A Distributed Greedy Algorithm for Connected Sensor Cover in Dense Sensor Networks

Amitabha Ghosh and Sajal K. Das

Center for Research in Wireless Mobility and Networking (CReWMaN),
Department of Computer Science and Engineering,
The University of Texas at Arlington, Arlington, TX 76019
{aghosh, das}@cse.uta.edu

Abstract. Achieving optimal battery usage and prolonged network lifetime are two of the most fundamental issues in wireless sensor networks. By exploiting node and data redundancy in dense networks, and by scheduling nodes efficiently, minimum battery drainage is possible. In this paper, we focus on the problem of *Minimum Connected Sensor Cover* (MCSC), an NP-hard problem, and describe a distributed greedy algorithm to generate sub-optimal connected sensor covers for homogeneous dense static sensor networks. Our greedy algorithm is based on the notions of maximal independent sets on random geometric graphs, and on the structure of Voronoi diagram. We provide complexity analysis and bounds on the cardinalities of maximal independent sets (MIS) for our problem scenario, and derive an analytical expression for the size of the sub-optimal minimum connected sensor cover. We verify the bounds on the MIS using simulation.

1 Introduction

Wireless sensor networks are distributed, self-organizing, pervasive systems that perform the tasks of sensing and collaborative data processing to provide useful information about some physical phenomenon, which is typically stochastic in nature. Sensor nodes are severely energy constrained due to limited battery power, and are not likely to be replenished during their lifetime. Therefore, it is very important to make use of their energy as optimally as possible and enhance the overall network lifetime. In dense sensor networks, the data sensed by geographically neighboring nodes exhibit a high degree of spatio-temporal correlation, and thus, many such data maybe redundant. Being able to exploit this redundancy is one of the ways to optimize energy usage and extend network lifetime.

Some of the existing works on conserving battery power have focussed on node scheduling algorithms based on the concept of sponsored sectors. Tian and Georganas [1] proposed a node self-scheduling scheme based on sponsorship criteria, by which each node decides whether to turn itself off or on using only local information. Gao, et. al [2] analyzed the problem of estimating redundant

V. Prasanna et al. (Eds.): DCOSS 2005, LNCS 3560, pp. 340–353, 2005.

sensing area and described observations concerning minimum and maximum number of neighbors required for complete redundancy of a particular node.

In literature some methods have been proposed ([3], [4]) to determine the optimal number of nodes and their locations to provide complete coverage of a given sensing region, while maintaining connectivity. Zhang and Hou [5] proved that if the communication radius of a node is at least twice the sensing radius, then complete coverage of a convex region guarantees a connected network. Although, on one hand, the problem of determining the optimal number of nodes for complete coverage is important; deploying redundant nodes in the region on the other hand, contributes to network robustness and can overcome degradation in signal propagation or loss of nodes. However, when there are redundant nodes, keeping them active all the time will lead to faster drainage of energy; and thus, will reduce network lifetime. Now, since the data sensed by geographically neighboring nodes are spatio-temporarily correlated, it is often sufficient to turn on only an optimal number of nodes, which can provide the required data reliability, while putting others to sleep. However, the optimal number of nodes selected should be able to guarantee the same quality of data, which would have been provided if all the nodes were kept active. Since two of the factors which determine data quality are coverage and connectivity [6], the problem boils down to finding an optimal number of nodes that will provide the same quality of coverage and network connectivity.

This leads to the problem of finding the minimum connected sensor cover (MCSC), which is proved to be NP-hard for random deployment of nodes, as the less general problem of covering points using line segments is already known to be NP-hard [7]. Gupta, et. al [8] described an algorithm to construct a connected sensor cover for a network topology with fixed sensing radius, within an $O(logN)$ factor of the optimal, where N is the number of nodes in the network. Zhou, et. al [9] approached the MCSC problem, where each node can vary its sensing and transmission radii and provided a Voronoi diagram based localized algorithm and a greedy algorithm to construct sub-optimal network topologies within a factor of $O(logN)$ of the optimal size. Funke, et. al [10] proposed improved approximation algorithms for connected sensor covers and gave worst case approximation factors of 6π and 12 for grid placement and fine grid algorithms, respectively. They also described a greedy algorithm that provides complete coverage with an approximation factor of $\Omega(logN)$ from the optimal connected sensor cover.

In this paper, we propose a distributed greedy algorithm that constructs a sub-optimal MCSC using only local neighborhood information. Our basic idea is to greedily select a large set of nodes in the first phase, such that none of their sensing circles overlap with each other. Then we make those nodes select an optimal number of neighbor nodes using local information, such that the whole query region gets covered. We use the notions of maximal independent sets (MIS) on *random geometric graphs* and the structure of *Voronoi diagram* to find out sub-optimal MCSC. We also do complexity analysis of our algorithm and provide bounds on the cardinality of MIS for our problem scenario. These bounds are directly related to the size of the sup-optimal MCSC, as will be seen later.

