On Solving Coverage Problems in a Wireless Sensor Network Using Voronoi Diagrams

Anthony Man–Cho So[1] and Yinyu Ye[2]

[1] Department of Computer Science, Stanford University, Stanford, CA 94305, USA.
`manchoso@cs.stanford.edu`
[2] Department of Management Science and Engineering and, by courtesy, Electrical Engineering, Stanford University, Stanford, CA 94305, USA.
`yinyu-ye@stanford.edu`

Abstract. Owing to numerous potential applications, wireless sensor networks have been the focus of a lot of research efforts lately. In this note we study one fundamental issue in such networks, namely the coverage problem, in which we would like to determine whether a region of interest is sufficiently covered by a given set of sensors. This problem is motivated by monitoring applications in sensor networks, as well as robustness concerns and protocol requirements. We show that the coverage problem and some of its variants can be treated in a unified manner using suitable generalizations of the Voronoi diagram. As a result, we are able to give algorithms that have better runtimes than those proposed in previous works (see, e.g., [5, 6]). Our approach also yields efficient algorithms for coverage problems where the sensing region of a sensor is an ellipse or an L_p–ball, where $p \geq 1$.

1 Introduction

1.1 Background

Due to recent advances in wireless communication and embedded micro–sensing MEMS technologies, inexpensive wireless nodes capable of sensing, storing, processing, and communicating data are becoming increasingly common and readily available. Such devices provide the means for building large–scale wireless sensor networks for various applications, such as distributed data storage [11], target tracking [12], and habitat monitoring [13], just to name a few. In these applications, sensors collect information about their surroundings, and the data so obtained are then aggregated to give a complete picture of the region of interest. Hence, a fundamental issue in the design of wireless sensor networks is coverage, i.e. how well are the sensors covering the region of interest. This issue has been tackled by various researchers in the sensor networks community. Typically, a sensor s is modelled as a point p_s in space, and its sensing region is modelled as an Euclidean ball centered around p_s. From this simple model, many different coverage problems have been proposed. For example, the authors of [10] considered the problems of finding the *maximal breach path* and the *maximal support*

path in a network, which informally can be viewed as the paths that are least and best monitored in the network, respectively. The maximal support path problem is further investigated by the authors of [9] in which energy consumptions of the sensors are also being taken into account. The authors of [14, 15], also motivated by energy consumption considerations, studied the problem of finding a subset of sensors $\mathscr{C}$ so that the region of interest S is covered by those sensors (i.e. each point in S is within the sensing range of some sensor in $\mathscr{C}$). This problem is later generalized by the authors of [16], in which they considered finding a subset of sensors $\mathscr{C}$ of *minimal cardinality* such that (i) the communication graph formed by the sensors in $\mathscr{C}$ is connected, and (ii) the region of interest S is k–covered by those sensors (i.e. each point in S is within the sensing ranges of k distinct sensors in $\mathscr{C}$). They proposed an algorithm that yields a subset whose cardinality is at most $O(\log n)$ times larger than the optimal and has the required properties.

In this note, we consider the following two coverage problems: (i) given a set of sensors $\mathscr{C}$ with their sensing regions and a region of interest S, determine whether every point in S is k–covered by the sensors in $\mathscr{C}$, where $k \geq 1$ is a given constant; (ii) given a set of sensors $\mathscr{C}$ with their sensing regions and a region of interest S, determine the largest k such that S is k–covered by the sensors in $\mathscr{C}$. These two problems are motivated by robustness concerns as well as protocol requirements. For example, triangulation–based localization protocols require at least three sensors to localize an object, and hence we would like every point in S to be at least 3–covered. Regarding the first problem, Huang and co–authors [5, 6] presented an $O(nm \log m)$ (resp. $O(nm^2 \log m)$) algorithm for determining whether a region in $\mathbb{R}^2$ (resp. $\mathbb{R}^3$) is k–covered by the given n sensors, where m is the maximum number of sensing regions that can intersect the sensing region of a particular sensor. Since m can be of order $\Theta(n)$ (see Figure 1), their algorithms have worst–case runtimes of $O(n^2 \log n)$ (for the 2–d case) and $O(n^3 \log n)$ (for the 3–d case), respectively. Also, it is curious that the runtimes of both algorithms are independent of k. Thus, it is natural to ask whether there exists an algorithm that can exploit the tradeoff between runtime and the parameter k. Regarding the second problem, a naïve approach would be to run the above algorithm $k(\leq n)$ times, resulting in a worst–case $O(n^3 \log n)$ (resp. $O(n^4 \log n)$) algorithm for the 2–d case (resp. 3–d case). However, to the best of our knowledge, this problem has not been addressed directly in the sensor networks community.

