Target Monitoring in Wireless Sensor Networks: A Localized Approach

Kamrul Islam and Selim G. Akl

{islam,akl}@cs.queensu.ca

School of Computing, Queen's University

Kingston, Ontario, Canada K7L 3N6

Abstract

We consider the following target monitoring problem: Given a set of stationary targets $T = \{t_1, \cdots, t_m\}$ and a set $V = \{v_1, \cdots, v_n\}$ of sensors, the target monitoring problem asks for generating a family of subsets of sensors $V_1, \cdots, V_s$ called *monitoring sets*, such that each V_i monitors all targets. In doing so, the objective of this problem is to maximize $z = s/k$, where $k = \max_{v_j \in V} |\{i : v_j \in V_i\}|$. Maximizing z means generating a large number of monitoring sets and minimizing the number of frequencies of a sensor (or node) in these sets. As energy is one of the main issues of wireless sensor networks and sensing consumes energy, maximizing z has a direct impact in prolonging the lifetime of sensor networks. This is because by maximizing z, we distribute the monitoring responsibilities to all sensors as equally as possible and at the same time, reduce the participations of individual nodes in these monitoring sets. Towards achieving this goal, we present a *localized* algorithm. The algorithm is simple and requires each node to know only its 2-hop neighborhood. Furthermore, nodes do not need to know their geographic positions. In short, the main idea is that each node maintains a record (counter, ID), where counter is the number of previous monitoring sets in which the node has been selected to monitor the targets. To select sensors for the next monitoring set, for each target, the lexicographically smallest sensor covering it, is selected. We provide special instances where the algorithm obtains an optimum result. As a by-product of the algorithm, it is shown that the size of a monitoring set is at most a constant times the size of a minimum monitoring set when the number of targets is a constant. We present extensive simulation results to evaluate the performance of the algorithm in randomly generated graphs.

1 Introduction

Sensors deployed in the plane are expected to provide us with relevant information depending upon the applications they are used for. Every sensor is equipped with a sensing and transmission device. A sensor can sense targets or other sensors if they are within its sensing range and similarly it can transmit to and receive from a sensor if the two sensors are within their respective transmission ranges. After deployment, sensors form an ad hoc wireless network among themselves by establishing virtual links if two sensors are within each other's transmission range. It is assumed that the network is connected. Throughout the paper the sensing range of a sensor is equal to its transmission range unless otherwise specified.

Consider Figure 1 where a network of six sensors is shown. Sensors u, w are within their transmission range and can sense a target (shown as a rectangle) as it falls in their sensing range. Sensor v can also sense the target but none of u or w is v's neighbor. After the ad hoc network is formed, sensors begin sensing the environment and depending on the application, such as target tracking, target monitoring, disaster-relief alarm, area coverage, and so on, they cooperate by sending and receiving sensed information among them. The goal of such applications is achieved when relevant and timely information generated from the collaboration of sensors is relayed to a central base station to take appropriate action.

As wireless sensors have limited battery power, one critical aspect of such networks is energy consumption which is caused by sensing and transmission activities by individual sensors. The more a sensor engages in sensing and transmitting, the more energy it consumes. Although in most applications a sensor's energy cannot be replenished, it is generally expected that the sensor will continue to provide information for a long period of time. This raises the
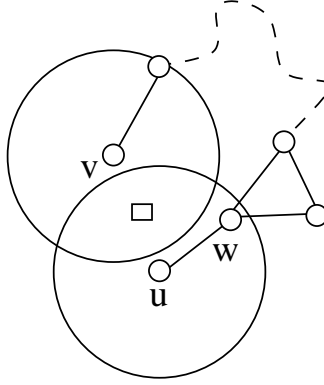
Fig. 1: A sensor network in which three sensors can sense a target which falls in their sensing range.

need to judicially utilize the sensors to obtain required information from the sensing field, since otherwise sensors will quickly drain their energy and be unable to function any longer. Therefore, an algorithm for a wireless sensor network should be energy-efficient in the sense that it ensures the careful use of sensors to reduce energy consumption for sensing and transmission, and at the same time meets the expected standard of quality in the solutions provided by them. If a sensor is engaged in sensing (and /or transmitting) call it *active* in which case it consumes energy, otherwise the sensor is in the *sleep* mode in which case it does not loose energy. We use 'sensor' and 'node' interchangeably in what follows.

In this paper we study the target monitoring problem. Given a set of stationary targets $T = \{t_1, \cdots, t_m\}$ and a set $V = \{v_1, \cdots, v_n\}$ of sensors, the target monitoring problem asks for generating a large family of subsets of sensors $V_1, \cdots, V_s$ called *monitoring sets*, such that each V_i monitors all targets. The idea is that only one such set is active for any certain period of time and after that time period another set becomes active and so on, thus providing continuous monitoring. However, the objective of this problem is to maximize $z = s/k$, where $k = \max_{u \in V} |\{i : u \in V_i\}|$. It is assumed that a large number of sensors are deployed in the plane which are in close proximity of the targets, such that for every target there is at least a sensor to monitor it. Considering each i ($1 \leq i \leq s$) as a round, we activate the sensors in V_i in round i to monitor all targets, while all other sensors $V \setminus V_i$ are put in the energy-efficient sleep mode.