The rest of the paper is organized as follows. In section 2, we formally introduce the problem of MCSC and discuss the notions of *Independent Set* (IS) and *Voronoi diagram*. In section 3, we describe a distributed greedy algorithm to find a sub-optimal MCSC and illustrate it on a grid and random deployment of nodes. In section 4, we show some preliminary simulation results and analyze the algorithm in terms of time complexity and provide bounds on the size of MIS. We conclude the paper in section 5.

2 Preliminaries

In this section, we give a formal description of the problem and introduce some of the basic concepts related to independent sets in graphs and the structure of Voronoi diagrams that we will use in our algorithm to find a sub-optimal connected cover set.

2.1 Problem Formulation

We consider homogeneous static (dense) sensor networks, where all the nodes have the same sensing radius, R_s and the same communication radius, R_c. We assume that the communication radius is α times the sensing radius, i.e., $R_c \geq \alpha R_s$, where $\alpha \geq 2$, and that the sensing range is a circular region of radius R_s with the node at the center. Throughout this paper, we also assume that a node is aware of its own location and its one-hop neighbors' locations. Such localization can be achieved by triangulation methods using received signal strengths or proximity measurements [11].

Let us define the *induced communication graph* on the network as the undirected graph $G_C = (V, E_{R_c})$, where the nodes act as vertices and an edge exists between any two nodes if the Euclidean distance[1] between them is less than the communication radius.

Definition 1. Connected Sensor Cover: *Let a set $S = \{s_1, s_2, ..., s_N\}$ of N nodes, be deployed in a sensing field of area A and let the sensing region covered by node s_i be denoted by A_i. Given a query Q over a region A_Q in the sensing field, where $A_Q \subseteq A$, a set $\Gamma = \{s_{i_1}, s_{i_2}, ..., s_{i_m}\}$ of m nodes is called a connected sensor cover if the following two conditions hold.*

1. $A_Q \subseteq A_{i_1} \cup A_{i_2} \cup ... \cup A_{i_m}$
2. *The induced communication graph G_C is connected, i.e., any pair of nodes in the connected sensor cover can communicate with each other, either directly or indirectly over a multi-hop communication path.*

The *Minimum Connected Sensor Cover* problem is to find the Γ with minimum number of nodes, such that the above two conditions hold. As mentioned

[1] We will denote $d(.)$ as the Euclidean distance function

earlier, MCSC is an NP-hard problem. In this paper, we propose a distributed greedy algorithm to find a sub-optimal MCSC, and derive an analytical expression for the cardinality of this sub-optimal set.

Under our particular problem scenario, where a communication link exists between a pair of nodes only if they are less than a certain distance (R_c) away, the structure of *Random Geometric Graphs* (RGG) provides the closest resemblance for modelling such networks. It is a more realistic model compared to the classical random graph models of Erdos and Renyi. We define the *induced sensing graph* over the set of nodes as a *random geometric graph*, $G_S = (V, E_{R_s})$, where the nodes act as vertices and an edge exists between two nodes s_i and s_j if the Euclidean distance between them is less than twice the sensing radius, i.e., $e(s_i, s_j) \in E_{R_s}$ if $d(s_i, s_j) < 2R_s$. The rationaile for defining the *induced sensing graph* in this way is to make sure that an edge exists between any pair of nodes, only when their sensing circles intersect with each other. We will see in latter sections how we select a large number of nodes, whose sensing circles do not overlap with each other, and hence, cover a maximum area in the sensing field. Next, we define the notion of *independent set* over the *induced sensing graph*.

Definition 2. Maximal and Maximum Independent Sets: *An independent set (IS) of a graph G is a subset of vertices, such that no two vertices in the subset has an edge in G. A maximal independent set (MIS) is an independent set that is not a proper subset of any other independent set, i.e., it is a largest set with respect to set containment. A maximum independent set is an independent set that has the largest possible number of vertices.*

Note that, a maximum independent set is always maximal, but the converse is not always true. In Fig. 1.(a), the set of vertices colored black forms a maximal independent set. In this particular case, this is also a maximum IS. However, the set of white vertices also forms a maximal independent set, but it it not the maximum. It is well known that finding the maximum IS for a general graph is NP-hard [12]. It is also NP-hard to approximate the size of the maximum IS. In the case of sensor networks, we are interested in finding an MIS on the induced sensing graph, which is basically a maximal set of nodes none of whose sensing circles overlap with each other (see Fig. 1.(b)).