1.2   Our Contribution

We propose to treat the aforementioned coverage problems in a unified manner using suitable generalizations of the Voronoi diagram. This provides a rigorous mathematical framework for tackling those problems and enables us to design algorithms that have much more favorable worst–case runtimes than those proposed in previous works. Specifically, we show that for the 2–d case, if the sensors have identical sensing ranges, then problem (i) can be solved in $O(n \log n + nk^2)$ time by computing the so–called *k–th order Voronoi diagram* [4, 8]. Thus, our algorithm is independent of the configuration of the sensors and depends only on
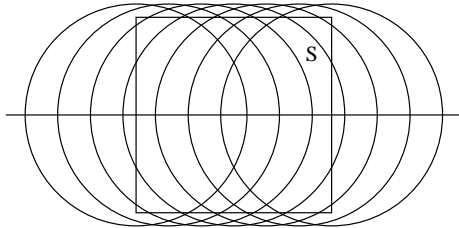
Fig. 1. A scenario where $m = \Theta(n)$: the $n + 1$ circles are centered at $(1 - 2i/n, 0)$, where $i = 0, 1, \ldots, n$, and each of them has radius $1 + \epsilon$ for some $\epsilon > 0$.

the number of sensors, n, and the parameter k. Moreover, for $k = o(\sqrt{n \log n})$, our algorithm has a better runtime than that proposed in [5]. In the case where the sensing ranges are different, we are able to obtain an $O(n \log n)$ algorithm to determine whether a region $S \subset \mathbb{R}^2$ is 1–covered. Our framework also allows us to study coverage problems in which the sensing region is not a ball. In particular, we give efficient algorithms for determining whether a region $S \subset \mathbb{R}^2$ is 1–covered when the sensing regions are modelled as ellipses or L_p–balls, where $p \geq 1$. Both of these algorithms are faster than the straightforward adaptation of the techniques presented in [5]. For problem (ii), we obtain an $O(n^3)$ (resp. $O(n^4)$) algorithm for the 2–d case (resp. for the 3–d case) by using a well–known relationship between power diagrams and arrangements [2]. This is again faster than a direct application of the algorithms described in [5, 6].

2 Problem Statement

We formulate the aforementioned coverage problems as follows. For simplicity of discussion, let the region of interest S be the cube $\{x \in \mathbb{R}^d : \|x\|_\infty \leq 1\}$. Let $a_i \in \mathbb{R}^d$ be the position of the i–th sensor, and let $B_i \equiv B(a_i, r_i) = \{x \in \mathbb{R}^d : \|x - a_i\|^2 \leq r_i^2\}$ be the sensing region of sensor i, where $1 \leq i \leq n$. In other words, the sensing region of sensor i is a ball centered at a_i with radius r_i. In the sequel, we shall mainly concern ourselves with the cases $d = 2$ and 3. For any given integer $k \geq 1$, we say that $x \in S$ is k–*covered* if there exist indices $1 \leq i_1 < i_2 < \cdots < i_k \leq n$ such that $x \in \cap_{j=1}^k B_{i_j}$. We would then like to answer the following queries in an efficient manner:

Query 1: (k–COVERAGE; k–COV) Is every point in S k–covered by the balls $B_1, \ldots, B_n$? In other words, is it true that for every $x \in S$, there exist indices $1 \leq i_1 < \cdots < i_k \leq n$ such that $x \in \cap_{j=1}^k B_{i_j}$?

Query 2: (MAX–k–COVERAGE; MAX–k–COV) Determine the largest k such that every point $x \in S$ is k–covered by the balls $B_1, \ldots, B_n$.