Maximizing z, that is, generating a large number of monitoring sets (in other words maximizing s) and at the same time minimizing the number of frequencies of a sensor in these sets (i.e., minimizing k) has a direct impact in prolonging the *lifetime* of sensor networks. By the lifetime of a sensor network we mean the elapsed time between the deployment of the sensors and the time when the first sensor runs out of energy. By maximizing z, we distribute the monitoring responsibilities among the sensors in a more balanced way. That is, by having a large number of monitoring sets we increase the chance of selecting all sensors in the network in these sets, and by minimizing the number of participations of individual nodes in these sets, we reduce the possibility of selecting the same nodes. Therefore, as nodes in the network are put in the active and the energy-efficient sleep mode alternatively, the lifetime of individual nodes is prolonged and hence the lifetime of the network.

Another important aspect of sensor networks is the issue of scalability. Since hundreds or even thousands of sensors may sometimes be deployed to accomplish certain tasks, it is generally expected that the algorithms for such networks be distributed (or more preferably *localized*) rather than centralized. This is because in a centralized system all sensors have to relay their information (such as id, neighborhood information, etc.) to the central base station which then executes the algorithm and returns the results to the sensors. As this maybe infeasible for large networks, an algorithm of a distributed nature is more acceptable in such cases, which relieves the nodes from sending their information to the central base station. In a distributed system the sensors run the same algorithm individually based on local knowledge of the network. This makes distributed algorithms suitable for large networks. However, a preferred approach is to use a localized system in which each sensor is allowed to know only a *constant* neighborhood of information to execute its algorithm and make decisions. Define an algorithm as p-localized if each node u is allowed to exchange messages with its neighbors v at most p−hops away and take decisions accordingly based on this information.

2

In order to solve the target monitoring problem we propose a 2-localized algorithm and therefore, a sensor is restricted to know only the 2-hop neighborhood. Another salient feature of the algorithm is that the sensors do not need to know their geographic positions. They only need to know the ids and the connectivity information of their 2-hop neighborhood.

The rest of the paper is organized as follows. In Section 2 we describe related work. In Section 3 we provide definitions and assumptions that are used throughout the paper. The localized algorithm is presented in Section 4 and a theoretical analysis of the algorithm follows in Section 5. Experimental results are presented in Section 6. We conclude in Section 7.

2   Related Work

Coverage (also called *monitoring*) has been one of the important topics in sensor networks and has received a lot of attention during the past several years [2, 4, 5, 6, 9, 8, 11, 13, 14]. The main goal of most of the is to devise scheduling algorithms such that individual sensors in the network are assigned rounds which indicate to them during which rounds they will be active and during which rounds they will be in the sleep mode. When a set of sensors monitors a certain area or a target, it is generally possible to monitor the area or the target by a *small* subset of them. So, it is redundant to make all the sensors active at the same time for the monitoring instead of using the small subset. This observation leads researchers to devise efficient algorithms such that at any time only a small number of sensors are set as active to monitor the area or the targets.

In [2] the authors consider the target coverage problem with the goal to minimize energy consumption. Given a set of targets, they propose an energy-efficient centralized scheme to provide coverage to all the targets by disjoint sets of sensors. They divide the sensor set into disjoint subsets and activate each set to perform the covering task. Basically, their main idea is to generate as many disjoint sets of sensors as possible such that each set covers all the targets. However, the algorithm is centralized and does not scale for larger networks. In [12] the authors address the area coverage problem where their algorithm works in two phases. In the first phase the algorithm determines how many different sensors cover the different parts of the monitored area. Like the algorithm in [2], the second phase allocates sensors into mutually independent sets where each set is active at any time to provide the area coverage. The algorithm is also centralized and the authors do not provide any theoretical results regarding the performance of their algorithm. Simulation results show that the algorithm achieves significant power savings (by generating more disjoint sets of sensors) in performing area coverage.

The work relevant to ours is [3] where the authors consider the target coverage problem by a set of subsets of sensors while the subsets do not need to be disjoint. They call this the maximum set covers (MSC) problem and present two centralized algorithms, one based on linear programming and the other is a greedy algorithm. They do not provide any theoretical analysis of their algorithms and give simulation results to verify their approaches. In this paper, we consider a target monitoring problem where we focus on generating a large number of monitoring sets and at the same time reducing the number of participations of each node in these generated set.

A recent result related to our problem is described in [1], where the authors consider the *monitoring schedule* problem: Given a set of sensors and a set of targets it is required to find a partition of the sensor set such that each part can monitor all targets. Each part of the partition is used for one unit of time and the goal is to maximize the number of parts in the partition. They present a randomized distributed algorithm which generates at least $(1 - \varepsilon) * opt$ $(0 < \varepsilon < 1)$ parts, with high probability, where opt is the maximum number of parts in the partition. However, they make the assumption that the sensors must know their geographic positions.

