Coverage in Wireless Ad-hoc Sensor Networks

Xiang-Yang Li, Peng-Jun Wan, Ophir Frieder

Abstract—Sensor networks pose a number of challenging conceptual and optimization problems such as location, deployment, and tracking [1]. One of the fundamental problems in sensor networks is the calculation of the coverage. In [1], it is assumed that the sensor has the uniform sensing ability. In this paper, we give efficient distributed algorithms to optimally solve the *best-coverage problem* raised in [1]. Here, we consider the sensing model: the sensing ability diminishes as the distance increases. As energy conservation is a major concern in wireless (or sensor) networks, we also consider how to find an optimum best-coverage-path with the least energy consumption. We also consider how to find an optimum best-coverage-path that travels a small distance. In addition, we justify the correctness of the method proposed in [1] that uses the Delaunay triangulation to solve the best coverage problem. Moreover, we show that the search space of the best coverage problem can be confined to the relative neighborhood graph, which can be constructed locally.

I. Introduction

Sensor networks pose a number of challenging conceptual and optimization problems such as location, deployment, and tracking [1]. In a pioneering work by Meguerdichian, *et al.*[1], they addressed one of the fundamental problems, namely coverage, which in general answers the questions about the quality of service that can be provided by a particular sensor network. They gave polynomial time algorithms to solve the questions optimally. However, their algorithms rely heavily on some geometrical structures such as the Delaunay triangulation and the Voronoi diagram which can not be constructed locally or even efficiently in a distributed manner. In addition, the correctness of using these two geometry structures is not presented in their paper.

In a wireless ad hoc network (or sensor network), each wireless node has a maximum transmission power so that it can send signals to all nodes within its transmission range. If a node v is not within the transmission range of the sender u, nodes u and v communicate through multi-hop wireless links by using intermediate nodes to relay the message. Each node in the wireless network also acts as a router, forwarding data packets for other nodes. We assume that each static wireless node knows its position information, either through a low-power Global Position System (GPS) receiver or through some other approach. For simplicity, we also assume that all wireless nodes have the same maximum transmission range, and we normalize it to one unit. By a simple broadcasting, each node u can gather the location information of all nodes within the transmission range of u. We also assume that all wireless nodes have distinctive identities. Consequently, all wireless nodes S together define a unit disk graph $UDG(S)$, which has an edge uv if and only if the Euclidean distance between u and v is less than one unit. We call all nodes within a constant k hops of a node u in the unit disk graph $UDG(S)$ as the *k-local nodes* of u. Usually, here the constant k is 1 or 2, denoted by $N_k(u)$, which will be omitted if it is clear from the context.

A distributed algorithm is a *localized algorithm* if it uses only the information of all local nodes plus the information of a constant number of additional nodes. A graph G can be constructed locally in the ad hoc wireless environment if each wireless node u can compute the edges of G incident on u by using only the location information of all local nodes. We are interested in designing a localized algorithm that finds a path connecting a point s and a point t, which maximizes the smallest observability of all points on the path. It is called the *best coverage problem*. On the other hand, the *minimum exposure problem* [2] is to find a path that connecting two points in the domain with the minimum integral observability. While the *worst coverage problem* is to find the path that maximizes the distance of the path to all sensor nodes. We provide efficient distributed algorithms to solve the best coverage problems. In addition, we justify the correctness of using the Delaunay triangulation to solve the best coverage problem. Moreover, we show that the search space of the best coverage problem can be confined to the relative neighborhood graph, which can be constructed locally.

Dept. of Comp. Sci., Illinois Institute of Technology, Chicago, IL, 60616. Email: xli@cs.iit.edu

The rest of the paper is organized as follows. In Section II, we give some preliminary definitions and notations that will be used in presenting our algorithms. We also briefly review the algorithms by Meguerdichian, *et al.*[1] and outline some discrepancies in their algorithms. In Section III, we present the first localized algorithm that solves the best coverage problem efficiently. We also discuss several extensions of the best-coverage problem. Specifically, we consider how to find an optimum best-coverage-path that conserves the energy, and how to find an optimum best-coverage-path with small travelling distance. Both the correctness of our algorithm and the correctness of the algorithm by Meguerdichian *et al.* are justified in Section IV. We conclude our paper and discuss possible future research directions in Section V.

II. Preliminaries

A. Problem formulation

We assume that the wireless sensor nodes are given as a set of n points S distributed inside a two-dimensional domain Ω. For simplicity, we assume that the domain Ω is given as a *planar-straight-line graph* (PSLG), which is a collection of line segments and points in the plane, closed under intersection. Let B be the set of points that define the domain boundary. For simplicity, we assume that the convex hull $CH(S)$ of the set of sensors S is contained inside the domain Ω. We also assume that every wireless node has the same maximum transmission range. Then the set of wireless sensors S defines a unit disk graph $UDG(S)$. We always assume that the graph $UDG(S)$ is connected.