3 The Proposed Solutions

In the above problems, we are required to certify that every point in S is k–covered by the given balls. Thus, in order to obtain efficient algorithms for both problems, we could attempt to find a finite and small set of points in S such that the queries can be answered by examining only these points. Indeed, our techniques are motivated by this idea, and they depend on the notion of *power* defined as follows. Given the ball $B(a, r)$, we define the *power* of a point $x \in \mathbb{R}^d$ with respect to $B(a, r)$ by $\mathrm{pow}(x, B(a, r)) = \|x - a\|^2 - r^2$. Note that:

$$\mathrm{pow}(x, B(a, r)) \begin{cases} < 0 \text{ if } x \in \mathrm{int}(B(a, r)) \\ = 0 \text{ if } x \in \partial B(a, r) \\ > 0 \text{ if } x \in \mathbb{R}^d \backslash B(a, r) \end{cases}$$

As we shall see, the notion of power allows us to treat the aforementioned problems in a unified manner.

3.1 The k–Cov Problem in $\mathbb{R}^2$ with Identical Disks

To begin, let us consider a version of the k–Cov problem where $d = 2$ and $r_i \equiv r$ for all $i = 1, 2, \ldots, n$. In other words, we are interested in the two–dimensional case of the k–Cov problem, with the sensing ranges of all sensors being identical. Let $B_1 \equiv B(a_1, r), \ldots, B_n \equiv B(a_n, r)$ be the given balls, and let $\mathscr{C} = \{a_1, \ldots, a_n\}$ be the set of centers. For any subset $U \subset \{1, 2, \ldots, n\}$ with $|U| = k$, define:

$$\mathrm{cell}(U) = \left\{ x \in \mathbb{R}^2 : \mathrm{pow}(x, B_u) < \mathrm{pow}(x, B_v) \ \forall u \in U, \ v \in \{1, 2, \ldots, n\} \backslash U \right\}$$

$$= \left\{ x \in \mathbb{R}^2 : \|x - a_u\|^2 < \|x - a_v\|^2 \ \forall u \in U, \ v \in \{1, 2, \ldots, n\} \backslash U \right\}$$

The set $\mathrm{cell}(U)$ is the so–called *Voronoi region* of U. Intuitively, $\mathrm{cell}(U)$ is the set of points in $\mathbb{R}^2$ closer to all points in U than to any point in $\mathscr{C} \backslash U$. It is well–known (see, e.g., [4, 8]) that the Voronoi regions of all subsets of cardinality k induce a subdivision of $\mathbb{R}^2$, called the k–*th order Voronoi diagram* of $\mathscr{C}$ and denoted by $V_k(\mathscr{C})$. Moreover, such a subdivision is polyhedral. Now, observe that for $x \in \mathrm{cell}(U) \cap S$, we have x being k–covered iff $\mathrm{pow}(x, B_u) < 0$ for all $u \in U$. This suggests the following algorithm for determining whether S is k–covered or not. First, we compute the k–th order Voronoi diagram $V_k(\mathscr{C})$ of the point set $\mathscr{C}$. Then, as we will demonstrate below, it suffices to check whether the following points are k–covered: (i) the vertices of $V_k(\mathscr{C})$ in S, (ii) the intersections between the sides of S and $V_k(\mathscr{C})$, and (iii) the corners of S. We say that $x \in S$ is a *critical point* if it belongs to one of the above three categories. Thus, in order to correctly answer the query, it suffices to check for the critical points. We summarize our algorithm below (see Algorithm 1).

Algorithm 1 Testing k–Coverage by Identical Disks in $\mathbb{R}^2$

Given: A set of n disks $B_1 \equiv B(a_1, r), \ldots, B_n \equiv B(a_n, r)$ in $\mathbb{R}^2$, and an integer $k \geq 1$.

