Minimum Cost Sensor Coverage of Planar Regions

Xiaochun Xu and Sartaj Sahni
Computer and Information Science and Engineering Department
University of Florida
Gainesville, FL 32611
Email: {xxu, sahni}@cise.ufl.edu

Nageswara S. V. Rao
Computer Science and Mathematics Division
Oak Ridge National Laboratory
Oak Ridge, TN 37831
Email: raons@ornl.gov

***Abstract*—We consider the placement of sensors with circular sensing regions for q-coverage of planar regions. We first consider the placement of sensors of multiple types and costs over a specified set of locations to minimize the total sensors' cost. We present two approximate solutions to this problem with multiplicative factors of 3 and $1 + 1/l$ of the optimal cost, where l is a tunable parameter. We then present a method to transform a region coverage instance into an equivalent point coverage instance and show a relationship between the cost of the optimal coverage of the two instances. This transformation enables us to use better studied approximation algorithms for point coverage to derive good sensor deployments for region coverage.**

Keywords: Minimum cost sensor deployment, region coverage, approximation algorithms.

I. Introduction

There has been a rapid proliferation of sensor network deployments in a variety of areas such as environmental and habitat monitoring, industrial sensing and diagnostics, critical infrastructure monitoring and protection, and situational awareness of battlefield environments. It is possible in some cases to randomly scatter sensor nodes to monitor a region, particularly when the sensor nodes are inexpensive. There are cases, however, where the deployment costs and domain-specific considerations such as population density, would require a more strategic approach to sensor placement. In general, the objectives of sensor placement could be quite varied due to the considerations of geometric coverage, network connectivity, probability of detection, affects on population, precision of localization and others. One of the basic formulations deals with placing sensors with limited range to monitor a region. We consider the particular problem of placing sensors with circular sensing regions to ensure that each point inside a planar region is covered by at least q sensors. This formulation captures the essence of detection by sensor networks wherein detection is assured even if $q - 1$ sensors fail.

Among the geometric coverage problems addressed in the sensor networks area, there are two basic formulations, namely, point-set coverage and region coverage. In the former, the objective is to ensure that a set of discrete points lie within a suitable union of the sensing regions of the sensors. For the region coverage, the objective is to ensure that the entire specified region is contained within the union of sensing regions. In the sensor networks area, more results have been reported on the point-set coverage than region coverage problems. Despite apparent similarities these two classes of problems often require significantly different approaches [4]. Also, region coverage problems are closely related to tiling and packing problems, and yet require subtly different approaches, often leading to computationally hard problems [3].

In this paper, we consider the *minimum cost deployment* problem: given multiple sensor types with different sensing ranges and costs, the objective is to q-cover the region at minimum cost by placing sensors only at a subset of the specified locations. We present two approximation methods with multiplicative factors of 3 and $1+1/l$, where l is a tunable parameter that determines the computational complexity. Since checking for q-coverage of a region is quite a bit harder than for a collection of points, we show how to transform a region coverage problem into an equivalent point coverage problem so that known point-coverage algorithms may be used to construct good region coverage deployments.

The organization of this paper is as follows. In Section II we briefly describe works related to our formulation. We present our approximation algorithms for minimum cost deployment problem in Section III. In Section IV we develop a transformation from region coverage to point coverage. We present computational results in Section V.

II. Related Work

The literature on sensor coverage problems is quite extensive due to the vast variability in sensor characteristics, regions to be monitored, sensor placement locations, and objectives of sensor deployment. See [8], [13] for a review of sensor coverage problems. Our minimum cost deployment problem is closely related to the circle covering problem studied by Kershner [10] in 1939. The following results from [10] are relevant to our work.

Theorem 1: [10] Let r be the sensing range and let A be the area of the planar region that is to be 1-covered. The minimum number, $N(r)$ of sensors needed for 1-coverage satisfies:

$$\lim_{r \to 0} \pi r^2 N(r) = (2\pi\sqrt{3}/9)A \tag{1}$$

As noted in [10], the constant $(2\pi\sqrt{3}/9) = 1.209$ gives the "unavoidable overlapping" that must occur in every sensor deployment that achieves 1-coverage.

Theorem 2: [10] For a rectangular region with area A and perimeter P,

$$(2\pi\sqrt{3}/9)(A-2\pi r^2) < \pi r^2 N(r) < (2\pi\sqrt{3}/9)(A+2Pr+16r^2) \quad (2)$$

A heuristic method for covering a convex region using circles has been proposed by Das et al [5], [4], wherein the centers of the circles can be arbitrarily located. Guo and Qu [6] claim optimal algorithms for 1-coverage of rectangles and planar regions. However, neither their rectangle coverage nor their planar region coverage algorithm actually 1-covers with the fewest number of circles. For example, their rectangle covering algorithm uses 6 circles to cover a square with side length $\sqrt{2}r$ when 1 circle suffices.

III. Minimum Cost Deployment

Consider that we have a sufficient supply of sensors of t different types. Let $r_i > 0$ and $c_i > 0$, respectively, be the sensing range and cost of a sensor of type i. In the *unrestricted minimum cost deployment* (UMCD) problem, we wish to q-cover a specified planar region R at minimum cost. This problem is NP-hard.

Theorem 3: UMCD is NP-hard even when the region R is a straight line.

Proof: Consider an instance I of the (simplified) integer knapsack problem which is known to be NP-hard [12]. I is described by integer item sizes $s_1, \cdots, s_n$ and a positive integer B. The objective is to determine whether or not there exist non-negative integers a_i such that $\sum a_i s_i = B$.

From I, we construct the UMCD instance I' in which we have n sensor types. The cost and range of a sensor of type i is s_i and the length of the line R that is to be covered is B. It is easy to see that the line R can be 1-covered at a cost $\leq B$ iff there is a solution to the restricted integer knapsack instance I. Hence, UMCD is NP-hard. ∎

Without loss of generality, in the following, we assume that no two sensor types have the same range or the same cost and that $r_i < r_j$ iff $c_i < c_j$. Further, we assume that we are given a set S of sites where it is feasible to place a sensor. R denotes the region in Euclidean space that is to be monitored. We are interested in deploying sensors, at most 1 sensor at each location of S, so as to provide q-coverage, $q \geq 1$, for region R at minimum cost. We may refer to this problem as the *restricted* MCD problem. We propose two algorithms. The first is a 3-approximation algorithm and the second is an approximation scheme.