We first give some geometry notations that are used in the remainder of this Section to mathematically formulate the problems considered. Let $\|xy\|$ denote the Euclidean distance of two points x and y.

Definition 1: The distance of a point x to a set of points V, denoted by $dist(x, V)$, is the smallest distance of x to all points of V. In other words, $dist(x, V) = \min_{y \in V} \|xy\|$.

Notice that the point set V may be infinite. For example, V could be all points lying on a segment uv. We use $dist(x, uv)$ to denote the smallest distance from x to all points on the segment uv.

Definition 2: The *coverage-distance* of a point set U by another point set V, denoted by $cover(U, V)$, is the maximum distance of every point $x \in U$ to V. That is, $cover(U, V) = \max_{x \in U} dist(x, V)$.

Notice that the breach distance $dist(U, V)$ is symmetric, i.e., $dist(U, V) = dist(V, U)$, while the coverage distance $cover(U, V)$ is *not* symmetric and that both point sets U and V can be infinite. For example, U can be a path connecting two points s and t and V all sensor nodes. Given a path $\pi(s,t)$ inside Ω connecting s and t, the *coverage-distance* $\max_{x \in \pi(s,t)} dist(x, S)$ of the path $\pi(s,t)$ specifies how well the path is protected by the sensors, while, on the reverse side, the breach distance $\min_{x \in \pi(s,t)} dist(x, S)$ specifies how far the path is from all sensors. Thus, for wireless sensor networks, the coverage problem has two folds: the best coverage and the worst coverage, which are defined as follows.

Given a set of sensors S inside a two-dimensional domain Ω, a starting point $s \in \Omega$, and an ending point $t \in \Omega$, we find a path $\pi(s,t)$ inside Ω to connect s and t such that the coverage distance $cover(\pi(s,t), S) = \max_{x \in \pi(s,t)} dist(x, S)$ is minimized. In other

words, we try to find a path connecting s and t such that every point x of the path is covered by some sensor nodes with small distance.

Definition 3: A path $\pi(s,t)$ that achieves the minimum coverage-distance $cover(\pi(s,t),S)$ is called a *best-coverage-path*. The minimum coverage-distance $cover(\pi(s,t),S)$ of all paths connecting s and t is called the *best-coverage-distance* or the *support-distance*.

This problem has several interesting applications. For example, consider a war-field denoted by a two-dimensional domain Ω. Assume that a postman soldier wants to travel from position s to position t in Ω. There are some randomly distributed protection soldiers, denoted by a set of points S, which will protect the postman soldier. Then it is always desirable to find a path in Ω such that the maximum distance of the postman soldier from the protection soldiers is minimized.

There are several variations for the best coverage problem. Notice that, as shown later, there are many paths that achieves the *best-coverage-distance*. As the energy consumption is a critical issue in the wireless networks, we wish to find a path that consumes the least energy possible while it still achieves the best-coverage-distance. The other variation is to find a path with the minimum total travelling distance among all optimum paths with the best-coverage-distance. This is justified by the above postman soldier example.

Sensing devices generally have widely different theoretical and physical characteristics. Interestingly, in most sensing device models, the sensing ability diminishes as distance increases. Let $S(s,p)$ be the sensing ability of sensor s at point p. When point p is out of the sensing range of the sensor s, i.e., $\|sp\|>1$, then $S(s,p)=0$. Notice that the sensing range is normalized to one unit here. In [2], they assumed that $S(s,p)=\frac{\lambda}{\|sp\|^{\kappa}}$ for sensor-technology dependent parameters λ and κ for conducting simulations. In this paper, we adopt the following sensing model.

1. The sensing ability of every sensor device is uniform. In other words, $S(s_i,u)=S(s_j,v)$ if $\|s_iu\|=\|s_jv\|$.
2. The sensing ability satisfies a *monotone property*: $S(s,u)>S(s,v)$ if $\|su\|<\|sv\|$.

Given a point p, its closest-sensor observability $I_c(p)$ is defined as $S(s_p,p)$, where s_p is the closest sensor to point p. The all-sensor observability of point p, denoted by $I_a(p)$, is defined as $\sum_{s_i}S(s_i,p)$. Given a path $\pi(s,t)$ connecting points s and t, its closest-sensor observability is defined as $I_c(\pi(s,t))=\min_{p\in\pi(s,t)}I_c(p)$. Similarly, we define the all-sensor observability as $I_a(\pi(s,t))=\min_{p\in\pi(s,t)}I_a(p)$. In this paper, we are interested in finding a path with the maximum closest-sensor observability. In other words, we try to find a path connecting s and t such that all point on the path is well-observed by the sensors. From the definitions of the best-coverage-path and the closest-sensor observability, it is easy to show that the best-coverage-path also achieves the maximum closest-sensor observability. Consequently, in the rest of the paper, we only have to study how to find the best-coverage-path, which also achieves the maximum closest-sensor observability.