1: Compute the k–th order Voronoi diagram $V_k(\mathscr{C})$ for the set of points $\mathscr{C} = \{a_1, \ldots, a_n\}$.
2: **for each** vertex v of $V_k(\mathscr{C})$ that lies within S **do**
3:     Let $\{a_{i_1}(v), \ldots, a_{i_k}(v)\}$ be the k closest neighbors of v in $\mathscr{C}$.
4:     **if** $\exists j$ such that $\text{pow}(v, a_{i_j}(v)) > 0$ **then**
5:       Return No
6:     **end if**
7: **end for**
8: **for each** side l of the square S **do**
9:     Let $\{p_j\}_{j=1}^M$ be the union of the two endpoints of l and the intersections between l and $V_k(\mathscr{C})$.
10:     **for each** $j = 1, 2, \ldots, M$ **do**
11:       Let $a_{i_1}(p_j), \ldots, a_{i_k}(p_j)$ be the k closest neighbors of p_j in $\mathscr{C}$.
12:       **if** $\exists t$ such that $\text{pow}(p_j, a_{i_t}(p_j)) > 0$ **then**
13:         Return No
14:       **end if**
15:     **end for**
16: **end for**
17: Return Yes

Theorem 1 *Algorithm 1 correctly answers the query.*

Proof. For each $x \in S$, let $a(x)$ be its k–th closest neighbor in $\mathscr{C}$. Clearly, we have $\text{pow}(x, a(x)) > 0$ if x is not k–covered. Let $\hat{x} \in S$ be such that $\text{pow}(\hat{x}, a(\hat{x}))$ is maximized. Note that such an $\hat{x}$ exists since S is compact and the power distance to the k–th closest point in $\mathscr{C}$ is a continuous function. We now consider two cases:

Case 1: $\hat{x} \in \text{int}(S)$
We claim that $\hat{x}$ is a vertex of $V_k(\mathscr{C})$. Suppose that this is not the case. Then, we have the following possibilities:

Case 1.1: $\hat{x} \in \text{cell}(U)$ for some $U \subset \{1, 2, \ldots, n\}$ such that $|U| = k$. Then, the gradient of $\text{pow}(\cdot, a(\hat{x}))$ at $\hat{x}$ is $d(\hat{x}) = 2(\hat{x} - a(\hat{x}))$. In particular, an ascent direction is given by $d(\hat{x})$ (if $\hat{x} = a(\hat{x})$, then all directions are ascent directions). Now, let $x' = \hat{x} + \alpha d(\hat{x})$, where $\alpha > 0$ is sufficiently small so that $x' \in \text{cell}(U)$. Then, we have $\|x' - a(\hat{x})\|^2 > \|\hat{x} - a(\hat{x})\|^2$. Moreover, since every $x \in \text{cell}(U)$ has the same k closest neighbors in $\mathscr{C}$, we have $\|x' - a(x')\|^2 \geq \|x' - a(\hat{x})\|^2$. This shows that $\hat{x}$ is not a maximizer, which is a contradiction.

Case 1.2: $\hat{x}$ lies on the interior of an edge e in $V_k(\mathscr{C})$ defined by the intersection of the closure of two cells, say $e \equiv \text{int}\left(\overline{\text{cell}(U)} \cap \overline{\text{cell}(V)}\right)$, where $|U| = |V| = k$. Let a_i, a_j be two distinct k–th closest neighbors of $\hat{x}$ in $\mathscr{C}$ (cf. [8]). Note that the equation of the line L_e containing e is given by $\|x - a_i\|^2 = \|x - a_j\|^2$, or equivalently,

$$L_e : 2(a_j - a_i)^T x = \|a_j\|^2 - \|a_i\|^2$$

Define $u = (a_j - a_i)/\|a_j - a_i\|$, and let $v \neq 0$ be such that $u^T v = 0$. Note that v is a vector along the direction of L_e. Now, consider the point $x' = \hat{x} + \alpha v$ for some $\alpha \in \mathbb{R}$ to be determined shortly. Note that for sufficiently small $|\alpha| > 0$, we have $x' \in e$. Moreover, we compute:

$$\|x' - a_j\|^2 = \|\hat{x} - a_j\|^2 + \alpha^2 \|v\|^2 + 2\alpha v^T (\hat{x} - a_j) \tag{1}$$

Thus, if $\alpha \neq 0$ is chosen with a suitable sign with $|\alpha|$ small enough, then we have $2\alpha v^T (\hat{x} - a_j) \geq 0$ and $x' \in e$. This implies that $\|x' - a_j\|^2 > \|\hat{x} - a_j\|^2$. Moreover, x' has the same k closest neighbors as $\hat{x}$ in $\mathscr{C}$ (since $x' \in e$), and for all points q on the edge e, we have $\|q - a_i\|^2 = \|q - a_j\|^2 \leq \|q - a_k\|^2$ for all $k \notin U \cup V$. This contradicts the definition of $\hat{x}$.