The authors also show that by modifying their algorithm they can find a constant approximation factor for the problem and the sensors do not need to know their geographic positions. Our work is related to theirs in the sense that we maximize the number of parts while trying to reduce the use of the same nodes in these parts. Besides, ours is a deterministic algorithm as opposed to their randomized one and the assumption that the sensors know their positions is excluded.

3 Model and Definitions

We assume that sensors are deployed in the plane and model the underlying sensor network by an undirected graph $G = (V, E)$, where the vertex set V denotes the set of sensors and E represents the links $(u, v) \in E$ between two nodes $u, v \in V$ if they are within their transmission range, TR. A sensor is equipped with a sensing device that can monitor a target if it falls inside its sensing range, SR. There are a number of reasonable assumptions relating TR and SR and we study two popular ones [1, 7]. In dealing with the target monitoring problem, we consider two situations (i) $TR = SR$ and (ii) $TR = 2 * SR$.

Define the neighbor sets $N(u)$ and $N[u]$ of node u as $N(u) = \{v | (u, v) \in E, u \neq v\}$ and $N[u] = N(u) \cup \{u\}$. By $N_f(u)$ we mean the set of nodes from u which are at most f hops away. For simplicity we use $N(u) = N_1(u)$. The degree $deg(u)$ of a node is the number of neighbors it has, i.e., $deg(u) = |N(u)|$. Each node u is identified by a unique index denoted as $id(u)$. When a shortest path between two nodes $u, v \in V$ is referred we mean the shortest hop distance between them and denote it by $d(u, v)$.

The target set is denoted as $T = \{t_1, \cdots, t_m\}$, where t_i is a target. The set of targets monitored by node u is referred to as $T(u)$. For u and $t \in T(u)$, $T_t(u)$ represents the set of sensors in $N_2[u]$ that monitors the target t (i.e., $T_t(u) = \{t | t \in T(u) \cap T(v), v \in N_2[u]\}$).

Each node u maintains an ordered pair at each round i (initially $i = 1$), $p_i(u) = \prec (ct_i(u), id(u)) \succ$, where the first element $ct_i(u)$ (also called the *counter*) denotes the number of monitoring sets in which u has already participated. Initially, $ct_1(u) = 0, \forall u \in V$, and this is incremented by one ($ct_i(u) = ct_i(u) + 1$) each time u participates in some monitoring set. For a subset X, and two nodes $u, v \in X$, we say that node u is lexicographically smaller than node v at round i if $p_i(u) < p_i(v)$, i.e., either $ct_i(u) < ct_i(v)$ or $ct_i(u) = ct_i(v)$ and $id(u) < id(v)$. The rank of node u with respect to X, denoted by $r(X(u))$, is the index of u in the lexicographically sorted (ascending order) nodes of X.

3.1 Problem Formulation

We formulate the target monitoring problem in the following way. Given a set of m targets $T = \{t_1, \cdots, t_m\}$ and a set of n sensors $V = \{v_1, \cdots, v_n\}$ (both) randomly deployed in the plane such that for each target in T there is at least one sensor that monitors it, we would like to find a family $\mathcal{V}$ of subsets $V_1, \cdots, V_s$ such that

i) $\forall i$, V_i monitors all targets in T,
ii) $z = s/k$ is maximized, where $k = \max_{v_j \in V} |\{i : v_j \in V_i\}|$.

4 The Algorithm

In this section we present a 2-localized algorithm for the target monitoring problem and discuss some problems related to the myopic view of localization. We assume that the transmission range TR is equal to the sensing range SR. As individual nodes execute the same algorithm, we describe what happens to an arbitrary node u in the network.

Our algorithm works in rounds starting from round 1. At the first round ($i = 1$), node u forms its monitored target list $T(u)$ by sensing (or monitoring) the targets in its sensing range. Then u sends two items to all its neighbors which are at most 2-hops away. It sends its list $T(u)$ and an ordered pair $p_i(u) = \prec (ct_i(u), id(u)) \succ$ to $v \in N_2(u)$. Note that initially $ct_1(u) = 0$. Similarly u receives such information ($T(v)$ and $p_i(v)$) from all its neighbors $v \in N_2(u)$. Thus u knows the targets that are monitored by its one and two hop neighbors.

After obtaining $T(v)$ and $p_i(v)$ from all its one and two hop neighbors, u forms the set $T_a(u)$ for each of its monitored target $a \in T(u)$ (recall that $T_a(u) = \{a | a \in T(u) \cap T(v), v \in N_2[u]\}$). In other words, the set $T_a(u)$ refers to the sensors monitoring target a that are at most 2-hops away from u. Then for each $T_a(u)$, u computes its rank $r(T_a(u))$ in this round. If u is the smallest ranked node in any $T_a(u)$, then it becomes active to monitor a, otherwise it goes into the sleep mode. According to the notation, all active nodes in round i are represented by V_i which monitor all the targets. For example, if sensors u_1, u_2, u_3 (assume the subscript denotes the respective id) monitor target b (see Figure 2) then $|T_b(u_1)| = 3$. At round 1, u_1 becomes active by being the smallest ranked node for monitoring b while the other two nodes can go into the sleep mode, then at rounds 2 and 3, u_2 and u_3 take their turn to be active

by becoming the smallest ranked node, respectively. Thus the monitoring sets in round 1,2, and 3 are $V_1 = \{u_1\}$, $V_2 = \{u_2\}$ and $V_3 = \{u_3\}$, respectively.

