the lifetime of the network due to the energy exhaustion of the nodes that must transmit the data to the BS. Finally, as our algorithm depends on the neighboring information (e.g. weighted graph construction), it can be easily transformed to a distributed algorithm, making it more scalable on networks with high number of nodes.

References

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, 2002.
- [2] M. Cardei and D-Z. Du. Improving wireless sensor network lifetime through power aware organization. *ACM Wireless Networks*, 11(3):333–340, 2005.

- [3] M. Cardei, M. Thai, Y. Li, and W. Wu. Energy-efficient target coverage in wireless sensor. In *Proc. of INFOCOM 05*, volume 3, pages 1976–1984. IEEE, March 2005.
- [4] Maggie Cheng, Lu Ruan, and Weili Wu. Coverage breach problems in bandwidth-constrained sensor networks. *ACM Trans. Sen. Netw.*, 3(2):12, 2007.
- [5] Maggie X. Cheng, Lu Ruan, and Weili Wu. Achieving minimum coverage breach under bandwidth constraints in wireless sensor networks. In *Proc. of INFOCOM 05*, pages 2638–2645, March 2005.
- [6] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson. Wireless sensor networks for habitat monitoring. In *Proc. of International Workshop on Wireless Sensor Networks and Applications*, pages 88–97. ACM, September 2002.
- [7] S. Slijepcevic and M. Potkonjak. Power efficient organization of wireless sensor networks. In *Proc. of International Conference on Communications (ICC'01)*, pages 472–476. IEEE, June 2001.
- [8] D. Tian and N. D. Georganas. A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Proc. of 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pages 32–41. ACM, September 2002.
- [9] Manish Bhardwaj Timothy, Timothy Garnett, and Anantha P. Ch. Upper bounds on the lifetime of sensor networks. In *Proc. of IEEE International Conference on Communications*, pages 785–790, 2001.
- [10] Chen Wang, M.T. Thai, Yingshu Li, Feng Wang, and Weili Wu. Minimum coverage breach and maximum network lifetime in wireless sensor networks. In *Proc. of GLOBECOM 07*, pages 1118–1123, 2007.
- [11] Qun Zhao and Mohan Gurusamy. Lifetime maximization for connected target coverage in wireless sensor networks. *IEEE/ACM Trans. Netw.*, 16(6):1378–1391, 2008.
- [12] D. Zorbas, D. Glynos, P. Kotzanikolaou, and C. Douligeris. B{GOP}: An adaptive coverage algorithm for wireless sensor networks. In *Proc. of the 13th European Wireless Conference*, April 2007.