Case 2: $\hat{x} \in \mathrm{bd}(S)$
Let l be the side of S such that $\hat{x} \in l$. Suppose that $\hat{x} \in \mathrm{int}(l) \cap \mathrm{cell}(U)$ for some U. We claim that $\hat{x}$ is not a maximizer. To see this, let $v \neq 0$ be a vector along the direction of l. Now, consider the point $x' = \hat{x} + \alpha v$ for some $\alpha \in \mathbb{R}$ to be determined shortly. As before, we have $x' \in \mathrm{int}(l) \cap \mathrm{cell}(U)$ if $|\alpha| > 0$ is chosen to be sufficiently small. Moreover, using (1), we see that α can be chosen such that $\|x' - a(\hat{x})\|^2 > \|\hat{x} - a(\hat{x})\|^2$ (recall that $a(\hat{x})$ is the k–th closest neighbor of $\hat{x}$ in $\mathscr{C}$). Since every $x \in \mathrm{cell}(U)$ has the same k closest neighbors in $\mathscr{C}$, we have $\|x' - a(x')\|^2 \geq \|x' - a(\hat{x})\|^2$. This again contradicts the definition of $\hat{x}$.

Theorem 2 *Algorithm 1 runs in time $O(n \log n + nk^2)$.*

Proof. The k–th order Voronoi diagram $V_k(\mathscr{C})$ of the n–point set $\mathscr{C}$ can be computed in $O(n \log n + nk^2)$ time using the algorithm described in [1]. As is shown in [8], the total number of vertices, edges and cells in $V_k(\mathscr{C})$ is bounded by $O(k(n-k))$. Thus, the remaining steps can be done in $O(k^2(n-k))$ time. This completes the proof.

The algorithm above possesses some interesting features that deserve further discussion. First, observe that we have reduced the coverage problem to an optimization problem, in which we are trying to find an $x \in S$ such that the distance to its k–th closest neighbor in $\mathscr{C}$ is maximized. Such a viewpoint enables us to answer the coverage query by examining only a set of critical points, and the efficiency of our algorithm comes from the facts that (i) the k–th order Voronoi diagram can be computed efficiently, and (ii) there exists a small set of critical points. As we shall see in subsequent sections, these ideas can be used to obtain efficient algorithms for other coverage problems. Secondly, note that the same analysis would go through if the region S is, say, a convex polygon with a constant number of sides. All we need is that the number of candidate maxima on the boundary of S is small. Thirdly, we observe that the k–th order Voronoi diagram depends only on the locations of the centers $a_1, \ldots, a_n$ but not on the common sensing range r. Thus, it is not necessary to re–compute $V_k(\mathscr{C})$ every time we change the value of r. It suffices to check the critical points described in the algorithm, and this requires only $O(nk^2)$ time. This opens up the possibility of a binary search strategy to determine the smallest r such that S is k–covered.

3.2 The 1–Cov Problem with Various Sensing Regions

In most previous works, the sensing region of a sensor is modelled as an Euclidean ball. In this section, we show how our techniques can be extended to yield efficient algorithms for the 1–Cov problem with various sensing regions. As before, we shall assume that S is the cube $\{x \in \mathbb{R}^d : \|x\|_\infty \leq 1\}$.