B. Geometry Notations

Delaunay triangulation and Voronoi diagram are widely used in many areas. We begin with definitions of the Voronoi diagram and the Delaunay triangulation. We assume that all wireless nodes are given as a set S of n vertices in a two dimensional space. Each node has some computational power. We also assume that there are no four vertices of S that are co-circular. A triangulation of S is a *Delaunay triangulation*, denoted by $Del(S)$, if the circumcircle of each of its triangles does not contain any other vertices of S in its interior. A triangle is called the *Delaunay triangle* if its circumcircle is empty of vertices of S. The *Voronoi region*, denoted by $Vor(p)$, of a vertex p in S is a collection of two dimensional points such that every point is closer to p than to any other vertex of S. The *Voronoi diagram* for S is the union of all Voronoi regions $Vor(p)$, where $p\in S$. The Delaunay triangulation $Del(S)$ is also the dual of the Voronoi diagram: two vertices p and q are connected in $Del(S)$ if and only if $Vor(p)$ and $Vor(q)$ share a common boundary. The shared boundary of two Voronoi regions $Vor(p)$ and $Vor(q)$ is on the perpendicular bisector line of segment pq. The boundary segment of a Voronoi region is called the *Voronoi edge*. The intersection point of two Voronoi edge is called the *Voronoi vertex*. When there is no four points of S that are co-circular, then every Voronoi vertex has only exactly three Voronoi edges incident on it. The Voronoi vertex is the circumcenter of some Delaunay triangle.

It is not appropriate, however, to require the construction of the Delaunay triangulation in the wireless communication environment because of the possible massive communications it requires. Using centralized method, it requires at least $O(n\log n)$ bits-communications to collect all nodes' coordinates. It also requires massive communications to broadcast the triangulation to all nodes. Therefore, Li [3] studied a subset of the Delaunay triangulation. Let $UDel(S)$ be the graph by removing all edges of $Del(S)$ that are longer than one unit, i.e., $UDel(S)=Del(S)\cap UDG(S)$. Call $UDel(S)$ the *unit Delaunay triangulation*. Li, *et al.* [3], [4] provided an efficient localized algorithm that constructs a planar graph, called localized Delaunay triangulation $LDel(S)$, which contains $UDel(S)$ as a subgraph. Thus, the constructed graph can be used by almost all algorithms that require to use the structure $UDel(S)$ or even $Del(S)$. The graph $LDel(S)$ is constructed as follows [4]:

1. Each node u broadcasts its identity and location and listens to messages from other nodes. Node u then computes $Del(N_1(u))$ of its 1-neighbors $N_1(u)$, including u itself. Node u marks all *Gabriel edges* uv, which will never be deleted.
2. Fro each $\triangle uvw\in Del(N_1(u))$ such that $\angle wuv\geq\frac{\pi}{3}$, node u broadcasts a message proposal(u,v,w) to form a triangle $\triangle uvw$ in $LDel^{(1)}(V)$.
3. When node v receives proposal(u,v,w), v accepts it by broadcasting accept(u,v,w) if $\triangle uvw\in Del(N_1(u))$; otherwise, it rejects it by broadcasting reject(u,v,w). Similarly does node w. Node u accepts $\triangle uvw$ if both nodes v and w accept proposal(u,v,w). Similarly do node v and w.

This constructs $LDel^{(1)}(S)$. Each node v broadcasts the edges of $LDel^{(1)}(S)$ incident on v. Assume node u gathered 1-local Delaunay triangles of each $v\in N_1(u)$. For two intersected triangles $\triangle x_1y_1z_1$ and $\triangle x_2y_2z_2$ known by u, u removes $\triangle x_1y_1z_1$ if its circumcircle contains a node from $\{x_2,y_2,z_2\}$. Node u broadcasts all remaining triangles incident on u to $N_1(u)$. Node u keeps triangle $\triangle uvw$ if both v and w have triangle $\triangle uvw$ remained. The collection of edges kept by all nodes is called planar $LDel(S)$.

Various proximity subgraphs of the unit disk graph were studied [5], [6]. For convenience, let $disk(u,v)$ be the closed disk with diameter uv; let $disk(u,v,w)$ be the circumcircle defined by the triangle $\triangle uvw$; let $B(u,r)$ be the circle centered at u with radius r. Call the interior of the intersection $B(u,\|uv\|)\cap B(v,\|uv\|)$ the *lune*, denoted by $lune(u,v)$, defined by two points u and v. The constrained *relative neighborhood graph* $RNG(V)$ over a point set V has an edge (u,v) if the $lune(u,v)$ does not contain any point from V in the interior. The *constrained Gabriel graph* of a point set V, denoted by $GG(V)$, consists of all edges uv such that $\|uv\|\leq 1$ and the $disk(u,v)$ does not contain any node from V.

C. Discussion of Previous Algorithms

In [1], Meguerdichian, *et al.* developed centralized algorithms to solve the best coverage problem using the Delaunay triangulation. No justification as to why the search space can be confined to the Delaunay triangulation for the best coverage problem was provided. We later provide a formal proof that there is an optimum best-coverage-path that uses only the edges of Delaunay triangulation. See Section IV. In their algorithm, they connect the starting point s to its closest sensor node u_s and connect the ending point t to its closest sensor node u_t. This is based on intuition [1]. We formally prove that there is an optimum best-coverage-path with this property if the unit disk graph $UDG(S)$ is connected.