2.2 Voronoi Diagrams

In 2-dimension, the Voronoi diagram [13] for a set of discrete points divides the plane into a set of convex polygons according to the nearest neighbor rule: all points inside a polygon are closest to only one point. In other words, if S is the set of points in the 2-D plane, then the convex polygon $V(p_i)$ for any point $p_i \in S$, is defined as the set of all points in the plane that are closest p_i than any other point p_j, for $i \neq j$. Mathematically,

$$V(p_i) = \left\{x \in \Re^2 \mid d(x, p_i) \leq d(x, p_j)\right\}, \;\; p_j \in S, \;\; i \neq j. \tag{1}$$

$V(p_i)$ is called the Voronoi polygon for p_i, and the edges that constitute the polygon are called its Voronoi edges. The Voronoi diagram VD(S) for the set of

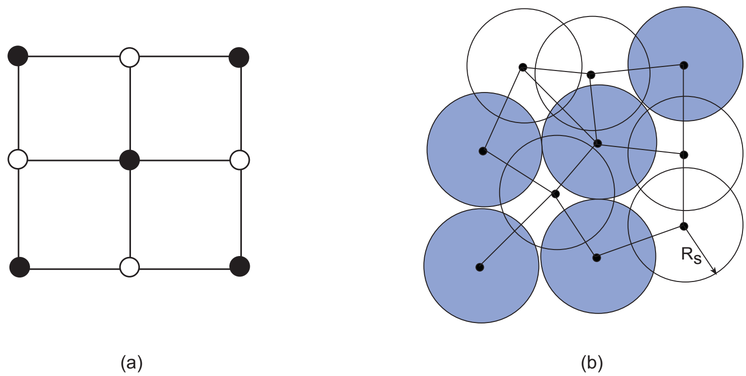


Fig. 1. (a) Set of black vertices forms a maximum IS; set of white vertices forms a MIS but not a maximum. (b) Induced sensing graph of 10 nodes. The set of 5 darkened nodes forms a MIS

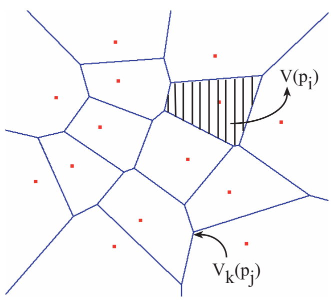


Fig. 2. Voronoi diagram for a set of randomply deployed points in 2-D. $V_k(p_j)$ denotes a Voronoi vertex

points S is the union of such polygons for all the points in S (see Fig. 2). A pair of points p_i and p_j are called Voronoi neighbors if their polygons share a common edge. The number of edges of $V(p_i)$ is equal to the number of neighbors of p_i, and each Voronoi edge is basically the perpendicular bisector between the two points which share the common edge. We will use these properties of Voronoi diagram in developing the proposed greedy algorithm for connected sensor cover.

3 Sub-optimal MCSC Algorithm

In this section, we describe the distributed greedy algorithm that generates a *sub-optimal MCSC* for homogeneous dense static sensor networks. Our algorithm runs in two phases, and always inherently maintains connectivity by selecting the best node at each step, only when it is connected to one or more already selected set of nodes.

Table 1. Notations and their meanings

$N_{s_i}(R_x)$	Set of nodes lying within radius R_x with s_i at the center
$N_{s_i}(R_x - R_y)$	Set of nodes lying in the annular region between the two circular regions of radii R_x and R_y with s_i at the center and $R_x > R_y$
Γ	Set that will contain the sub-optimal MCSC
$V(s_i)$	Voronoi polygon of s_i
$V_k(s_i)$	A Voronoi vertex of $V(s_i)$
$A_{holes}(s_i)$	Total area of the holes lying within $V(s_i)$
A_Γ	Area covered by the nodes in the sub-optimal set Γ

The basic idea behind the first phase is to select a maximum number of nodes, such that none of their sensing circles overlap with each other, and they form a connected network. This essentially boils down to finding a *connected MIS* on the induced sensing graph. In case of certain special graphs, it is possible to find out the maximum IS in polynomial time; however, in case of RGG, upon which we base our network model, finding the maximum IS is NP-hard. It is well known and has been experimentally evaluated in [14], that for *random graphs* $G(N, p)$ [2], the standard randomized and greedy algorithms can generate an IS of size $log_{1/(1-p)}N$, with high probability for fixed p. However, in our case of induced sensing graph, where the sensing region is bounded and the degree of a node is directly proportional to its number of neighbors, we need to consider inter-node distances and maintain network connectivity while choosing the next node to be included in the *connected MIS*. Thus, the sensing radius, rather than the node density, plays a bigger role in determining the cardinality of the *connected MIS*, as derived analytically and verified by simulation later, for our RGG model. Here, we present a greedy algorithm that generates a large MIS on the induced sensing graph, and derive lower and upper bounds on the cardinality of the MIS in section 4.