If u becomes active in round i then its counter is incremented by one ($ct_{i+1}(u) = ct_i(u) + 1$), otherwise the counter value remains the same. Node u then sends its $p_{i+1}(u) = \prec (ct_{i+1}(u), id(u)) \succ$ to, and receives $p_{i+1}(v)$ from $v \in N_2(u)$ and a new round $i + 1$ begins. As before u computes its rank (since $p_{i+1}(.)$ values are updated) for each target $a \in T(u)$. The smallest ranked node for a target a becomes active and starts monitoring a. All smallest ranked nodes in the network form the monitoring set V_{i+1} which monitors all targets in T and so on. The algorithm is given in Figure 3.

4.1 A Problem with Locality

In general, it is assumed that sensors which are in close proximity in the plane sense or observe the same data due to the spatial correlation. For a subset X_t of nodes that monitor the same target t in some round, it is supposed that the nodes will be in close proximity. However, the nodes in X_t, although they monitor the same target, can have arbitrarily long hop distance between each other, while the Euclidean distance maybe slightly more than their transmission range. We call this situation the *Locality Effect*, where the monitoring nodes for a certain target do not know about each other about their monitoring.

In Figure 1, the Euclidean distance between u and v is $TR + \varepsilon$ (recall that $0 < \varepsilon < 1$) and they both monitor the same target (shown as a rectangle) whereas their shortest path $d(u, v)$ (in terms of hop distances) can be arbitrarily long.
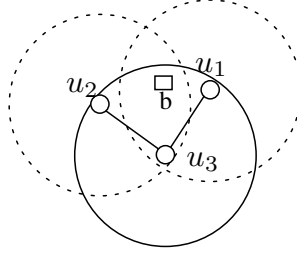

Fig. 2: Nodes u_1, u_2, and u_3 monitor target b and the monitoring sets are $V_i = \{u_i\}, \forall i$

5 Theoretical Analysis

In this section we give an overview of the theoretical analysis of the algorithm. Consider a target $t \in T$ and let X_t be the set of sensors that monitors it in some round. For any node $u \in X_t$, u knows only whether other nodes, which are within its 2-hop neighborhood (i.e., $N_2[u]$), can monitor t and therefore chooses exactly one among $N_2([u])$ for the monitoring. However, due to the locality effect there can be two nodes $u, v \in X_t$ such that $v \notin N_2(u)$ and both u and v monitor t and u does not have any clue about v. We would like to determine an upper bound on how many sensors can monitor a target while none of them is aware of the other.

First, we have the following simple observation from our algorithm.

Observation 5.1 *The set V_i of all active nodes monitors all targets in T at round i.*

As one of the main results, we show that at any round at most five sensors can simultaneously monitor a target, none of which is within the 2-hop neighborhood of the other. In the following, we assume that $TR = SR$.

Lemma 5.2 *For any target $t \in T$, at most five sensors are set as active.*

```
Input: A connected graph G = (V, E) and a set of targets
T such that each target is monitored by at least one sensor.

Output: A family of subsets of sensors V₁, ⋯ , Vₛ such
that each Vᵢ monitors all targets in T.

1: i = 1
2: Broadcast pᵢ(u) and T(u) to v ∈ N₂(u)
3: Receive pᵢ(v) and T(v) from v ∈ N₂(u)
4: Compute Tₐ(u) = {a|a ∈ T(u) ∩ T(v), v ∈ N₂(u)},
∀a ∈ T(u)
5: /*Tₐ(u) represents the set of sensors that monitor a*/
6: For each round i
7:    Compute rank r(Tₐ(u)), ∀a ∈ T(u)
8:    If ∃a ∈ T(u) such that
9:      r(Tₐ(u)) < r(Tₐ(v)), v ∈ N₂(u) Then
10: /*Selects the smallest ranked node for each target*/
11:      u becomes active
12:   Endif
13:   i = i + 1
14: If u is active Then
16:    ctᵢ(u) = ctᵢ₋₁(u) + 1
17: Endif
18: If ctᵢ(u) ≠ ctᵢ₋₁(u) Then
19:    Send pᵢ(u) to v ∈ N₂(u)
20: Endif
21:    Receive pᵢ(v) from v ∈ N₂(u)
22: Endfor
```

Fig. 3: A 2-Localized algorithm for the target monitoring problem

Proof Assume X_t to be the set of sensors that monitors a target $t \in T$ in some round. Suppose for the sake of contradiction that $|X_t| > 5$ and no two nodes $u, v \in X_t$ are within the 2-hop neighborhood of the other. As t is monitored by all sensors in X_t, the Euclidean distance between t and any node $w \in X_t$ must be at most SR. Consider a disk D centered at t with radius equal to SR. Therefore, all the nodes in X_t must be within the disk D. Since $|X_t| > 5$ and nodes of X_t are in D, there will be at least two nodes $u', v' \in X_t$ whose Euclidean distance must be smaller than TR (we can consider TR the radius of D instead of SR, since $SR = TR$). Then u' and v' are direct neighbors to each other, a contradiction. ∎

A tight example of the above lemma is shown in Figure 4, where five sensors are put on the vertices of a regular pentagon with side $TR + \varepsilon$. They can all monitor any target t in the striped region S which is the intersection of their respective sensing disks. Note that for any u_i, $u_i \notin N_2[u_j]$ ($i \neq j$ and $i, j \in \{1, 2, 3, 4, 5\}$).