A. 3-Approximation

Let $tMax = \text{argmax}_{1\leq i\leq t}\{r_i\}$, $rMax = r_{tMax}$, and $L = 2rMax + \epsilon$, where ϵ is a positive constant. Our 3-approximation algorithm, 3-approx (Figure 1), begins by tiling R with regular hexagons whose sides have length L; some of the hexagons that overlap the boundary of R may contain portions of the Euclidean space that are not in R. Next, for each hexagon, H_i, of the tiling, we find an optimal (i.e., least cost) sensor deployment that q-covers $H_i \cap R$. Finally, the optimal sensor deployments for the hexagons in the tiling are combined by ensuring that no site in S has two or more sensors. To the algorithm of Figure 1, we may add an optional pruning step in which redundant sensors (i.e., sensors whose omission doesn't affect the q-coverage property) are eliminated.

Step 1: Tile R with regular hexagons whose side length is $L = 2rMax + \epsilon$.

Step 2: For each hexagon H_i of the tiling find an optimal deployment of sensors to sites in S so as to q-cover $H_i \cap R$.

Step 3: Combine the optimal deployments found in Step 2 for all of the hexagons in the tiling. In case a site of S has two or more sensors assigned to it, discard all but the sensor with maximum range.

Fig. 1. Algorithm 3-approx

It is easy to see that 3-approx constructs a q-cover for R with at most one sensor per site in S provided such a q-cover exists. In the following, we establish that the cost of the constructed q-cover is at most 3 times that of an optimal q-cover and we analyze the complexity of 3-approx.

Lemma 1: A sensor at a site $s \in S$ can cover points in at most three of the hexagons of the tiling of Step 1 of 3-approx.

Proof: We consider only the case when s is located within one of the hexagons (say H_1) of the tiling (see Figure 2). The proof for the case when s is in no H_i (this may happen when S has locations outside of R) is similar. Let $d(p_1, p_2)$ be the Euclidean distance between two points p_1 and p_2 and let $d(H_i, H_j)$ be the smallest distance between two points one of which is in H_i and the other is in H_j. Since, $d(p_1, s) \geq L > rMax$ for points outside of the 7 hexagons shown in Figure 2, the sensor at s cannot cover points outside of the 7 hexagons. Notice that $d(H_i, H_j) = L$ for $i, j \in \{2, 4, 6\}$ as well as for $i, j \in \{3, 5, 7\}$. From this observation, $L = 2rMax + \epsilon > 2rMax$, and the triangle inequality, it follows that for any point p_1 in H_i and p_2 in H_j, $i, j \in \{2, 4, 6\}$ or $i, j \in \{3, 5, 7\}$, $d(p_1, s) + d(p_2, s) > 2rMax$. Hence, either $d(p_1, s) > rMax$ or $d(p_2, s) > rMax$ or both. So, the sensor at s can cover points in at most one of H_2, H_4, and H_6 and at most one of H_3, H_5, and H_7. This sensor may cover also points in H_1. Hence, points in at most 3 of the hexagons of the tiling may be covered. ∎

Theorem 4: 3-approx is a 3-approximation algorithm.

Proof: Consider any instance of the q-cover problem for which there is a feasible solution (i.e., a selection of sensors, at most one per site in S, that q-covers R). Start with an optimal solution O for this instance. Let $cost(O)$ be the cost of sensors deployed in O. To each hexagon H_i in the tiling of Step 1, assign the subset of sensors of O that cover points in H_i. Let $cost(H_i)$ be the cost of the sensors assigned to H_i. From Lemma 1, it follows that the assignment of sensors to hexagons, assigns each sensor of O to at most 3 hexagons. Hence, $\sum cost(H_i) \leq 3cost(O)$. Since the sensors assigned to H_i q-cover $H_i \cap R$, their cost must be at least that of the deployment computed in Step 2. So, the sensor deployment

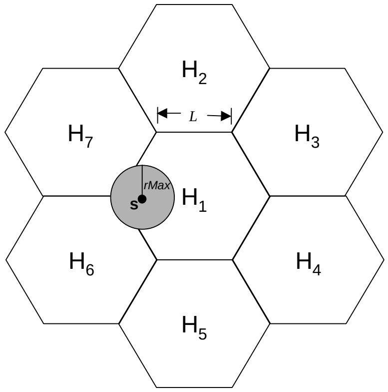

Fig. 2. Sensor at $s \in S$ can cover points in at most 3 hexagons

following Step 3 has a cost that is at most $3cost(O)$. ■

The complexity of algorithm 3-approx is governed by the time to determine the optimal q-cover for each of the hexagons in the tiling. When computing the optimal q-cover for a hexagon H, we need consider only those sites in S that line in the shaded region shown in Figure 3. The area A of this region is $area(H) + perimeter(H) * rMax + \pi rMax^2$. = $1.5\sqrt{3}L^2 + 6L * rMax + \pi rMax^2$. Under the assumption that the site density (i.e., number of sites in any region of area A) is bounded (this is the case, for instance, when there is a fixed lower bound on the distance between two sites) by fixed constant and the number of sensor types t is bounded, the size of the state space for each H_i is bounded by a (potentially very large) constant. The optimal deployment for each H_i may, therefore, be found in constant time by simply searching the state space of H_i. Under these assumptions, the complexity of 3-approx is linear in the number of hexagons in the tiling.