In the second phase, the nodes that form the *connected MIS* in the first phase construct a localized Voronoi diagram. Each of the nodes then finds out coverage holes (regions in the sensing field that are not covered by any sensor) [15] within its Voronoi polygon and chooses the best nodes that can optimally cover those holes. Therefore, if each node follows the principle of optimally covering all the holes within its own polygon, then at the end of phase 2 the whole sensing field will get covered, and the selected set of nodes will form a sub-optimal MCSC. In the next subsections, we describe in detail the two phases of the algorithm and illustrate how they run on a grid and random deployment of nodes. We introduce some notations in Table 1 that are used in our discussion.

[2] N is the number of vertices and p is the probability that an edge exists between a pair of nodes.

3.1 Phase 1

The *connected MIS* finding algorithm starts with the node that has comparatively fewer neighbors, i.e., one of the perimeter nodes, because intuitively they will have fewer neighbors than the ones which are towards the center of the field. The successive steps in this phase differ from that of the standard greedy algorithms for random graphs, in the sense that we follow a distributed approach that makes it practically difficult at every step to choose the minimum degree node from the remaining eligible set of nodes. Also, we want the network to be connected at every step, and because of which our eligible set of nodes is constrained within one-hop neighbors of the last selected node. At every step, the last selected node chooses a new eligible node which is closest to itself, and includes it in Γ. A node s_j is called *eligible* to another node s_i, if it satisfies the following three criteria:

1. s_j has not yet been included in the *connected MIS*, Γ
2. s_j is a one-hop neighbor of s_i, i.e., $s_j \in N_{s_i}(R_s)$
3. s_j's sensing circle does not overlap with any of the already selected node's sensing circles.

Every node also informs the chosen closest eligible node about the area covered so far. However, if there exists no eligible node for s_i, then it passes over the responsibility to its farthest one-hop neighbor and requests it to choose the next node within that node's one-hop neighborhood, such that none of the sensing circles overlap with each other. This process continues until none of the nodes has any more eligible nodes left to choose from, at which point phase 1 terminates with Γ containing a *connected MIS* that covers a total sensing area of $|\Gamma|\pi R_s^2$ (because none of the sensing circles overlap with each other). Formalizing the afore-mentioned rules we state the *best eligibility criteria* as follows.

Definition 3. Best Eligibility Criteria*: If the the last selected node in Γ is s_i, then the best eligibility criteria for it to choose the next node s_j are the following:*

1. $s_j \in N_{s_i}(R_c - 2R_s) \setminus \left(N_{s_i}(R_c - 2R_s) \bigcap \left(\bigcup_{k=0}^{i-1} N_{s_k}(2R_s)\right)\right)$
2. $d(s_i, s_j) = \min\{d(s_i, s_k), \forall\, s_k \in N_{s_i}(R_c - 2R_s)\}$

The first condition makes sure that s_j's sensing circle does not overlap with the sensing circle of any other already selected node, while the second condition chooses the nearest eligible node of s_i. Note that, there can be more than one node satisfying the best eligibility criteria, in which case, one of them is chosen at random to break the tie. The stepwise description of phase 1 is given in Algorithm 1. Next, we illustrate the algorithm running on a grid and on randomly deployed set of nodes.

Grid Network: Let the nodes are deployed on the intersection points of a grid as shown in Fig. 3.(a). Without loss of generality, we assume that $R_c = 2R_s$. Phase 1 begins by choosing the first node s_0 having minimum node degree (though not unique in this particular case) that lies on the intersection of the first row and first column. In the next step, s_0 finds that there are two nodes s_1 and s_3, which tie on the best eligibility criteria. Let s_0 choose s_1 at random to break the tie. Next, s_1 chooses s_2 because that is the only node that satisfies the best eligibility criteria, and in a similar way, s_2 chooses s_3. At this point, phase 1 ends because no more nodes can be chosen without their sensing circles getting overlapped. Note that, these four nodes construct a MIS on the grid, which in this case is also the maximum IS. For special graphs like this, it is always possible to find the maximum IS in polynomial time.

Random Deployment: We illustrate few steps of phase 1 of the algorithm in case of random deployment as shown in Fig. 3.(b). Let s_0 be the first node chosen. In the second step, since s_1 is the nearest one-hop neighbor of s_0, which falls in the annular region and belongs to the set $N_{s_0}(R_c - 2R_s)$, it is included in Γ. In the fourth step, while it is s_2's responsibility to choose the next best eligible node, it chooses s_3 that satisfies the best eligibility criteria. That is, s_3 is the closest one-hop neighbor of s_2 that falls outside the hashed region in the diagram. This hashed region is where the set of nodes $N_{s_2}(R_c - 2R_s) \bigcap \left(\bigcup_{k=0}^{2} N_{s_k}(2R_s)\right)$ fall.