Then we derive the corollary from the above lemma.

Corollary 5.3 *If V_i^* denotes the minimal set of sensors monitoring targets at round i, then $|V_i| \leq 5m|V_i^*|$, where $m = |T|$ is the number of targets given.* ∎

An implication of the above corollary is that if we have a constant number of targets ($|T| = m \leq c$, c is a constant) to be monitored then, at any round, the number of sensors becomes active by the algorithm is within a constant factor of the optimal, i.e., $|V_i| \leq 5c|V_i^*|$.

Now we show how the algorithm performs towards maximizing the value of $z = s/k$. First we need some notation. Let $X_1, X_2, \cdots, X_m$ be the subsets of sensors ($X_i \subseteq V$, $1 \leq i \leq m$) that monitor the targets $t_1, t_2, \cdots, t_m$, respectively, in some round. Let X_p be the minimum cardinality subset among all X_i's (ties are broken arbitrarily).
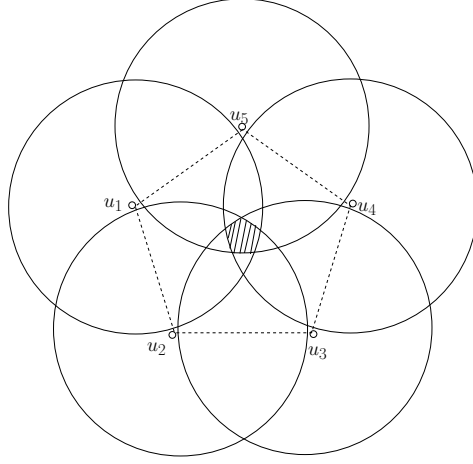
Fig. 4: For a target in the striped region (intersection of the five disks) we set as active at most five sensors whereas only one sensor suffices.

Now we divide each X_i (X_i monitors target i) in the following way. Due to the locality effect we can have at most five subsets $X_i(i_1), X_i(i_2), X_i(i_3), X_i(i_4)$ and $X_i(i_5)$ of X_i (i.e., $X_i(i_a) \subseteq X_i$, $1 \le a \le 5$, $1 \le i \le m$) such that no node in one subset knows whether any nodes in other subsets are monitoring i. In other words, there is no node $u \in X_i(i_a)$ such that $v \in X_i(i_b)$, $a \ne b$ and $v \in N_2[u]$. Let $X^* = \min_{1 \le i \le m, 1 \le a \le 5} |X_i(i_a)|$. Note that $|X^*| \ge 1$.

Denote by z_{opt} the optimal value of z and $z_{opt} > 0$, that is, $z_{opt} \ge z = s/k$ for all possible values of s and k. If z_{alg} denotes the value of z obtained by our algorithm then we have the following lemma:

Lemma 5.4 $z_{opt} = |X_p|$ and $z_{alg} \ge 1$. Hence z_{opt} is at most $|X_p|$ times the value of z_{alg}.

Proof First we derive the value of z_{opt}. In order to compute the value z_{opt}, notice that there can be $|X_p|$ monitoring sets $V_1, V_2, \cdots, V_{|X_p|}$ such that each node in X_p can participate in these monitoring sets exactly once. It is easy to see that for the optimal algorithm to generate the $|X_p| + 1$-st monitoring set, $V_{|X_p|+1}$, there will be at least one node in X_p such that the number of participations of the node in $V_1, V_2, \cdots, V_{|X_p|}, V_{|X_p|+1}$ is at least two (because of the Pigeonhole Principle). Thus $z_{opt} = s/k = |X_p|$, where $s = |X_p|$ (i.e., $|X_p|$ monitoring sets $V_1, V_2, \cdots, V_{|X_p|}$) and $k = 1$.

We now analyze the behavior of the algorithm to find z_{alg}. With each node participating exactly once, the algorithm can generate X^* monitoring sets $V_1, V_2, \cdots, V_{X^*}$, since only one node is selected in each iteration from X^*. Hence $z_{alg} \ge |X^*| \ge 1$. Hence z_{opt} is at most $|X_p|$ times the value of z_{alg}. ■

5.1 $TR = 2 * SR$

In this section we assume that the transmission range of a sensor is twice its sensing range, $TR = 2 * SR$. In the previous subsection it is shown that we can achieve a good approximation factor (a constant factor) for the size of each of the monitoring sets. However, when the number of targets is a constant, the approximation factor is much worse towards minimizing the value of k. This is because of the very limited information of the 2-hop neighborhood available to individual sensors and we assumed that the transmission range is equal to the sensing range of a sensor. Here we investigate whether the value of k can be improved by assuming $TR = 2 * SR$. This is a very reasonable assumption in the context of coverage (monitoring) problems in sensor networks [7]. However, we use this assumption and obtain some useful results for the target monitoring problem.