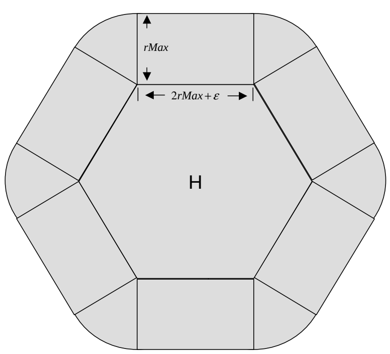

Fig. 3. Location of sites that can cover points in H

B. An Approximation Scheme

We employ the shifting strategy of [1], [7] to arrive at an approximation scheme for the q-cover problem. Let $l \geq 1$ be an integer shifting parameter. The cost of the sensor deployments computed by our approximation scheme will be within a multiplicative factor of $1 + 1/l$ of the cost of an optimal deployment. By making l suitably large, we can obtain deployments as close to optimal as desired. Figure 4 gives our approximation scheme AS.

Step 1: Let U be the smallest bounding rectangle for the region R. Let V_i be a tile whose height equals that of U and whose width is $i * L$, $1 \leq i \leq l$. Let T_i be the tiling of U in which the first tile is V_i. This tile is followed by zero or more tiles of type V_l. An additional tile is used at the right end (if necessary) to cover the remainder of U. The width of this last tile is $< l * L$ and is chosen so as to cover the remaining uncovered width of U.

Step 2: Do Steps 3 and 4 for $1 \leq i \leq l$.

Step 3: For each tile T in the tiling T_i, find an optimal deployment of sensors to sites in S so as to q-cover $T \cap R$.

Step 4: Combine the optimal deployments found in Step 3 for the tiles in T_i. In case a site of S has two or more sensors assigned to it, discard all but the sensor with maximum range.

Step 5: From the constructed deployments for $T_1, \cdots, T_l$, select the deployment that has least cost.

Fig. 4. Algorithm AS

Unlike algorithm 3-approx, which considers a single tiling of the region R that is to be q-covered, algorithm AS considers a family, $T_1, \cdots, T_l$, of tilings. To obtain a tiling in this family, we begin by determining the smallest bounding rectangle U of R. Next, this bounding rectangle is tiled using tiles whose height equals that of U but whose width is $l * L$, where $L = 2rMax + \epsilon$ and ϵ is a positive constant. The first and last tiles in the tiling are exceptions. In T_i, the width of the first tile is $i * L$. In case tiling with a tile of width $i * L$ followed by tiles of width $l * L$ doesn't exactly cover U, a last tile whose width is less than $l * L$ is used in T_i.

In Steps 2 through 4, we compute an optimal deployment for each of the tilings T_i using a strategy similar to that used for the hexagonal tiling used in 3-approx (i.e., find an optimal deployment to q-cover the sub-region of R included in each tile of T_i and then combine these optimal deployments to obtain a q-cover for R). Finally, in Step 5, the best of the q-covers over all l tilings $T_1, \cdots, T_l$ is selected as the deployment to use.

As in the case of algorithm 3-approx, we may add an optional pruning step in which redundant sensors (i.e., sensors whose omission doesn't affect the q-coverage property) are eliminated.

It is easy to see that AS finds a q-cover for every instance for which there is a q-cover. The following theorem establishes that AS is, in fact, an approximation scheme for the q-cover problem.

Theorem 5: Algorithm AS is an approximation scheme for the q-cover problem. Specifically, the computed q-cover has a cost that is within a multiplicative factor $(1 + 1/l)$ of the cost of an optimal q-cover.

Proof: Consider any instance of the q-cover problem for which there is a feasible solution. Let O be an optimal solution for this instance, let $cost(O)$ be the cost of sensors deployed in O, and let $cost(T_i)$ be the cost of the sensor deployment computed in Step 4 of AS for the tiling T_i. Since each tile in T_i, except possibly the first and last, has a width $l * L > 2rMax$, no sensor can cover points in 3 or more consecutive tiles. Let O_i be the sensors deployed in O that cover points in 2 tiles of T_i. Using a distribution scheme similar to that used in the proof of Theorem 4, we obtain

$$cost(T_i) \leq cost(O) + cost(O_i)$$

Suppose that a sensor of O that lies in the first tile of T_i covers a point in the second tile of T_i. Since $L > 2rMax$, this sensor is part of the second tile of T_j, $j < i$ and does not cover a point in any tile of T_j that is not part of this second tile. For $j > i$, this sensor remains in the first tile for T_j and is unable to cover points that are not in the first tile of T_j. Hence, this sensor, which is in O_i, is not in any O_j, $j \neq i$. By reasoning in a similar fashion, we may show that all O_is are disjoint and so $\sum_{i=1}^{l} cost(O_i) \leq cost(O)$. Hence,

$$\begin{aligned} &\min_{1\leq i\leq l}\{cost(T_i)\} \\ &\leq \frac{1}{l}\sum_{i=1}^{l} cost(T_i) \leq \frac{1}{l}\sum_{i=1}^{l}(cost(O) + cost(O_i)) \\ &= cost(O) + \frac{1}{l}\sum_{i=1}^{l} cost(O_i) \leq (1 + \frac{1}{l})cost(O) \end{aligned}$$

■

Under assumptions similar to those made in the analysis of 3-approx, the complexity of AS is linear in the product of l and the number of rectangles in a tiling T_i.

IV. Region Coverage Via Point Coverage

Known algorithms [2], [9], [14], [17] to verify q-coverage of a region are rather cumbersome, while algorithms for point coverage are rather straightforward. Thus, we are motivated to transform a region coverage instance into an equivalent point coverage instance. We assume a single sensor type with sensing range r, and that at most 1 sensor may be placed at each location in S. The objective is to find the minimum cost deployment or equivalently minimum number of sensors for q-coverage of the region R. Such a sensor deployment, denoted by $OPT_R(r)$, is called an *optimal deployment* and its cost is $cost(OPT_R(r))$.

The transformation from region coverage to point coverage may be accomplished by superimposing a grid of points over R such that every point of R is within a distance $\frac{d}{\sqrt{2}}$ of at least one grid point that is inside R. Here, $d < \sqrt{2}r$ is the distance between adjacent grid points and is an optimization parameter. Let $G(d)$ denote those grid points that are in (or on the boundary of) R. Note that locations in S may not be points of $G(d)$ and may not even be in R. Figure 5 shows $G(d)$ for the case when R is a rectangle. $G(d)$ is composed of the small circles (both shaded and unshaded) in this figure; the outermost grid points are a distance $d/2$ from the boundary of R. Let $r' = r - \frac{d}{\sqrt{2}} > 0$.