To find an optimum best-coverage-path, they assign each Delaunay edge a weight equal to its Euclidean distance and then apply some graph algorithms on it. We show that this is likewise erro-

neous. Here the weight of an edge uv denotes the maximum distance $\max_{x \in uv}\ dist(x, S)$ of every point x on uv to its closest sensor. Remember that we want to find a path such that the maximum distance of all points of this path to its closest sensor node is minimized. Thus, the weight of an edge uv should be $\max_{x \in uv}\ dist(x, S)$, which is *at most* $\frac{1}{2}\|uv\|$. It is insufficient to just consider the midpoint of a Delaunay edge uv to compute its weight $\max_{x \in uv}\ dist(x, S)$. This is because it is possible that there is some other sensor node w that lies inside $dist(u, v)$. See Figure 1. The weight of the Delaunay edge uv is less than $\frac{1}{2}\|uv\|$. We later will show that the search space can be confined to a much smaller graph, namely the relative neighborhood graph. Moreover, the weight of each edge uv in that graph is guaranteed to be *exactly* $\frac{1}{2}\|uv\|$.

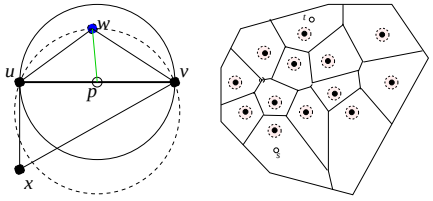

Fig. 1. Left: The weight of a Delaunay edge uv is less than $\frac{1}{2}\|uv\|$. Right: Each sensor node has a disk specifying the region covered by it. All disks have the same radius and grow with the same speed.

D. Growing Disks

Assume that every sensor node originally has a disk centered at it with radius 0 and every disk starts growing with the same speed. See Figure 1. Let $D(S, r)$ be the region covered by all disks centered at points of S with radius r. Let $\overline{D(S, r)}$ be the complementary region of $D(S, r)$ in domain Ω. Then the best coverage problem asks what is the smallest radius value r such that there is a path, inside the region $D(S, r)$, connecting points s and t. On the other hand, the worst coverage problem asks what is the largest radius value r such that there is a path, inside the region $\overline{D(S, r)}$, connecting points s and t.

III. The Best Coverage Problem

A. Algorithm

We first give an efficient distributed algorithm that solves the best coverage problem optimally. Here assume that we are given a set of sensors S, a starting point s, and an ending point t in a two-dimensional domain Ω such that the unit disk graph $UDG(S)$ is connected and the convex hull $CH(S)$ of S is contained inside Ω.

Algorithm 1: FindBestCoverage(S, Ω, s, t)

1. Find the closest sensor node of the starting point s if itself is not a sensor node. Assume u_s is the closest sensor node. Similarly, find the closest sensor node u_t of the ending point t.
2. Each sensor node u locally constructs all edges uv of the relative neighborhood graph $RNG(S)$, where v is also a sensor node. This can be constructed as follows. Each node u broadcasts its location information and listens to the broadcasting by its neighbors. Thus, after this step, we assume that each node u has the location information of $N_1(u)$. Node u adds an edge uv if and only if the $lune(u, v)$ does not contain any nodes from $N_1(u)$ inside.
3. Assign each constructed edge uv with weight $\frac{1}{2}\|uv\|$.
4. Run a distributed *shortest path* algorithm to compute the *shortest path* connecting u_s and u_t. Here, the *weight* of a path is the maximum weight of all of its edges. Here a path is the shortest path if it has the minimum weight among all paths connecting u_s and u_t.The Bellman-Ford algorithm [7] can be modified to solve this shortest path problem.
5. Let $\pi(u_s, u_t)$ be a computed path and $\|\pi(u_s, u_t)\|$ be the weight of the path. Then the path concatenating the edge su_s, path $\pi(u_s, u_t)$, and the edge $u_t t$ is an optimum best-coverage-path. The best-coverage-distance is $\max(\|su_s\|,\|\pi(u_s, u_t)\|,\|u_t t\|)$. Here $\|su_s\|$ and $\|u_t t\|$ are the Euclidean distance between points.

B. The time and communication complexity

constructing the relative neighborhood graph $RNG(S)$, and then apply the Bellman-Ford algorithm [7] to find the *shortest path* between nodes u_s and u_t. The time complexity of this centralized algorithm is $O(n \log n)$: The first step costs $O(n)$ time; we can construct the relative neighborhood graph in $O(n \log n)$ time; and we can compute the shortest path connecting two vertices in a planar graph in time $O(n \log n)$. This algorithm is much more efficient than that given in [1], which has time complexity $O(n^2 \log n)$.