Covering with Non–Uniform Disks Suppose that we are given a set of balls $B_1 \equiv B(a_1, r_1), \ldots, B_n \equiv B(a_n, r_n)$, and we would like to determine whether S is covered by these balls. To answer this query, we first compute the *power diagram* of the collection $\mathscr{B} = \{B_1, \ldots, B_n\}$ [2], which is a collection of cells of the form:
$$\text{cell}(i) = \left\{ x \in \mathbb{R}^2 : \text{pow}(x, B_i) < \text{pow}(x, B_j) \ \forall j \neq i \right\}$$

where $\text{pow}(x, B_i) = \|x - a_i\|^2 - r_i^2$ for $x \in \mathbb{R}^2$ and $i = 1, 2, \ldots, n$. Clearly, a point $x \in S$ is not covered iff $\min_{1 \leq i \leq n} \text{pow}(x, B_i) > 0$. Thus, we see that S is not covered iff the optimal value of the optimization problem

$$
\begin{aligned}
\text{maximize} \quad & \min_{1 \leq i \leq n} \text{pow}(x, B_i) \\
\text{subject to} \quad & x_j^2 \leq 1 \qquad\qquad j = 1, 2
\end{aligned}
\tag{2}
$$

is positive. Using the arguments in the proof of Theorem 1, it can be shown that the candidate maxima of (2) are precisely the critical points introduced before, i.e. (i) the vertices of the power diagram $P(\mathscr{B})$, (ii) the intersections between the sides of S and $P(\mathscr{B})$, and (iii) the corners of S. Thus, to determine the optimal value of (2), we can proceed as in Algorithm 1. Since the power diagram $P(\mathscr{B})$ of the collection $\mathscr{B}$ can be computed in $O(n \log n)$ time and the total number of vertices, edges and cells in $P(\mathscr{B})$ is bounded by $O(n)$ [2], we obtain the following theorem:

Theorem 3 *The* 1–Cov *problem in* $\mathbb{R}^2$ *can be solved in* $O(n \log n)$ *time.*

We remark that the above approach works in the three–dimensional case as well. Specifically, by following the arguments in the proof of Theorem 1, one can show that it is enough to check whether the following points are covered: (i) the vertices of the power diagram $P(\mathscr{B})$ that lie in S, (ii) the 0–flats that arise from the intersections between the sides of S and $P(\mathscr{B})$, and (iii) the corners of S. Since the power diagram $P(\mathscr{B})$ in $\mathbb{R}^3$ can be computed in $O(n^2)$ time and the total number of features (i.e. vertices, edges, ...) in $P(\mathscr{B})$ is bounded by $O(n^2)$ [2], we have the following theorem:

Theorem 4 *The* 1–Cov *problem in* $\mathbb{R}^3$ *can be solved in* $O(n^2)$ *time.*

Note that in both cases, our algorithms have much better worst–case runtimes than those proposed in [5, 6].

Covering with Ellipses Suppose that we are given a collection of ellipses $E_1, \ldots, E_n$, where E_i is the ellipse defined by $(x - a_i)^T Q_i (x - a_i) \leq r_i^2$, with Q_i symmetric positive definite, and we would like to answer the query with the B_i's replaced by the E_i's. As before, we may define $\mathrm{pow}(x, E_i) = (x - a_i)^T Q_i (x - a_i) - r_i^2$ for $x \in \mathbb{R}^2$. To answer the query, we first compute the "power diagram" of the collection $\{E_1, \ldots, E_n\}$. Although this "power diagram" is no longer polyhedral, we can still show that the optimum must occur at a critical point as defined before. First, by using a similar argument as before, it is easy to see that $\hat{x}$ cannot lie in the interior of a cell. Now, the intersection σ of the closure of two cells is a portion of a quadratic curve $\mathcal{C}_\sigma$, whose equation is given by:

$$\mathcal{C}_\sigma : (x - a_i)^T Q_i (x - a_i) - r_i^2 = (x - a_j)^T Q_j (x - a_j) - r_j^2$$

Let $\hat{x} \in \mathrm{int}(\sigma)$, and consider the two vectors $v_1 = Q_i(\hat{x} - a_i)/\|Q_i(\hat{x} - a_i)\|$ and $v_2 = Q_j(\hat{x} - a_j)/\|Q_j(\hat{x} - a_j)\|$. Note that v_i is the gradient vector of the function $\mathrm{pow}(\cdot, E_i)$ at $\hat{x}$, where $i = 1, 2$. Thus, if the angle spanned by v_1 and v_2 is strictly less than π, then we can increase the objective value by moving along the curve $\mathcal{C}_\sigma$ in the direction that lies within the cone spanned by v_1 and v_2, which is a contradiction. On the other hand, if the angle spanned by v_1 and v_2 is exactly π, then we can find an $v \neq 0$ such that $v^T v_1 = v^T v_2 = 0$. Now, for any $\alpha \neq 0$, we have:

$$(\hat{x} + \alpha v - a_i)^T Q_i (\hat{x} + \alpha v - a_i) - r_i^2$$
$$= (\hat{x} - a_i)^T Q_i (\hat{x} - a_i) - r_i^2 + 2\alpha v^T Q_i (\hat{x} - a_i) + \alpha^2 v^T Q_i v$$
$$= (\hat{x} - a_i)^T Q_i (\hat{x} - a_i) - r_i^2 + \alpha^2 v^T Q_i v$$

and similarly,

$$(\hat{x} + \alpha v - a_j)^T Q_j (\hat{x} + \alpha v - a_j) - r_j^2 = (\hat{x} - a_j)^T Q_j (\hat{x} - a_j) - r_j^2 + \alpha^2 v^T Q_j v$$

Since Q_i and Q_j are positive definite, we see that $(\hat{x} + \alpha v - a_i)^T Q_i (\hat{x} + \alpha v - a_i) > (\hat{x} - a_i)^T Q_i (\hat{x} - a_i)$ and $(\hat{x} + \alpha v - a_j)^T Q_j (\hat{x} + \alpha v - a_j) > (\hat{x} - a_j)^T Q_j (\hat{x} - a_j)$ for all $\alpha \neq 0$. It follows that by moving along $\mathcal{C}_\sigma$, we can increase the objective value, which again contradicts the definition of $\hat{x}$. Finally, the case where $\hat{x} \in \mathrm{bd}(S)$ can be handled as before.

The "power diagram" for ellipses can have complexity $\Theta(n^{2+\epsilon})$ for any $\epsilon > 0$, and it can be computed in $O(n^{2+\epsilon})$ time as well [3]. Thus, we have the following theorem:

Theorem 5 *The* 1–COV *problem in* $\mathbb{R}^2$ *with ellipsoidal sensing regions can be solved in* $O(n^{2+\epsilon})$ *time, for any* $\epsilon > 0$.

Covering with L_p–Balls Let $p \geq 1$ and $r > 0$ be fixed, and suppose now that we are given a collection of L_p–balls $B_1, \ldots, B_n$, where B_i is defined by $\|x - a_i\|_p^p \leq r^p$ (i.e. all the B_i's have the same radius). As before, we would like to answer the query of whether S is covered by the B_i's. For this purpose,

we define $\mathrm{pow}(x, B_i) = \|x - a_i\|_p^p - r^p$ for $x \in \mathbb{R}^2$ and formulate an optimization problem similar to (2). The optimal value and the optimal solution can then be determined by first computing the L_p–Voronoi diagram of the point set $\{a_1, \ldots, a_n\}$ (see [7]) and then checking the objective values at the corresponding critical points. We omit the details and summarize the results as follows:

Theorem 6 *The* 1–Cov *problem in* $\mathbb{R}^2$ *with identical* L_p *sensing regions can be solved in* $O(n \log n)$ *time.*

3.3 The Max–k–Cov Problem

We now turn our attention to the Max–k–Cov problem, which is to determine the largest k such that S is k–covered by the given balls $B_1, \ldots, B_n$ (here, we consider the Euclidean metric). To do this, we first use the algorithm in [2] to compute all the k–th order power diagrams (where $1 \le k \le n - 1$). This takes $O(n^3)$ time for the 2–d case and $O(n^4)$ time for the 3–d case. Then, we can check the critical points in each of these diagrams as before and determine the largest k such that S is k–covered by the balls $B_1, \ldots, B_n$. Since the total number of features in all the k–th order power diagrams is bounded by $O(n^3)$ for the 2–d case and by $O(n^4)$ for the 3–d case [2], we obtain the following theorem:

Theorem 7 *The* Max–k–Cov *problem in* $\mathbb{R}^2$ *(resp.* $\mathbb{R}^3$*) can be solved in* $O(n^3)$ *(resp.* $O(n^4)$*) time.*

Notice that this is more efficient than a direct application of the algorithms in [5, 6].