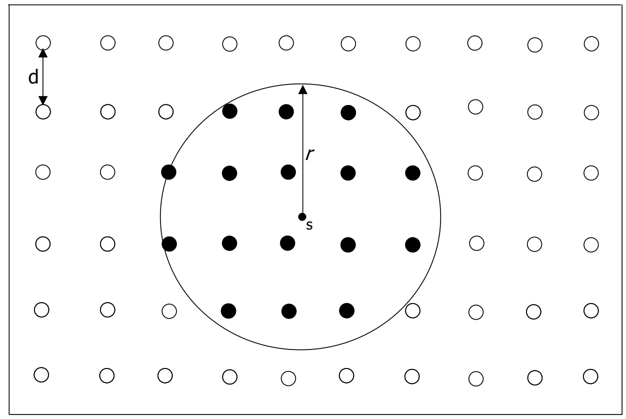

Fig. 5. $G(d)$ for a rectangular region R. The 16 shaded grid points are within a disk of radius r centered at the location s.

Lemma 2: Let $D(r')$ be a deployment of sensors whose range is r'. If $D(r')$ q-covers $G(d)$, then $D(r' + \frac{d}{\sqrt{2}})$ q-covers the region R.

Proof: Consider any point P in the region R. Let $P' \in G(d)$ be the grid point closest to P. From the definition of $G(d)$, it follows that $d(P, P') \leq \frac{d}{\sqrt{2}}$. Since $D(r')$ q-covers $G(d)$, there are at least q sensors in $D(r')$ that are within a distance r' of P'. From the triangular inequality, it follows that these q sensors are within a distance $r' + d(P, P') \leq r$ of P. Hence, $D(r)$ q-covers P. ■

Corollary 1: Let $OPT_G(d, r')$ be an optimal deployment of sensors, whose range is r', that q-covers $G(d)$. $cost(OPT_G(d, r')) \geq cost(OPT_R(r))$.

Note that when $G(d)$ cannot be q-covered using sensors whose range is r', $cost(OPT_G(d)) = \infty$. Note also that sensors may be deployed only to locations in S.

Lemma 3: Consider an infinite grid of points with point separation d. Let $MAX(r)$ and $MIN(r)$, respectively, be the maximum and minimum number of grid points covered by a sensor whose range is r. $MAX(r) \leq \pi(\frac{r}{d})^2 + 2\lfloor 2\frac{r}{d}\rfloor + 1$ and $MIN(r) \geq \pi(\frac{r}{d})^2 - 2\frac{r}{d} - \lfloor 2\frac{r}{d}\rfloor$.

Proof: Without loss of generality, assume that the sensor is located at $(0, 0)$ and that P_i, $i = 1, 2, 3, 4$ are the 4 grid points closest to the sensor (Figure 6). We may further assume that $P_1 = (x * d, y * d)$ is closer to the sensor than are P_i, $i = 2, 3, 4$. Clearly, $0 \leq x \leq \frac{1}{2}$ and $0 \leq y \leq \frac{1}{2}$. Label the row in which P_1 lies row 0, the row right above (below) P_1 is row 1 (-1), and so on. Let $maxrow$ be the row number of the highest row that is partially covered by the sensor and let $minrow$ be corresponding lowest row. Note that $\lfloor 2\frac{r}{d}\rfloor \leq maxrow - minrow + 1 \leq \lfloor 2\frac{r}{d}\rfloor + 1$.

Let $row(i)$ be the number of grid points in row i that are covered by the sensor. Clearly, $\lfloor 2\sqrt{(\frac{r}{d})^2 - (y_1 + i)^2}\rfloor \leq row(i) \leq \lfloor 2\sqrt{(\frac{r}{d})^2 - (y_1 + i)^2}\rfloor + 1$.

$$\begin{aligned} MAX(r) &= \sum_{i=minrow}^{maxrow} row(i) \\ &\leq \sum_{i=minrow}^{maxrow} (\lfloor 2\sqrt{(\frac{r}{d})^2 - (y_1 + i)^2}\rfloor + 1) \end{aligned}$$

$$
\begin{aligned}
&\leq \sum_{i=minrow}^{maxrow} (\lfloor 2\sqrt{(\frac{r}{d})^2-(y_1+i)^2}\rfloor) + \lfloor 2\frac{r}{d}\rfloor + 1 \\
&\leq \sum_{i=minrow}^{-1} (2\sqrt{(\frac{r}{d})^2-(y_1+i)^2}) + \lfloor 2\frac{r}{d}\rfloor \\
&\quad + \sum_{i=1}^{maxrow} (2\sqrt{(\frac{r}{d})^2-(y_1+i)^2}) + \lfloor 2\frac{r}{d}\rfloor + 1 \\
&\leq 2\int_{-\frac{r}{d}}^{\frac{r}{d}} \sqrt{(\frac{r}{d})^2-y^2 dy} + 2\lfloor 2\frac{r}{d}\rfloor + 1 \\
&\leq \pi(\frac{r}{d})^2 + 2\lfloor 2\frac{r}{d}\rfloor + 1
\end{aligned}
$$

$$
\begin{aligned}
MIN(r) &= \sum_{i=minrow}^{maxrow} row(i) \\
&\geq \sum_{i=minrow}^{maxrow} \lfloor 2\sqrt{(\frac{r}{d})^2-(y_1+i)^2}\rfloor \\
&> \sum_{i=minrow}^{maxrow} (2\sqrt{(\frac{r}{d})^2-(y_1+i)^2} - 1) \\
&\geq \sum_{i=minrow}^{maxrow} (2\sqrt{(\frac{r}{d})^2-(y_1+i)^2}) - \lfloor 2\frac{r}{d}\rfloor \\
&\geq \pi(\frac{r}{d})^2 - 2\frac{r}{d} - \lfloor 2\frac{r}{d}\rfloor
\end{aligned}
$$