For wireless sensor networks, however, it is impractical to collect the location information of all sensors due to the massive communication it requires. Thus, a distributed algorithm is a must. Notice that the relative neighborhood graph of all sensors S can be constructed efficiently by using a localized approach. Therefore, the communication cost is also small as compared to collecting the locations information of all sensor nodes. The communication cost of constructing the graph $RNG(S)$ using a distributed manner is $O(n \log n)$ bits. We assume that the identity of each wireless node can be represented by $O(\log n)$ bits and the geometry information can be represented by $O(1)$ bits.

C. Extensions

In addition, we consider some extensions of the best coverage problem by presenting efficient distributed algorithms to solve them. Notice that, the coverage-distance of two points s and t depends on their distances to the closest sensors. If we want to improve the coverage-distance of all pairs of points in the domain by adding more sensors, these new sensors should be placed at the circumcenters of Delaunay triangles that have the largest circumradii.

C.1 Energy Conservation

The first extension is to find a path with the best-coverage-distance while the total energy consumed by this path is minimized among all optimum best-coverage-paths. We assume that the energy needed to support a link uv is proportional to $\|uv\|^\alpha$, where α is a real constant between 2 and 5. In best-coverage problem, finding areas of high observability from sensors and identifying the best support and guidance regions are of primary concern [1]. For example, in a fire detection sensor networks, it is not only required that the sensor networks observe well a given region and it is also necessary that the sensor that detects the fire can report the fire to a center station efficiently.

Algorithm 2: EnergyConsrvngBestCoverage(S, Ω, s, t)

1. Run a distributed *shortest path* algorithm to compute the *coverage distance* of the best-coverage-path connecting u_s and u_t. Let β be the best coverage distance.
2. Construct the Gabriel graph $GG(S)$ and prune out all edges of the Gabriel graph $GG(S)$ with weight larger than β, and call the remaining graph the residue graph G.
3. Assign each edge uv of the residue graph G the weight equal to $\|uv\|^\alpha$, where α is the propagation constant depending on the environment.
4. Run a distributed *shortest path* algorithm to compute the *shortest path* connecting u_s and u_t. Here, the *weight* of a path is the total weight of all of its edges. A path is the shortest path if it has the minimum weight among all paths connecting u_s and u_t.
5. Let $\pi(u_s, u_t)$ be a computed path and $\|\pi(u_s, u_t)\|$ be the weight of the path. The path concatenating the edge su_s, path $\pi(u_s, u_t)$, and the edge $u_t t$ is an optimum best-coverage-path with the minimum energy consumption. The best-coverage-distance is $\max(\|su_s\|,\beta,\|u_t t\|)$. Here $\|su_s\|$ and $\|u_t t\|$ are the Euclidean distance between points.

The correctness of the algorithm is based on the following observation. Consider an edge uv of the best-coverage-path that consumes the minimum energy among all best-coverage-paths. If there is a sensor node w inside $disk(u, v)$, then $\|wu\| \leq \|uv\|$ and $\|wv\| \leq \|uv\|$. It is obvious that the path uwv is in the residue graph G. Thus, the path by substituting edge uv with edges uw and wv is still a best-coverage-path and consumes less energy, which is a contradiction. Consequently, edge uv must be a Gabriel edge.

The time complexity of the above algorithm is $O(n \log n)$ if it is implemented using a centralized manner. The total communication cost by all wireless nodes of the above algorithm is $O(n \log n)$ bits if it is implemented in a distributed manner. Here, we assume that we use a synchronized distributed algorithm to construct the shortest path between two given wireless nodes.

C.2 Travel Distance

The second extension is to find a path with the best-coverage-distance with the total length of the edges of this path is not more than $5/2$ times the shortest path among all optimum best-coverage-paths.

Algorithm 3: SmallTravellingBestCoverage(S, Ω, s, t)

1. Run a distributed *shortest path* algorithm to compute the *coverage distance* of the best-coverage-path connecting u_s and u_t. Let β be the best coverage distance.
2. Construct the local Delaunay triangulation. Prune out all edges of the local Delaunay triangulation $LDel(S)$ with weight larger than β, and call the remaining graph the residue graph G.
3. Assign each edge uv of the residue graph G the weight equal to $\|uv\|$.
4. Run a distributed *shortest path* algorithm to compute the *shortest path* connecting u_s and u_t. Here, the *weight* of a path is the total weight of all of its edges. A path is the shortest path if it has the minimum weight among all paths connecting u_s and u_t.
5. Let $\pi(u_s, u_t)$ be a computed path and $\|\pi(u_s, u_t)\|$ be the length of the path. The path concatenating the edge su_s, path $\pi(u_s, u_t)$, and the edge $u_t t$ is an optimum best-coverage-path with small travelling distance. The best-coverage-distance is $\max(\|su_s\|,\beta,\|u_t t\|)$. Here $\|su_s\|$ and $\|u_t t\|$ are the Euclidean distance between points.