Observation 5.5 Let node $u \in V$ monitor $T(u) \subseteq T$ targets. If there is a node $v \in V$, $v \ne u$ such that $T(u) \cap T(v) \ne \phi$, then $v \in N(u)$.

Proof Suppose for contradiction, $v \notin N(u)$. Since $T(u) \cap T(v) \neq \phi$, that means v can monitor a target which u can monitor. In this case the Euclidean distance between u and v can be at most twice their sensing range (see Figure 5, where dotted and solid circles represent the transmission and the sensing ranges, respectively) contradicting the fact that v is not a direct neighbor of u. ∎
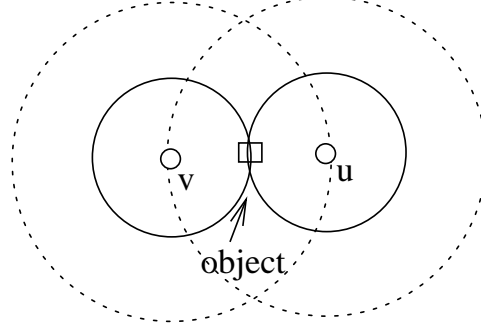

Fig. 5: If $T(u) \cap T(v) \neq \phi$ then $v \in N(u)$.

The above observation implies that if u monitors t then u knows exactly which other nodes are also monitoring t and moreover, all such nodes are u's direct neighbors.

Lemma 5.6 *Under the assumption that $TR = 2 * SR$, for any target $t \in T$, exactly one sensor is set as active.*

Proof The proof is similar to the proof of Lemma 5.2. Consider a target $t \in T$ and let X_t be the subset of sensors that monitors it. There is no locality effect as evident from Observation 5.5. So each node in X_t knows whether all other nodes in X_t are monitoring t. Hence the node with the smallest rank in X_t will be active to monitor t. ∎

We obtain the following directly from the above lemma:

Corollary 5.7 *If V_i^* denotes the minimal set of sensors monitoring targets at round i, then $|V_i| \leq m * |V_i^*|$, where $m = |T|$ is the number of targets given.* ∎

As before, if the number of targets is a constant, the number of active sensors at each round will be within a constant factor of the optimal number of active sensors, $|V_i| \leq c' * |V_i^*|$, where c' is a constant.

In the following, we show that our algorithm obtains optimal results in special instances. Let $X_1, X_2, \cdots, X_m$ be the subsets of sensors that monitor the targets $t_1, t_2, \cdots, t_m$, respectively, where $X_i \subseteq V$ in some round. Let X_p be the minimum cardinality subset among all X_i's (ties are broken arbitrarily). If $X_i \cap X_j = \phi$, $i \neq j$ and $1 \leq i, j \leq m$, that is, a sensor exactly monitors one target, then we have the following lemma.

Lemma 5.8 *With the above assumption and $TR = 2 * SR$, we have $z_{alg} = z_{opt}$*

Proof Let $X_p = \{u_1, u_2, \cdots, u_{|X_p|}\}$ denote the subset of sensors that monitor p (recall that X_p has minimum cardinality among $|X_i|$'s). The optimal algorithm can produce $|X_p|$ monitoring sets $V_1, V_2, \cdots V_{|X_p|}$ with each node in X_p participating exactly once. Since X_p has the smallest cardinality, for the optimal algorithm to produce the next monitoring set $V_{|X_p|+1}$, at least one of the nodes in X_p must be selected in $V_{|X_p|+1}$. So, $z_{opt} = |X_p|/1 \geq (|X_p|+1)/2$.

Now we see how the algorithm behaves towards finding z_{alg}. Since there is no locality effect (all nodes $u_1, u_2, \cdots, u_{|X_p|}$ are aware of each other about monitoring p), the algorithm sets as active exactly one node from X_p (say node u_j is selected in some round). Since $X_p \cap X_i = \phi$, for $i \neq p$ and $1 \leq i \leq X_m$, no other nodes from X_p will be selected (active) in that round. Therefore, at any round exactly one node is set as active from X_p and hence we have $V_1, V_2, \cdots V_{|X_p|}$ monitoring sets.

Now we show that any node $u_j \in X_p$ is exactly used once in $V_1, V_2, \cdots V_{|X_p|}$ monitoring sets.

8

Suppose for contradiction that a node u_j is used in monitoring sets V_r and V_w, where $1 \le r, w \le |X_p|$ and $r < w$. After being selected first in V_r, u_j's $ct(u_j)$ value is increased by one. If u_j is selected (from X_p) again in V_w, that means its rank is smaller than other nodes in X_p. Since $w \le |X_p|$ and exactly one node is selected from X_p, there is at least some node in X_p which has not been selected in previous monitoring sets $V_1, V_2, \cdots, V_{w' < w}$. If this has not been selected in previous monitoring sets its rank will be smaller than u_j's, a contradiction. The claim follows. ∎

6 Simulation

We conducted extensive simulations on random networks to study the performance of our proposed 2-localized algorithm. In this section, we provide and analyze experimental results regarding (i) maximizing $z = s/k$ and (ii) the size of monitoring sets $V_1, V_2, \cdots, V_s$. In the experiments, first we distribute a set of $m \in \{10, 20, 30, 40, 50\}$ targets randomly and uniformly in a square field of size 400m x 400m. Then we generate a random graph G by placing $n \in \{100, 200, 300, 400, 500\}$ nodes uniformly and randomly and test the connectivity of G. If the underlying graph is connected then we apply the algorithm to construct monitoring sets for the targets and analyze its performance in in terms of maximizing the value of z and the size of each V_i.