4 Conclusion

In this note we have proposed to use suitable generalizations of the Voronoi diagram to treat various coverage problems that arise from the design of wireless sensor networks. We have shown that in many cases, the runtimes of our algorithms are better than those proposed in earlier works. Moreover, our approach allows us to handle sensing regions whose shapes cannot be conveniently modelled as Euclidean balls. We remark, however, that the proposed algorithms are not distributed in nature. Since it is desirable to have decentralized computations in a wireless sensor network, an interesting future direction would be to find efficient distributed algorithms for the coverage problems discussed in this note.

References

1. Alok Aggarwal, Leonidas J. Guibas, James Saxe, Peter W. Shor, *A Linear–Time Algorithm for Computing the Voronoi Diagram of a Convex Polygon*, Discrete and Computational Geometry 4:591–604, 1989.

2. F. Aurenhammer, *Power Diagrams: Properties, Algorithms and Applications*, SIAM Journal of Computing 16(1):78–96.

3. Pankaj K. Agarwal, Otfried Schwarzkopf, Micha Sharir, *The Overlay of Lower Envelopes in Three Dimensions and Its Applications*, Proceedings of the 11th Annual Symposium on Computational Geometry, 1995.

4. Bernard Chazelle, Herbert Edelsbrunner, *An Improved Algorithm for Constructing k–th Order Voronoi Diagrams*, IEEE Transactions on Computers C–36(11):1349–1354, 1987.

5. Chi–Fu Huang, Yu–Chee Tseng, *The Coverage Problem in a Wireless Sensor Network*, Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications (WSNA), pp. 115–121, 2003.

6. Chi–Fu Huang, Yu–Chee Tseng, Li–Chu Lo, *The Coverage Problem in Three–Dimensional Wireless Sensor Networks*, IEEE GLOBECOM, 2004.

7. D. T. Lee, *Two–Dimensional Voronoi Diagram in the L_p–Metric*, Journal of the ACM 27(4):604–618, 1980.

8. Der–Tsai Lee, *On k–Nearest Neighbor Voronoi Diagrams in the Plane*, IEEE Transactions on Computers C–31(6):478–487, 1982.

9. Xiang–Yang Li, Peng–Jun Wan, Ophir Frieder, *Coverage in Wireless Ad–Hoc Sensor Networks*, Proceedings of the IEEE International Conference on Communications (ICC), 2002.

10. Seapahn Meguerdichian, Farinaz Koushanfar, Miodrag Potkonjak, Mani B. Srivastava, *Coverage Problems in Wireless Ad–Hoc Sensor Networks*, IEEE INFOCOM, 2001.

11. Sylvia Ratnasamy, Brad Karp, Scott Shenker, Deborah Estrin, Ramesh Govindan, Li Yin, Fang Yu, *Data–Centric Storage in Sensornets with GHT, a Geographic Hash Table*, Mobile Networks and Applications 8(4):427–442, 2003.

12. Jaewon Shin, Anthony Man–Cho So, Leonidas Guibas, *Supporting Group Communication among Interacting Agents in Wireless Sensor Networks*, Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), 2005.

13. Robert Szewczyk, Eric Osterweil, Joseph Polastre, Michael Hamilton, Alan Mainwaring, Deborah Estrin, *Habitat Monitoring with Sensor Networks*, Communications of the ACM 47(6):34–40, 2004.

14. Di Tian, Nicolas D. Georganas, *A Coverage–Preserving Node Scheduling Scheme for Large Wireless Sensor Networks*, Proceedings of the 1st ACM International Conference on Wireless Sensor Networks and Applications (WSNA), pp. 32–41, 2002.

15. Fan Ye, Gary Zhong, Jesse Cheng, Songwu Lu, Lixia Zhang, *PEAS: A Robust Energy Conserving Protocol for Long–Lived Sensor Networks*, Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS), 2003.

16. Zongheng Zhou, Samir Das, Himanshu Gupta, *Connected K–Coverage Problem in Sensor Networks*, Proceedings of the 13th International Conference on Computer Communications and Networks (ICCCN), 2004.