Recently, Li, *et al.* [4] proposed a new method that can construct the local Delaunay triangulation $LDel(S)$ using $O(n \log n)$ communication cost in bits. The communication complexity of the above algorithm is thus $O(n \log n)$.

The correctness of the algorithm is based on the following observation. Consider an edge uv of the best-coverage-path that has the minimum total edge lengths among all best-coverage-paths. It is proved in [3] that if edge uv is not in the localized Delaunay triangulation $LDel(S)$, then there exists a path $\pi(u, v)$ in $LDel(S)$ such that all edges of $\pi(u, v)$ are shorter than uv and the total length of all edges of $\pi(u, v)$ is no more than $\frac{4\sqrt{3}\pi}{9}\|uv\|$. It is obvious that the path $\pi(u, v)$ is in the residue graph G. Consequently, the shortest path in the residue graph G has length no more than $\frac{4\sqrt{3}\pi}{9}$ (which is less than $5/2$) factor of the length of the shortest best-coverage-path in the unit disk graph $UDG(S)$. In other words, here we have a trade-off between the quality performance and the time-complexity. If the graph $UDG(S)$ is used, we get the shortest best-coverage-path but the communication complexity of the algorithm is $O(m \log n)$, where m is the number of edges in $UDG(S)$ which could be as large as $O(n^2)$. On the other hand, our algorithm appropriates the shortest best-coverage-path with total communication cost $O(n \log n)$.

IV. Algorithm Correctness

This Section is devoted to study the correctness of Algorithm 1. Given two points s and t, let $b_{s,t}$ be the smallest radius r such that points s and t are connected inside the region $D(S, r)$. Let $D(s_{i_1}, r)$, $D(s_{i_2}, r)$, $\cdots$, $D(s_{i_k}, r)$ be the sequence of disks centered at sensor nodes travelled by a path connecting s and t. Then, obviously, the following path starting from s, then using the path $s_{i_1} s_{i_2} \cdots s_{i_k}$, and finally ending at t has the same optimum best-coverage-distance (i.e., the radius r) as any optimum best-coverage-path. This implies the following lemma.

Lemma 1: There is an optimum best-coverage-path that uses only the following edges: the edges of the unit disk graph $UDG(S)$, the edges by connecting s to every sensor node, and the edges by connecting t to every sensor node.

We show that it is sufficient to consider only the edges su_s and $u_t t$, where u_s and u_t are the closest sensor nodes to s and t, respectively. Notice that Meguerdichian, *et al.* [1] had already applied this approach. We just give a formal proof here.

Lemma 2: There is an optimum best-coverage-path connecting s to its closest sensor node u_s and t to its closest sensor node u_t.

Proof: Consider an optimum path that does not connect s to its closest sensor node u_s. Assume that s is connected to a node v. We concentrate on the edge sv. We construct an alternative subpath connecting s and v using the edge su_s. Without loss of generality, let $u_0 = u_s, u_1, u_2, \cdots, u_{m-1}, u_m = v$ be the vertices corresponding to the sequence of Voronoi regions traversed by walking from s to v along the segment sv. See Figure 2. If a Voronoi edge or a Voronoi vertex happens to lie on the segment uv, then choose the Voronoi region lying above sv. Assume that line sv is the x-axis. The sequence of vertices u_i, $0 \le i \le m$, defines a path from u_s to v. In general, we refer to the path constructed this way between some nodes u_s and v as the *direct DT path* from u_s to v, denoted by $DT(u_s, v)$, which is also used by [8]. Then we show that the path,

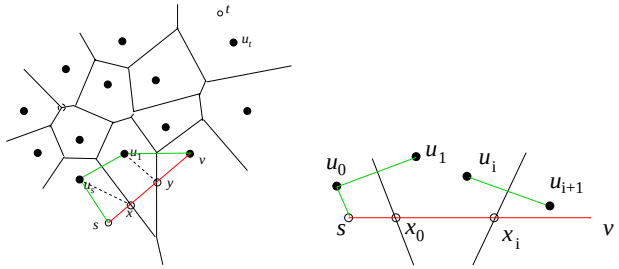

Fig. 2. Left: Connect the starting point s to its closest sensor node u_s. Right: Proof of of the correctness of this connection.