6.1 Experimental Results

Experiments are done using our own simulator in Java. Setting both the transmission and the sensing range to 60m for all nodes, we apply the algorithm in a 400m x 400m square with $n = 100$, $m = 10$ and first compute the minimum cardinality set X_p. Recall that given $t_1, t_2, \cdots, t_m$ targets, X_p is the minimum cardinality set among $X_1, X_2, \cdots, X_m$ where $X_i \subseteq V$ denotes the set of sensors that monitors the target t_i, $(1 \le i \le m)$. Then we generate the monitoring sets $V_1, V_2, \cdots, V_{|X_p|}$. In the experiments we set the optimal z value, $z_{opt} = s/k = |X_p|/1$ since the optimal algorithm can generate at most $|X_p|$ monitoring sets with $k = 1$. In order to find the value of z_{alg} we figure out the maximum frequency k of a node in X_p in these monitoring sets and obtain $z_{alg} = |X_p|/k$. We then generate 100 random graphs successively with the same number ($m = 10$) of randomly distributed targets and for each graph we compute the size of the minimum cardinality set $|X_p|$, and maximum frequency k of a node in the monitoring sets and hence obtain z_{opt} and z_{alg}. Finally we obtain the average values of z_{opt} and z_{alg}.

Keeping $m = 10$, we increment the value of n by 100 each time and repeat the whole procedure until $n = 500$. In this way we obtain averages of the parameters (z_{opt} and z_{alg}) for each value of n. Thereafter, for each value of $m \in \{20, 30, 40, 50\}$ for each value of $n = \{200, 300, 400, 500\}$, we run the above experiment 100 times and find the averages of the z_{opt} and z_{alg}.

The results are plotted in Figure 6 where the x-axis shows the number of targets in the field and the y-axis represents the average values of z_{opt} and z_{alg}. Optimal values z_{opt} are shown in solid lines and and the values z_{alg} by the algorithm are shown in dashed lines. Interestingly, in these average graphs the experimental results are much better than the theoretical bounds established in Lemma 5.8. Note that the average z_{alg} values are within a small constant (at most 3) of the optimal value average z_{opt} values. As it is obvious from the figures that the results are even better and consistent in dense graphs (we assume, having 500 sensors in a square of 400m x 400m is denser than having 100 or 200 sensors in the same field). This is intuitive because having more sensors in small areas reduces the locality effect and sensors are likely to be aware of other sensors monitoring the same object.

Finally, as a by product of the algorithm we also consider the sizes of the individual monitoring sets. We compute the cardinality of each of the monitoring sets V_i's and compare it with the optimal size of the monitoring set V_i^*. To find the optimal value $|V_i^*|$ we have used the number of targets as the lower bound of $|V_i^*|$, that is, $|V_i^*| \ge m$. We compute the size of each of the monitoring sets V_i generated by the algorithm and we find that in average it is 3.16 times the size of V_i^*.