■

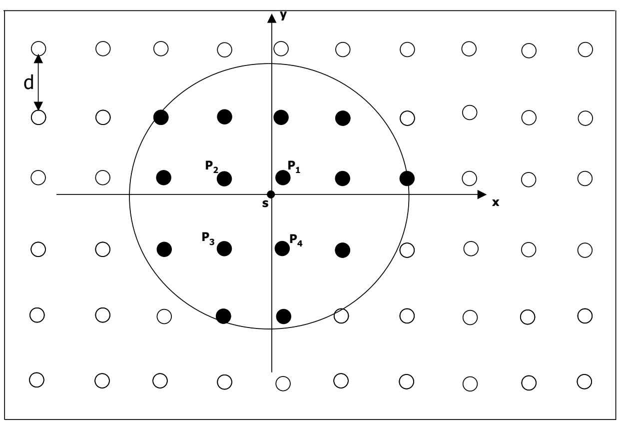

Fig. 6. The sensor s is located inside the square formed by four grid points P_i, $i = 1, 2, 3, 4$, and covers $MAX(r)$ grid points, shown in dark color. W.l.o.g, assume the grid point $P_1 = (x_1, y_1)$ is closer to s' than is P_i, $i = 2, 3, 4$.

Lemma 4: When $D(r')$ as in Lemma 2 exists, $cost(OPT_G(d, r')) \leq F * cost(OPT_R(r))$, where $F \leq MAX(r) - MIN(r') + 1$.

Proof: The optimal deployment $OPT_R(r)$ q-covers every point in R and hence q-covers every point in $G(d)$. When the sensing range of the sensors in the deployment $OPT_R(r)$ is reduced to r', this deployment may no longer q-cover every point in $G(d)$. For each sensor s in $OPT_R(r)$, the difference between the number of grid points covered by s before and after the sensing range reduction is at most $MAX(r) - MIN(r')$. Note that even though $MAX(r)$ and $MIN(r')$ are defined with respect to infinite grids, the bound $MAX(r) - MIN(r')$ applies to $G(d)$, which is a finite grid. This is so because the stated bound may overestimate (but not underestimate) the difference. Since the deployment $D(r')$ exists, by placing a sensor whose range is r' at at most $MAX(r) - MIN(r')$ points of S at which there is no sensor, we can cover all points of $G(d)$ covered by s when its sensing range was r. Repeating this process for each $s \in OPT_R(r)$, we obtain a deployment that has at most $F * size(OPT_R(r))$ sensors with sensing range r' that q-covers $G(d)$. ■

Lemma 5: $F \leq 12.443\frac{r}{d} - 2.399$.

Proof:

$$
\begin{aligned}
&MAX(r) - MIN(r') \\
&\leq \pi(\frac{r}{d})^2 + 2\lfloor 2\frac{r}{d}\rfloor + 1 \\
&\quad -(\pi(\frac{r-\frac{d}{\sqrt{2}}}{d})^2 - 2\frac{r-\frac{d}{\sqrt{2}}}{d} - \lfloor 2\frac{r-\frac{d}{\sqrt{2}}}{d}\rfloor) \\
&\leq \pi(\frac{r}{d})^2 + 2\lfloor 2\frac{r}{d}\rfloor + 1 - (\pi(\frac{r}{d} - \frac{1}{\sqrt{2}})^2 \\
&\quad -2(\frac{r}{d} - \frac{1}{\sqrt{2}}) - \lfloor \frac{2r}{d} - \sqrt{2}\rfloor) \\
&\leq \pi(\frac{r}{d})^2 + 4\frac{r}{d} + 1 - \pi(\frac{r}{d} - \frac{1}{\sqrt{2}})^2 \\
&\quad +2(\frac{r}{d} - \frac{1}{\sqrt{2}}) + 2\frac{r}{d} - \sqrt{2} \\
&\leq (8 + \sqrt{2}\pi)\frac{r}{d} + 1 - 2\sqrt{2} - \frac{\pi}{2} \\
&\leq 12.443\frac{r}{d} - 3.399
\end{aligned}
$$

Hence, $F \leq MAX(r) - MIN(r') + 1 \leq 12.443\frac{r}{d} - 2.399$. ■

Theorem 6: Let R be a region for which the deployment $D(r - \frac{d}{\sqrt{2}})$ exists. Then, $cost(OPT_R(r)) \leq cost(OPT_G(d, r - \frac{d}{\sqrt{2}})) \leq (12.443\frac{r}{d} - 2.399)cost(OPT_R(r))$.

Proof: Follows from Corollary 1 and Lemma 5. ■

Notice that the use of a finer grid (i.e., smaller d) results in a larger approximation factor. This would argue in favor of using the maximum possible d; that is $d = \sqrt{2}r$. For this value of d, the approximation factor is

$$(12.443\frac{r}{d} - 2.399) = 12.443/\sqrt{2} - 2.399 = 6.4$$

Of course, while the approximation factor is minimized when $d = \sqrt{2}r$, it is entirely possible, and intuitively, we expect so, that on some instances, better deployments are obtained using a smaller d. Our somewhat nonintuitive result that the approximation factor is smaller when d is larger is an artifact of the proof method used to establish the approximation factor. Further, it may be necessary to use a smaller d to ensure the existence of the deployment $D(r - \frac{d}{\sqrt{2}})$.