denoted by $DT(s, v)$, consisting of edge su_s and the direct DT path $DT(u_s, v)$ from u_s to v is not worse than the edge sv in terms the coverage-distance. Figure 2 illustrates the proof that follows. Let x_i denote the point on the x-axis that also lies on the boundary between the Voronoi regions $Vor(u_i)$ and $Vor(u_{i+1})$ for $i = 0, 1, \cdots, m-1$. The definition of the Voronoi diagram immediately gives that the circle centered at x_i passing through the vertices u_i and u_{i+1} contains no points of S in its interior. We denote such circle as C_i, i.e., $C_i = disk(x_i, \|x_i u_i\|)$. For each point x_i on the subpath $\pi(s, v)$, its coverage-distance is exactly $\|x_i u_i\|$. Consequently, the coverage-distance of the edge sv is at least (by considering only the point s and all points x_i, $0 \le i < m$) $\max(\|su_0\|, \max_{0 \le i < m}(\|x_i u_i\|))$. Notice that the coverage-distance of any edge $u_i u_{i+1}$, $0 \le i < m$, is at most $\frac{1}{2}\|u_i u_{i+1}\|$; the coverage distance of the edge su_0 is exactly $\|su_0\|$. Consequently, the coverage-distance of the subpath $DT(s, v)$ is at most $\max(\|su_0\|, \max_{0 \le i < m}(\frac{1}{2}\|u_i u_{i+1}\|))$. The definition of the Voronoi region immediately implies that $\|x_i u_i\| \ge \frac{1}{2}\|u_i u_{i+1}\|$. Consequently, the coverage-distance of the subpath $DT(s, v)$ is at most as large as the coverage-distance of the edge sv. Substituting the subpath $\pi(s, v)$ by the subpath $DT(s, v)$ gives an optimum best-coverage-path that connects s to its closest sensor node $u_s = u_0$. ■

For simplicity, from now on, we will not consider the starting point s and the ending point t. Instead, we must only determine the best-coverage-path connecting a pair of sensor nodes. As shown by Lemma 1, the search can be confined to the paths in the unit disk graph $UDG(S)$. However, the unit disk graph $UDG(S)$ may have too many edges, which in the worst case could be as large as $O(n^2)$. We then show that the search space of the best covering problem can be further confined to the Delaunay triangulation $Del(S)$ of the set S of sensors. Notice that the algorithm given in [1] uses this approach without the justification of its correctness. We prove this by the following lemma.

Lemma 3: There is an optimum best-coverage-path that uses only the edges of the Delaunay triangulation $Del(S)$.

Proof: Consider any optimum best-coverage-path connecting two sensor nodes. We show that there is another optimum best-coverage-path such that every edge uv in the path is a Delaunay edge. Remember that an edge uv is a Delaunay edge if and only if the Voronoi regions $Vor(u)$ and $Vor(v)$ share some common Voronoi edge. Consider any edge uv of an optimum best-coverage-path. We show that the direct DT path $DT(u, v)$ has a coverage-distance at most of that of uv. The proof is similar to the proof of Lemma 2. Without loss of generality, let $b_0 = u, b_1, b_2, \cdots, b_{m-1}, b_m = v$ be the vertices corresponding to the sequence of Voronoi regions traversed by walking from u to v along the segment uv. See Figure 3. Let x_i denote the point on the x-axis that also lies on the boundary between the Voronoi regions $Vor(b_{i-1})$ and $Vor(b_i)$ for $i = 1, 2, \cdots, m$. Then $\|x_i b_i\| \geq \frac{1}{2}\|b_i b_{i+1}\|$. Consequently, the coverage-distance of the subpath $DT(u, v)$, which is at most $\max_{1 \leq i \leq m} \frac{1}{2}\|b_{i-1} b_i\|$, is at most as large as the coverage-distance of the edge uv, which is at least $\max_{1 \leq i \leq m} \|x_i b_i\|$. Notice that every edge $b_{i-1} b_i$ of the direct DT path $DT(u, v)$ is a Delaunay edge because the Voronoi regions $Vor(u_i)$ and $Vor(u_{i+1})$ are adjacent. Thus, the lemma follows. ■

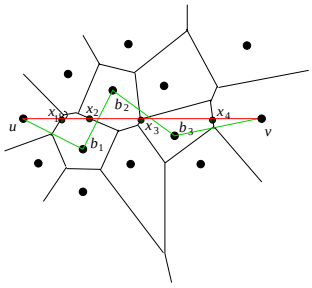

Fig. 3. There is an optimum best-path using only edges of Delaunay triangulation.

We can confine our search space to a much smaller graph $GG(S)$, which can be constructed in a distributed manner efficiently.

Lemma 4: There is an optimum best-coverage-path that uses only the edges of the Gabriel graph $GG(S)$.

Proof: The following Figure 4 illustrates the proof that follows. Similar to the proof of Lemma 3, we know that there is an optimum best-coverage-path connecting any two sensor nodes such that every edge uv of the path only intersects the Voronoi edge shared by $Vor(u)$ and $Vor(v)$. Let p be the midpoint of the segment uv. Then the circle centered at p with radius $\frac{1}{2}\|uv\|$ is empty. It implies that the edge uv is an Gabriel edge. The lemma then follows. ■

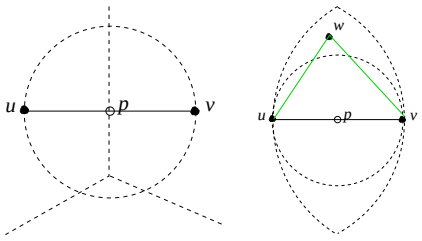

Fig. 4. Left: There is an optimum best-coverage-path from Gabriel graph. Right: There is an optimum best-coverage-path from relative neighborhood graph.