Algorithm 1. Phase 1: Distributed greedy algorithm to find a *connected MIS*

1: **Initialization:**
2: $\Gamma \leftarrow \phi$;
3: Choose the first node s_0 and include it in Γ; $s_b \leftarrow s_0$;
4: **Steps at each** s_b**:**
5: $N_{s_b}(R_c - 2R_s) \leftarrow \phi$;
6: **for all** $s_k \in N_{s_b}(R_c)$ **do**
7:    **if** $2R_s \leq d(s_b, s_k) \leq R_c$ **then**
8:       $N_{s_b}(R_c - 2R_s) \leftarrow N_{s_b}(R_c - 2R_s) \bigcup s_k$;
9:    **end if**
10: **end for**
11: **if** $N_{s_b}(R_c - 2R_s) \neq \phi$ **then**
12:    Find that $s_k \in N_{s_b}(R_c - 2R_s) \setminus \left(N_{s_b}(R_c - 2R_s) \bigcap \left(\bigcup_{s_j \in \Gamma, s_j \neq s_b} N_{s_j}(2R_s)\right)\right)$, such that $d(s_b, s_k)$ is minimum;
13:    $\Gamma \leftarrow \Gamma \cup s_k$;
14: **else if** $N_{s_b}(R_c - 2R_s) == \phi$ **then**
15:    $s_k \leftarrow s_q$, such that $d(s_b, s_q) = \max\{d(s_b, s_i), \forall s_i \in N_{s_b}(R_c)\}$;
16: **end if**
17: s_k becomes the next s_b to execute the same steps $5 - 16$.

3.2 Phase 2

In this phase, the nodes that were chosen in phase 1 construct a Voronoi diagram using neighborhood information. By properties of Voronoi diagram, the number of edges of the Voronoi polygon for node s_i is equal to the number of its one-hop neighbors. Now, since the nodes selected in phase 1 do not have their sensing circles overlapped with each other, there will definitely exist holes in each of the

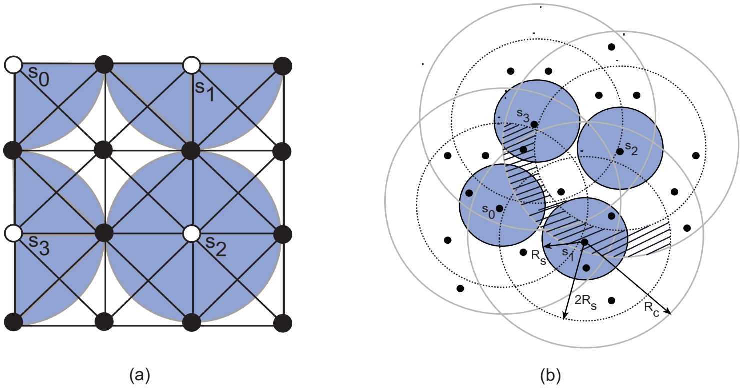


Fig. 3. (a) The induced sensing graph and the total coverage (shaded area) achieved by the four nodes (s_0, s_1, s_2, s_3), selected in phase 1. (b) Best Eligibility Criteria for a set of nodes deployed randomly

Voronoi polygons. In the first step of phase 2, one of these nodes $s_i \in \Gamma$ is selected at random, which then finds out holes existing within its polygon by the method described in [15]. It basically splits the convex Voronoi polygon into a set of disjoint and mutually exhaustive triangles, and using simple techniques of line and curve intersections finds out the area of coverage holes lying within its polygon. Next, the best one-hop neighbor is chosen, such that it covers maximum amount of hole within its polygon. The criteria for choosing the best node is as follows. Node s_i determines a set of optimal points $P_{s_i} = \{p^k_{s_i}, k = 1, ..., |N_{s_i}(R_c)|\}$, in the neighborhood of each of its Voronoi vertex $V_k(s_i)$. Each optimal point satisfies the following rules [15]:

1. Rule1: $p^k_{s_i}$ should lie on the angle bisector of the Voronoi vertex,
2. Rule2: $d(s_i, p^k_{s_i}) = \min\{2R_s, d(s_i, V(s_i))\}$.

These criteria ensure that if a node is placed at $p^k_{s_i}$, then the amount of coverage hole lying in the vicinity of $V_k(s_i)$ will get eliminated maximally. From these optimal locations, node s_i also finds out approximate estimates $Cov(p^k_{s_i})$ of the amount of coverage holes that will get eliminated if nodes were placed at these optimal locations. This is illustrated in Fig. 4. Nodes A, B, C, D form Voronoi diagram, and let A be the first node to eliminate holes within its polygon. P, Q, R are the intersection points of the Voronoi edges with the lines that connect the nodes. From the properties of Voronoi diagram, $Area(\Delta AVP) = Area(\Delta BVP)$ and $Area(\Delta AVR) = Area(\Delta BVR)$, and node A knows that if it chooses the best node (s_b) close to vertex V, then that node will also eliminate almost an equal amount of holes in the polygons of B and C combined, as it will do within its own polygon. The point which corresponds to maximum such elimination of holes is chosen as the best location for the next best node.