V. Experimental Results

In Section III. we presented several strategies to q-cover a region when the sensor locations are limited to a specified set S. We believe that all but the region coverage via point

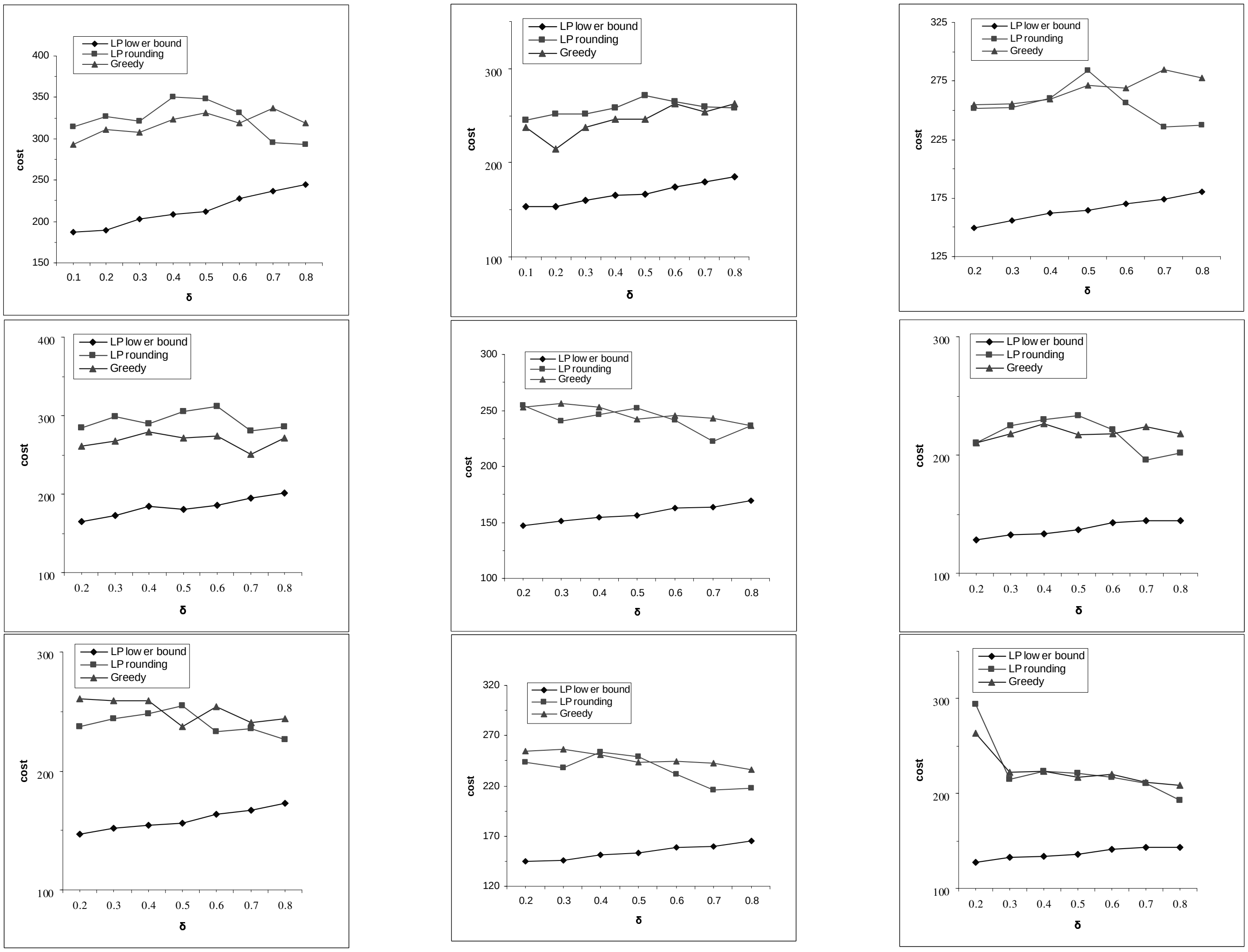

Fig. 7. Total sensor cost required v.s. various δs, where q=1.

coverage method of Section IV are too compute intensive to be used on large deployment instances. This is because each of the other methods requires us to find optimal deployments for several small hexagons or tiles. While an optimal deployment for a hexagon or tile may be found in $O(1)$ time, the required constant time is expected to be large as the optimal solution is found by searching a potentially large though constant size space of possible solutions. This search will perform many explicit tests for region rather than point coverage are time consuming and region coverage tests are compute intensive. As a result, our experiments focus on the method of Section IV adapted to instances with multiple sensor types.

We programmed the region coverage via point coverage scheme of Section IV on a Dell Dimension PC with a 2.13 GHz dual-core processor and 2GB memory. To cover the resulting grid $G(d)$, we used an ILP (integer linear programming) relaxation method as well the greedy algorithm of Xu and Sahni [16]. For the ILP relaxation, we started with the ILP formulated in [16] for point coverage, relaxed it to a linear program, solved the linear program using *lp-solver* 5.0 [11], and then converted the solution to the linear program to a feasible integer solution to the original ILP using the rounding method proposed by Wang and Zhang [15]. Since both the ILP formulation and greedy method of [16] are for heterogeneous sensor deployment, we experimented with heterogeneous instances with up to 4 sensor types.

For test data, we used a 35×35 region R. To construct $G(d)$ for any given d, we tiled R with $d \times d$ squares and considered the centers of these tiles. Centers that were outside R were relocated to their nearest point on the boundary of R. The resulting center locations define $G(d)$. To construct the set S of permissible sensor locations, we tiled R with $7r_{max}/12 \times 7r_{max}/12$ squares and included the center of each square in S, where r_{max} is the maximum range of a sensor. Next, S was augmented by adding $\lceil 3area(R)/(\pi r_{max}^2) \rceil$ randomly chosen points from R. For each value of q, $1 \leq q \leq 3$ (q is the desired coverage degree), we experimented with the following 9 test cases. The sensor set T specifies the sensor types as pairs (r, c), where r is the sensor range and c is the cost of the sensor.