7 Conclusions

In this paper, we have presented a simple 2-localized algorithm to compute a family of monitoring sets such that each set monitors all the targets given in some field. Generating a number of different monitoring sets with the purpose of
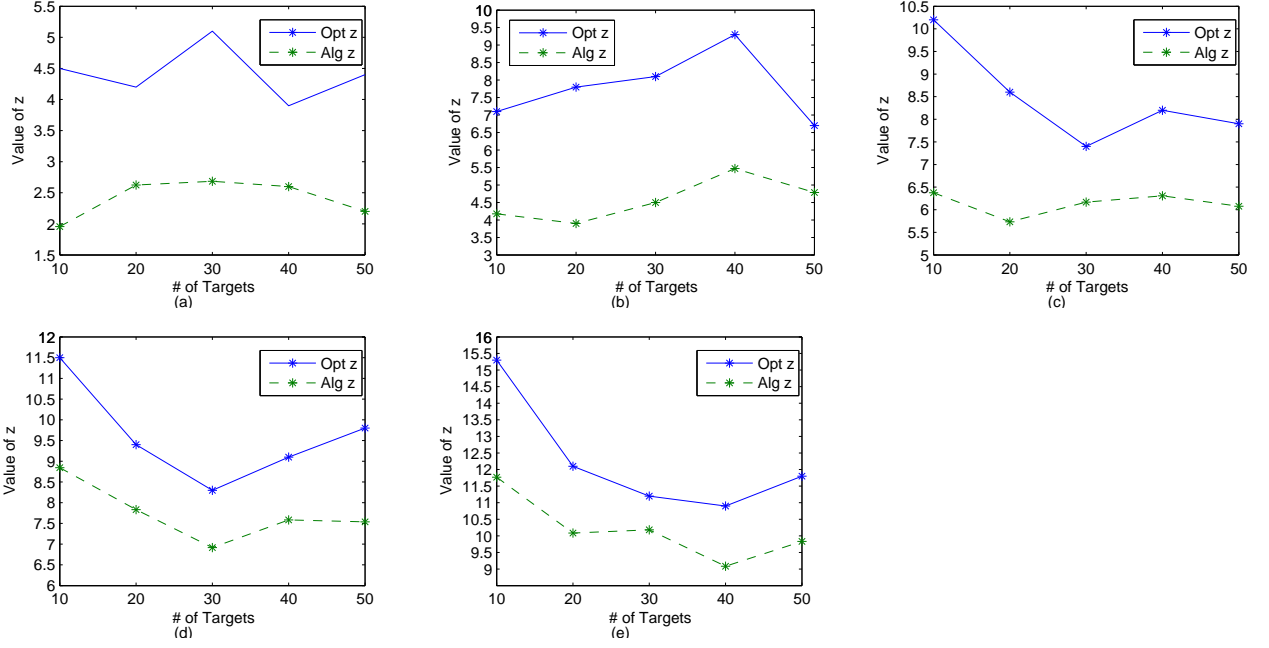
Fig. 6: Average z values (optimal z_{opt} values by solid lines and z_{alg} by dashed lines) for (a) 100 (b) 200 (c) 300 (d) and (e) 500 sensors.

minimizing the number of participations of a node in these sets increases the probability of prolonging the lifetime of sensor networks. Our algorithm is simple and strictly localized, since each node only requires the knowledge of its two hop neighborhood and thus it can be easily applicable to sensor networks. We provide theoretical results about the size of monitoring sets, determine bounds on the value of $z(z = s/k)$ when the transmission range is equal to and twice, the sensing range, respectively. Although the worst-case bound we prove is not appealing (this is because of the fact that the nodes only have very limited information about the topology of the network), we believe it can be improved if nodes have more knowledge about the global network architecture. However, we provide extensive simulation results to assess the performance of our algorithm where the experiments in average random graphs show much better results than the theoretical bound established in the paper.

References

[1] Calines, G., and Ellis, R.: Monitoring Schedules for Randomly Deployed Sensor Networks. In: Proc. Dial-POMC, 2008.

[2] Cardei, M., and Du, D.: Improving Wireless Sensor Network Lifetime through Power Aware Organization. ACM Wireless Networks, Vol. 11, No. 3, pp. 333-340, May 2005.

[3] Cardei, M., Thai, M., Li, Y., Wu, W.: Energy-Efficient Target Coverage in Wireless Sensor Networks. In: Proc. INFOCOM 2005.

[4] Cardei, M., and Wu, J.: Energy-Efficient Coverage Problems in Wireless Ad Hoc Sensor Networks. Computer Communications Journal (Elsevier), Vol. 29, No. 4, pp. 413-420, Feb. 2006.

[5] Carle, J., and Simplot-Ryl, D.: Energy Efficient Area Monitoring by Sensor Networks. IEEE Computer 37, 2 (2004), 40-46.

[6] Gallais, A., and Carle, J.: An Adaptive Localized Algorithm for Multiple Sensor Area Coverage. In Ad Hoc and Sensor Wireless Networks journal (AHSWN), 3(4): 271-288, 2007.

[7] Gallais, A., Carle, J., Simplot-Ryl, D., and Stojmenovic, I.: Localized Sensor Area Coverage with Low Communication Overhead. In: Proc. PerCom 2006.

[8] Heinzelman, W., Chandrakasan, A., and Balakrishnan, H.: Energy-efficient Communication Protocol for Wireless Microsensor Networks. In: Proc. HICSS 2000.

[9] Li, X., Wan,P., and Frieder, O.: Coverage Problems in Wireless Ad-hoc Sensor Networks. IEEE Transactions for Computers. June 2003, page(s): 753- 763, Volume: 52, Issue: 6.

[10] Moscibroda, T., and Wattenhofer, R.: Maximizing the Lifetime of Dominating Sets. In: Proc. IPDPS 2005.

[11] Simplot-Ryl, D., Stojmenovic, I., and Wu, J.: Energy Efficient Backbone Construction, Broadcasting, and Area Coverage in Sensor Networks. In Handbook of Sensor Networks: Algorithms and Architectures (I. Stojmenovic, ed.), Wiley, 2005, 343-379.

[12] Slijepcevic, S., and Potkonjak, M.: Power Efficient Organization of Wireless Sensor Networks. In: Proc. ICC 2001.

[13] Tian, D., and Georganas, N.: A Coverage-preserving Node Scheduling Scheme for Large Wireless Sensor Networks. In: Proc. WWSNA 2002.

[14] Wang, X., Xing, G., Zhang, Y., Lu, C., Pless, R., and Gill, C.: Integrated Coverage and Connectivity Configuration for Energy Conservation in Sensor Networks. In: Proc. CENSS 2003.