Once the new best node s_b is selected, node s_i rechecks whether there still exist holes within its polygon. If so, it recalculates the area of the holes lying

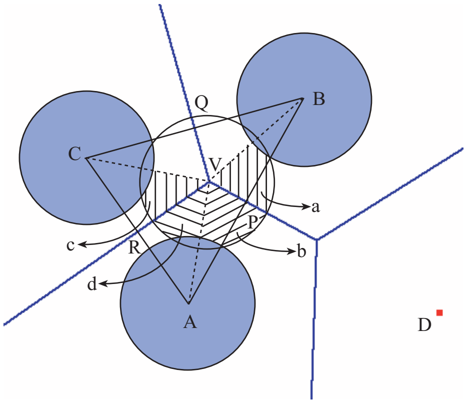


Fig. 4. Optimal position to choose a best node using Voronoi diagram. The hashed areas are equal: $a = b, c = d$

in the vicinity of the Voronoi vertices, except the one where s_b lies and repeats the same steps to choose another best node. This process continues until all the holes get eliminated from s_i's polygon, at which point, one of the neighbor nodes of s_i that was part of the *connected MIS* in phase 1 gets the chance to fill the holes within its polygon and so on. Thus, at the end of this phase, when each node chosen from the first phase have guaranteed the existence of no more holes within their polygons, the monitoring region gets completely covered with a connected set of nodes that form the sub-optimal MCSC. Phase 2 of the algorithm is described in Algorithm 2.

4 Analysis and Simulation

In this section, we present our analysis and preliminary simulation results of the distributed greedy algorithm to calculate a sub-optimal MCSC. We derive lower and upper bounds on the cardinality of the MIS generated in phase 1, using which we calculate an approximate size of the sub-optimal MCSC. We also provide a brief discussion on time complexity analysis for both the phases of the algorithm.

4.1 Analysis

Let the total number of nodes deployed in the square sensing field of area A, following a uniform random distribution be N, and let the cardinality of the *connected MIS* generated in phase 1 be ζ. At any step during the first phase, when a node gets included in Γ, all nodes that lie within a distance of $2R_s$ from that node become ineligible to be included in Γ at a latter step. For instance, if the first node chosen falls atleast $2R_s$ distance away from any edge of the field, then the number of nodes that become ineligible is $4\rho\pi R_s^2 - 1$, where $\rho = N/A$ denotes uniform node density (here 1 is subtracted for the node which gets

Algorithm 2. Phase 2 of the sub-optimal MCSC algorithm

Notations:
  S_{MIS}: Set of *connected MIS* that were selected in Phase 1;
Initialization:
  $S_{MIS} \leftarrow \Gamma$, $A_\Gamma \leftarrow \pi |S_{MIS}| R_s^2$;
  Nodes $\in S_{MIS}$ construct a localized Voronoi diagram;
  Randomly choose one of the nodes $s_i \in S_{MIS}$ as the starting node;
Steps at each $s_i \in S_{MIS}$**:**
if $A_\Gamma < A_Q$ **then**
  Calculate $A_{holes}(s_i)$;
  while $A_{holes}(s_i) \neq 0$ **do**
    Find optimal points $\{p_{s_i}^k\}$ that satisfy Rules 1, 2 near all unmarked $V_k(s_i)$;
    Calculate $Cov(p_{s_i}^k)$ for each optimal point;
    Choose the point for which $Cov(p_{s_i}^k)$ is maximum, call it $p_{s_i}^b$;
    s_i chooses the node (s_b) closest to $p_{s_i}^b$ and includes it in Γ;
    Update $A_{holes}(s_i)$, A_Γ and mark the vertex near $p_{s_i}^b$;
    s_i informs its neighbors of the amount of holes it eliminated, so that they can update their calculations;
  end while
end if

included). Note that, this is the maximum number of ineligible nodes at any step, because there could be some nodes which are ineligible to more than one node, and hence, should be counted only once. Similarly, the minimum number of ineligible nodes at any step is zero. Therefore, let the number of ineligible nodes at any step be $\beta(4\rho\pi R_s^2 - 1)$, where $0 \leq \beta \leq 1$.