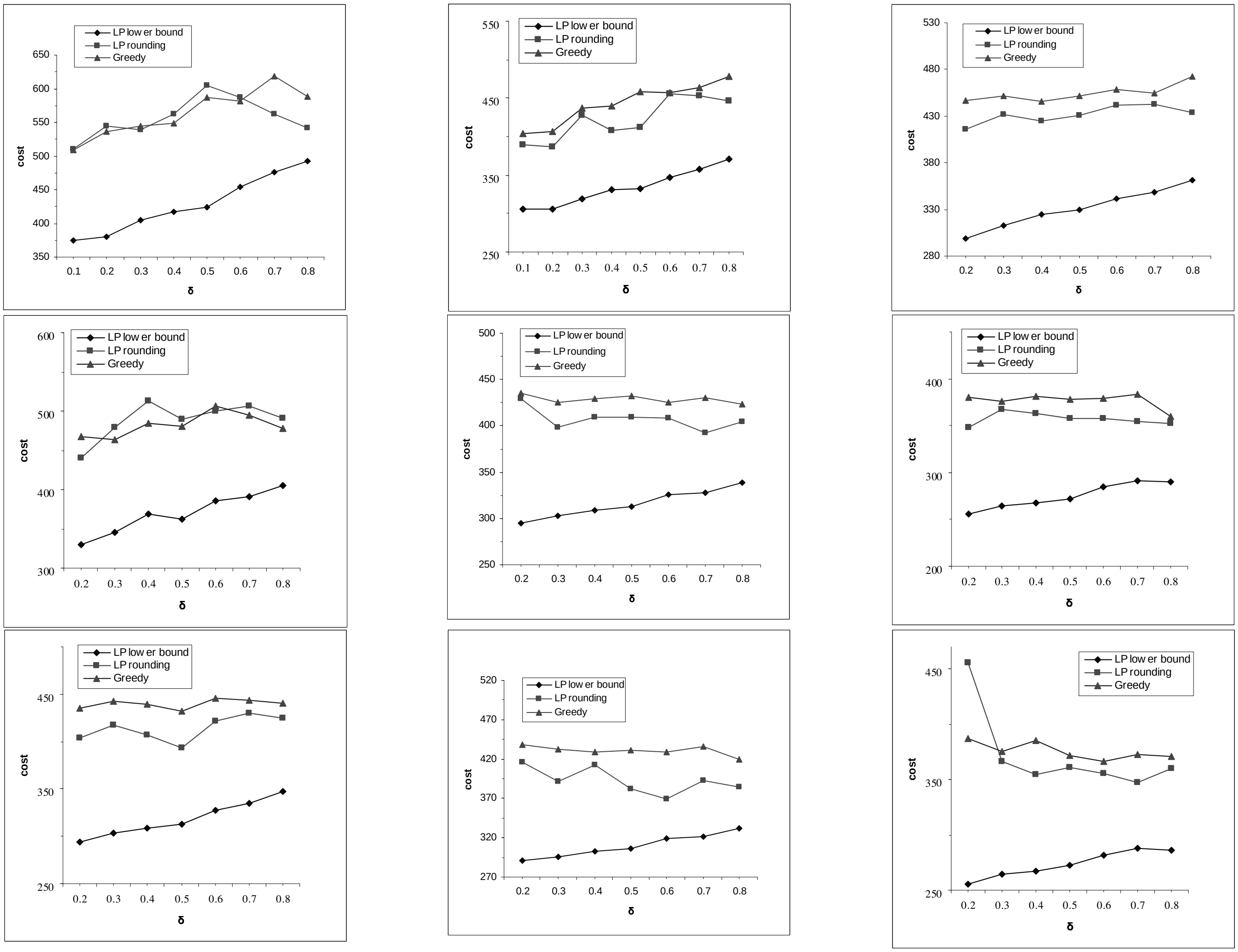

Fig. 8. Total sensor cost required v.s. various δs, where q=2.

1. $|S|$ = 362 T = {(4,5)}; 2. $|S|$ = 242 T = {(5,6)}
3. $|S|$ = 242 T = {(3.5,4),(5,6)}
4. $|S|$ = 313 T = {(3,4),(4.5,5.5)}
5. $|S|$ = 242 T = {(3,3.5),(4,4),(5,6)}
6. $|S|$ = 207 T = {(3.5,4),(4.5,5),(5.5,6)}
7. $|S|$ = 242 T = {(3.5,4),(4.5,5),(5,6)}
8. $|S|$ = 242 T = {(3,3.5),(4,4),(4.5,5),(5,6)}
9. $|S|$ = 207 T = {(3,3),(3.5,4),(4,5),(5.5,6)}

For each test case and each q, we experimented with different d values and for each combination of test case, d, and q, we generated 10 random instances (these differed only in the set S of allowable locations). Figures 7-9 plot the average of the cost of the deployments obtained for these 10 random instances using the ILP and greedy methods. The figures show also the lower bound for the cost of an optimal deployment for $G(d)$ as determined by the solution to the linear programming relaxation for the ILP. Note that when determining a deployment to q-cover $G(d)$, sensor ranges were reduced from r as specified in the test case to r'. Our experiments show that the relaxed ILP method generally produces lower cost deployments than does the greedy method. Further, the cost of the deployment is not well correlated with d. Often, though, a smaller d resulted in a smaller deployment cost. The use of a smaller d does, however, increase the time needed to find a deployment. Each of the instances used by us was solved in a few seconds when d was large and in a little under 15 minutes when d was small.

VI. Conclusions

We presented two approximate solutions to the problem of minimizing the cost of placing sensors with multiple costs at certain locations to ensure q-coverage of a planar region. We then developed a transformation from region coverage to point coverage. Several natural extensions of the problems studied here are possible. It would be interesting to incorporate costs associated with the region such as population within sensor regions in addition to the sensor costs. It would also be interesting to consider non-circular and probabilistically specified sensor regions.