Actually we can further confine our search space based on the following lemma.

Lemma 5: There is an optimum best-coverage-path that uses only the edges of the relative neighborhood graph $RNG(S)$.

Proof: Figure 4 illustrates the proof that follows. Consider an optimum best-coverage-path using edges of the Gabriel graph $GG(S)$. Consider any edge uv of this path. Assume that the $lune(u, v)$ contains a sensor w from S in the interior. Then node w can not be inside the circle $disk(u, v)$ because uv is a Gabriel edge. Thus, the coverage-distance of the midpoint p of edge uv is exactly $\frac{1}{2}\|uv\|$. Notice that the coverage-distance of edge uw is at most[1] $\frac{1}{2}\|uw\|$, and the coverage-distance of edge wv is at most $\frac{1}{2}\|wv\|$. Thus, the coverage-distance of the subpath uwv is at most $\max(\frac{1}{2}\|uw\|, \frac{1}{2}\|wv\|) \leq \frac{1}{2}\|uv\|$. Consequently, substituting the edge uv by the subpath uwv will not increase the coverage-distance of the optimum best-coverage-path. It then follows that there is an optimum best-coverage-path such that it only uses the edges of the relative neighborhood graph $RNG(S)$. ■

Notice that, originally, we thought that the following simpler proof[2] is sufficient to confine the search space to the relative neighborhood graph $RNG(S)$. For every edge uv of an optimum best-coverage-path, if there is a sensor w inside the lune $lune(u, v)$, we can substitute edge uv by the subpath uwv because of $\max(\frac{1}{2}\|uw\|, \frac{1}{2}\|wv\|) \leq \frac{1}{2}\|uv\|$. However, this claim is erroneous because the coverage-distance of edge uv could be much smaller than $\frac{1}{2}\|uv\|$. Thus, $\max(\frac{1}{2}\|uw\|, \frac{1}{2}\|wv\|) \leq \frac{1}{2}\|uv\|$ is not sufficient to show that substituting the edge uv by the subpath uwv will not increase the coverage-distance of the optimum best-coverage-path.

Since there is no sensor node inside the disk using a Gabriel edge uv as diameter, the coverage-distance of the Gabriel edge uv is exactly $\frac{1}{2}\|uv\|$, which is achieved at the midpoint of the edge uv. For an edge uv of the relative neighborhood graph, same reasoning holds.

V. Conclusion

In this paper, we discussed efficient algorithms to find a path with maximum observability under a general assumption of the sensing model. We proved that it is the same as the best coverage problem, which can be solved by an efficient distributed algorithm using the relative neighborhood graph. In addition, we considered some extensions of the best coverage problem: to find a path with the best-coverage-distance while the total energy consumed by this path is minimized among all optimum best-coverage-paths; to find a path with the best-coverage-distance with the total length of edges of this path is no more than 2.5 times the shortest best-coverage-path. We gave efficient distributed algorithms for both extended problems. We also justified the correctness of the algorithm proposed by Meguerdichian, *et al.* [1] using the Delaunay triangulation to solve the best coverage problem. Moreover, we showed that the search space of the best coverage problem can be confined to the relative neighborhood graph, which can be constructed locally. We leave it as an open problem to find a path with the maximum all-sensor observability.

References

[1] Seapahn Meguerdichian, Farinaz Koushanfar, Miodrag Potkonjak, and Mani Srivastava, "Coverage problems in wireless ad-hoc sensor network," in *IEEE INFOCOM'01*, 2001.

[2] Seapahn Meguerdichian, Farinaz Koushanfar, Gang Qu, and Miodrag Potkonjak, "Exposure in wireless ad-hoc sensor network," in *IEEE MOBICOM'01*, 2001.

[3] Xiang-Yang Li, "Localized delaunay triangulation is as good as unit disk graph," 2001, Manuscript.

[4] Xiang-Yang Li, G. Calinescu, and Peng-Jun Wan, "Distributed construction of planar spanner and routing for ad hoc wireless networks," in *IEEE INFOCOM*, 2002, To appear.

[5] Xiang-Yang Li, Peng-Jun Wan, and Yu Wang, "Power efficient and sparse spanner for wireless ad hoc networks," in *IEEE International Conference on Computer Communications and Networks (ICCCN01)*, 2001.

[6] Xiang-Yang Li, Peng-Jun Wan, Yu Wang, and Ophir Frieder, "Sparse power efficient topology for wireless networks," in *HICSS, Hawaii*, 2002, to appear.

[7] T. J. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, MIT Press and MxGraw-Hill, 1990.

[8] D.P. Dobkin, S.J. Friedman, and K.J. Supowit, "Delaunay graphs are almost as good as complete graphs," *Discrete Computational Geometry*, 1990.

[1] It is possible that there is some other sensor node inside the circle $disk(u, w)$. Thus, we can only claim that it is at most $\frac{1}{2}\|uw\|$.

[2] Meguerdichian (private communication) also gave the same erroneous proof.