Now, since the ζ nodes do not have their sensing circles overlapped with each other, the total number of nodes that lie within those ζ sensing circles is $\zeta\rho\pi R_s^2$. The remaining $(N - \zeta\rho\pi R_s^2)$ nodes must have become ineligible at some step while finding the *connected MIS*. Therefore, we have:

$$\zeta\beta(4\rho\pi R_s^2 - 1) = (N - \zeta\rho\pi R_s^2) \tag{2}$$

or

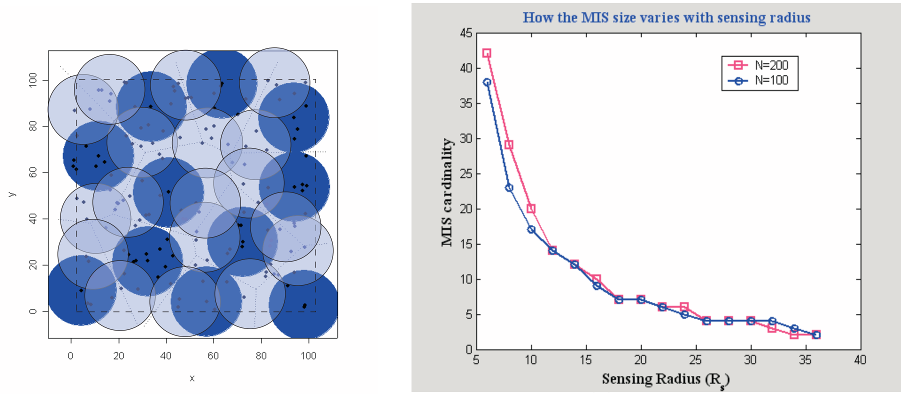
$$\zeta = \frac{N}{\rho\pi R_s^2(1 + 4\beta - \beta/\rho\pi R_s^2)} \tag{3}$$

Neglecting the term $\beta/\rho\pi R_s^2$ because it is very small, we get

$$\zeta = \frac{N}{\rho\pi R_s^2(1 + 4\beta)}. \tag{4}$$

Now, substituting $\beta = 0$ and $\beta = 1$ we get the upper and lower bounds, respectively, for the cardinality of the *connected MIS*, which gives us

$$\frac{N}{5\rho\pi R_s^2} \leq \zeta \leq \frac{N}{\rho\pi R_s^2}. \tag{5}$$



(a) (b)

Fig. 5. Simulation: Dark circles represent sensing ranges of nodes belonging to *connected MIS*: (a) $N = 150$, $R = 15m$, $A_Q = 10,000m^2$. (b) Variation of *connected MIS* cardinality with sensing radius R_s

Using Eq. (4) and assuming that each of the ζ nodes on the average selects η nodes in phase 2 to optimally cover the holes within its Voronoi polygon, we can estimate the cardinality of the sub-optimal MCSC Γ as,

$$|\Gamma| = \zeta(1+\eta) = \frac{N(1+\eta)}{\rho\pi R_s^2(1+4\beta)}. \qquad (6)$$

The time complexity of the first phase of the algorithm is output sensitive, i.e., it depends on the cardinality of the *connected MIS* generated. Now, since the "for loop" in Algorithm 1, in the worst case, runs over all the one-hop neighbors of a node while choosing the best eligible node, the time complexity of the first phase is O(ζN). For deriving the complexity of the second phase, note that the localized Voronoi diagram is constructed by the nodes in the *connected MIS*. This can be performed in O($\zeta log\zeta$) time. Next, each node checks for coverage holes in the vicinity of each of its Voronoi vertices, the time complexity of which is bounded by its number of one-hop neighbors. Hence, time complexity of the second phase is O($\zeta log\zeta$).

4.2 Simulation

We considered a sensing field of size 100 x 100 m^2 (the query region A_Q is also assumed to be of the same size) and performed simulation using Matlab and GNU R, a software package for statistical computing and graphics. We deployed the nodes uniformly randomly and varied their number as well as their sensing radii, and assumed that the communication radius is thrice the sensing radius. In Fig. 5.(a), we show a sample simulation run for $A_Q = 10,000m^2, N = 150, R = 15m$. The 11 darker nodes form the MIS in the first phase, which then construct Voronoi diagram to select another 18 nodes (the ones with ligher shade in the figure) to optimally cover the holes. In Fig. 5.(b), we show the variation of the *connected MIS* size with respect to sensing radius. Note that, the cardinality of the *connected MIS* satisfies the bounds given by Eq. (5).