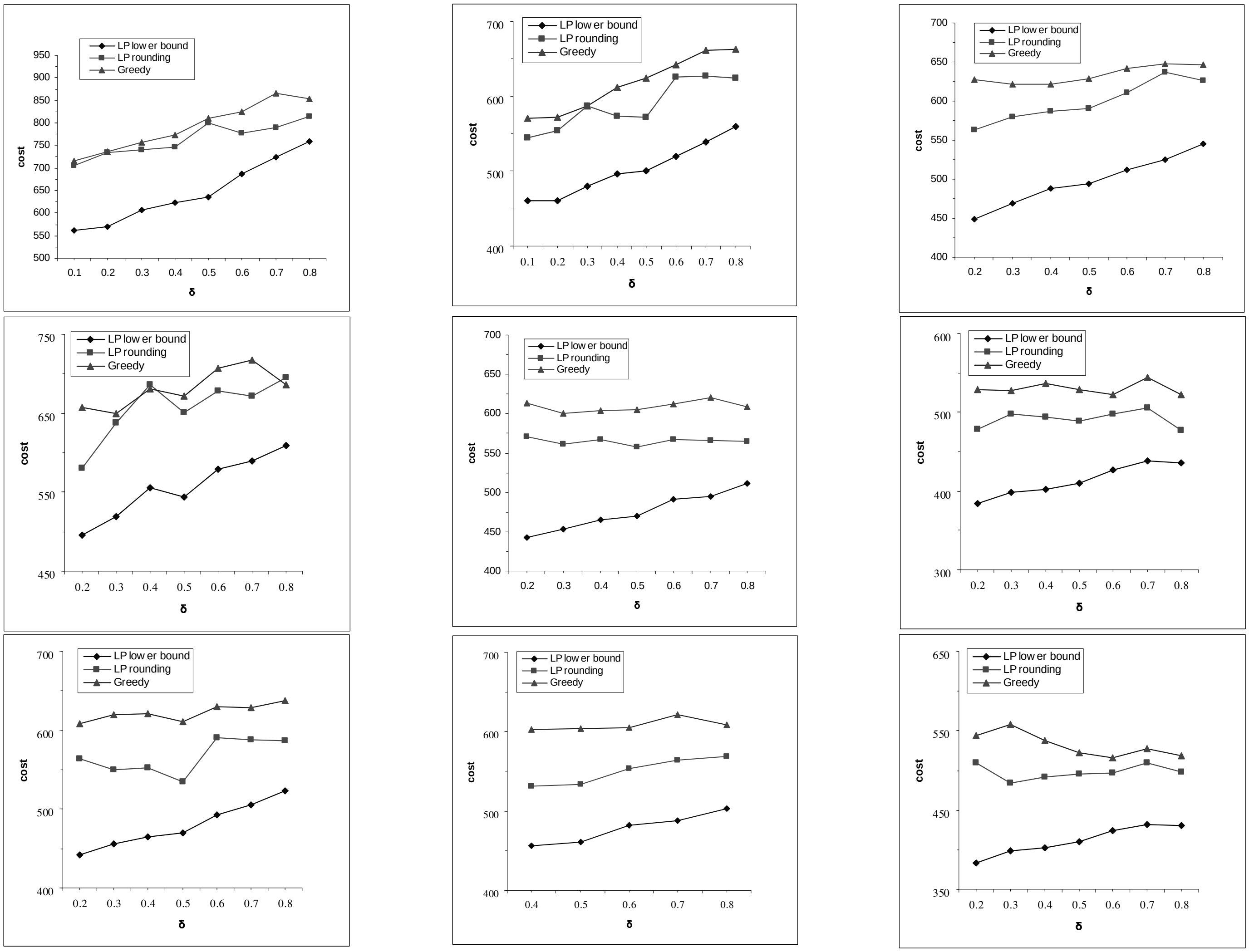

Fig. 9. Total sensor cost required v.s. various δs, where q=3.

ACKNOWLEDGMENTS

This work is funded by SensorNet program at Oak Ridge National Laboratory managed by UT-Battelle, LLC for U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

REFERENCES

[1] B. S. Baker. Approximation algorithms for np-complete problems in planar graphs. *Journal of the ACM*, 41(1):153–180, 1994.

[2] B. Carbunar, A. Grama, J. Vitek, and O. Carbunar. Redundancy and coverage detection in sensor networks. *ACM Trans. on Sensor Networks*, 2(1):94–128, 2006.

[3] J. C. Culberson and R. A. Reckhow. Covering polygons is hard. In *Proc. of Foundations on Computer Science*, 1988.

[4] G. K. Das. Placement and range assignment in power-aware radio networks, 2007. PhD Dissertation, Indian Statistical Institute.

[5] G. K. Das, S. Das, S. C. Nandy, and B. P. Sinha. Efficient algorithm for placing a given number of base stations to cover a convex region. *Journal of Parallel and Distributed Computing*, 66:1353–1358, 2006.

[6] Y. Guo and Z Qu. Coverage control for a mobile robot patrolling a dynamic and uncertain environement. In *Proceedings of World Congress on Intelligent Control*, pages 4899–4903. 2004.

[7] D. S. Hochbaum and W. Maass. Approximation schemes for covering and packing problems in image processing and vlsi. *Journal of ACM*, 32(1):130–136, 1985.

[8] C. Huang, P. Chen, Y. Tseng, and W. Chen. Models and algorithms for coverage problems in wireless sensor networks. In *Handbook on Theoretical Aspects and Algorithmic Aspect of Sensor, Ad-Hoc Wireless, and Peer-to-Peer Networks, Ed. Jie Wu*. CRC Press, 2006.

[9] C. Huang and Y. Tseng. The coverage problem in a wireless sensor network. In WSNA, pages 115–121. 2003.

[10] R. Kershner. The number of circles covering a set. *American Jr. of Mathematics*, 61:665–671, 1939.

[11] http://groups.yahoo.com/group/lp_solve.

[12] G. Lueker. Two np-complete problems in nonnegative integer programming. *Report 178, Princeton University*, 1975.

[13] S. Sahni and X. Xu. Algorithms for wireless sensor networks. *Intl. Jr. on Distr. Sensor Networks*, pages 35–56, 2004. Invited Paper, Preview Issue.

[14] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill. Integrated coverage and connectivity configuration in wireless sensor networks. In *Proc. First Intl. Conf. on Embedded Network Sensor Systems*, pages 28–39, 2003.

[15] X. Wang and N. Zhong. Efficient point coverage in wireless sensor networks. *Combinatorial Optimization*, 11:291–304, 2006.

[16] X. Xu and S. Sahni. Approximation algorithms for sensor deployment. *IEEE Trans. on Computers*, 56:1681–1695, 2007.

[17] H. Zhang and J. Hou. Maintaining sensing coverage and connectivity in large sensor networks. In *NSF International Workshop on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless, and Peer-to-Peer Networks*, 2004.