5 Conclusions

In this paper, we described a distributed greedy algorithm for generating a sub-optimal minimum connected sensor cover (MCSC) for homogeneous dense sensor networks. Possibility to extend network lifetime by exploiting node redundancy and guaranteeing 100% coverage of the query region has been the motivation to this problem. We used the concepts of independent sets and Voronoi diagrams to construct a sub-optimal MCSC. We also provided upper and lower bounds on the cardinality of the MIS for random geometric graph model used to model the sensor network. The exact determination of the heuristic parameter, η, that we use to find the size of the sub-optimal MCSC is part of our future work. We also plan to do extensive simulations and derive tighter bounds on the size of the MIS. Instead of constructing the connected cover set greedily in a single phase, like in [8], where the sensing ranges of nodes could overlap right from the first step, we follow a different strategy of finding a large set (the *connected MIS*) of nodes in the first phase where none of the sensing circles overlap with each other. This gives us the maximum coverage that can be achieved by the set of nodes in phase 1, which is $\zeta \pi R_s^2$ (where, ζ is given by Eq. (4). In phase 2, since these set of non overlapping nodes optimally cover the holes within their Voronoi polygons, the sensing field can be covered with a number of nodes, that is close to optimal. The exact derivation of this sub-optimal bound is part of our future work.

Acknowledgment

This work is supported by NSF ITR grant under Award No. IIS-0326505.

References

1. Di Tian and Nicolas D. Georganas, "A coverage-preserving node scheduling scheme for large wireless sensor networks", *in Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications - WSNA'02*, pp. 32–41, Atlanta, Georgia, USA, Sep 2002.
2. Yong Gao, Kui Wu, and Fulu Li, "Analysis on the redundancy of wireless sensor networks", *in Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications - WSNA'03*, pp. 108–114, San Diego, California, USA, Sep 2003. ACM Press.
3. Chi-Fu Huang and Yu-Chee Tseng, "The coverage problem in a wireless sensor network", *in Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications - WSNA'03*, pp. 115–121, San Diego, CA, Sep 2003.

4. Xiaorui Wang, Guoliang Xing, Yuanfang Zhang, Chenyang Lu, Robert Pless, and Christopher Gill, "Integrated coverage and connectivity configuration in wireless sensor networks", *in Proceedings of the 1st International Conference on Embedded Networked Sensor Systems - Sensys'03*, pp. 28–39, Los Angeles, CA, Nov 2003.
5. Honghai Zhang and Jennifer C Hou, "Maintaining sensing coverage and connectivity in large sensor networks", *in Proceedings of the International Workshop on Theoretical and Algorithmic Aspects of Sensor, Ad hoc Wireless and Peer-to-Peer Networks - AlgoSensors'04*, Florida, USA, Feb 2004.
6. Amitabha Ghosh and Sajal K. Das, *Coverage and Connectivity Issues in Wireless Sensor Networks: Wireless, Mobile and Sensor Networks*, John Wiley & Sons, 2005.
7. V. S. Anil Kumar, Sunil Arya, and H. Ramesh, "Hardness of set cover with intersection 1", *in Proceedings of the 27th International Colloquium on Automata, Languages and Programming - ICALP'00*, pp. 624–635, Geneva, Switzerland, July 2000. Springer-Verlag.
8. Himanshu Gupta, Samir R. Das, and Quinyi Gu, "Connected sensor cover: self-organization of sensor networks for efficient query execution", *in Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing - MobiHoc'03*, pp. 189–200, Annapolis, Maryland, USA, Jun 2003. ACM Press.
9. Zongheng Zhou, Samir Das, and Himanshu Gupta, "Variable radii connected sensor cover in sensor networks", *in Proceedings of the 1st IEEE International Conference on Sensor and Ad Hoc Communications and Networks - SECON'04*, pp. 387–396, Santa Clara, California USA, Oct 2004.
10. Stefac Funke, Alex Kesselman, Zvi Lotker, and Michael Segal, "Improved approximation algorithms for connected sensor cover", *in Proceedings of 3rd International Conference on Ad Hoc Networks & Wireless ADHOC-NOW'04*, pp. 56–69, Vancouver, British Columbia, Canada, July 2004.
11. Neal Patwari and III Alfred O. Hero, "Using proximity and quantized rss for sensor localization in wireless networks", *in Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications - WSNA'03*, pp. 20–29, San Diego, CA, Sep 2003.
12. Michael R. Garey and Davis S. Johnson, *Computers and Intractability, a Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, 1991.
13. Franz Aurenhammer, "Voronoi diagrams a survey of a fundamental geometric data structure", *ACM Computing Surveys*, vol. 23, pp. 345–405, 1991.
14. Mark K. Goldberg, D. Hollinger, and M. Magdon-Ismail, "Experimental evaluation of the greedy and random algorithms for finding independent sets in random graphs.", *in Proceedings of the 4th International Workshop on Efficient and Experimental Algorithms, WEA'05*, Santorini Island, Greece, May 2005 (to appear).
15. Amitabha Ghosh, "Estimating coverage holes and enhancing coverage in mixed sensor networks", *in Proceedings of the 29th Annual IEEE Conference on Local Computer Networks, LCN'04*, pp. 68–76, Tampa, Florida, Nov 2004.