

Deploying Directional Sensor Networks with Guaranteed Connectivity and Coverage

Xiaofeng Han^{*} Xiang Cao^{*} Errol L. Lloyd[◇] Chien-Chung Shen^{*}

Abstract—In contrast to existing work on the connected coverage problem in wireless sensor networks which assumes omnidirectional sensors with disk-like sensing range, this paper investigates a suite of novel problems related to connected coverage in *directional* sensor networks where sensors only sense directionally and have a sector-like sensing range. We first consider the problems of deploying a minimum number of directional sensors to form a connected network to cover either a set of point-locations (Connected Point-Coverage Deployment (CPD)) or the entire target sensing area (Connected Region-Coverage Deployment (CRD)). CPD is NP-hard as its subproblem of Geometric Sector Cover (GSC) is NP-hard. We present two approximation algorithms for GSC as subroutines, and develop a general solution framework for CPD with approximation ratio $\sigma + O(1)$, where σ is the approximation ratio of the selected GSC subroutine. We also describe two efficient deployment patterns with guaranteed covering density for CRD, and analyze their performance bounds with respect to arbitrary non-crossing deployment patterns. Extensive simulation results validate the correctness and merits of the presented algorithms and analysis.

Index Terms—Directional wireless sensor networks, Coverage, Connectivity, Approximation algorithm.

I. Introduction

In wireless sensor networks, to successfully accomplish the assigned sensing tasks, the deployed sensors must both cover certain specific point-locations or the entire target sensing area, and form a connected network through multi-hop wireless communications. Therefore, deploying wireless sensor networks to provide *connected coverage* is a critical research issue. Most of the existing solutions to the connected coverage problem in wireless sensor networks assume *omni-directional* sensors with disk-like sensing range [1], [2], [3]. However, in real applications, due to equipment constraints or environmental impairments, certain sensors may only sense directionally and facilitate a sector-like sensing range, which are termed *directional* sensors. For instance, multimedia and smart camera sensors [4], [5], and the sensors that provide sensing through ultrasonic or infrared [6], [7] are widely used directional

The authors are with the Department of Computer and Information Sciences, University of Delaware, U.S.A. Email: {han,cao,elloyd,cshen}@cis.udel.edu. Preliminary work of this paper has appeared as a two-page poster in IEEE INFOCOM 2007, titled "Directional Wireless Sensor Networks with Guaranteed Connectivity and Coverage."

^{*}The research of these authors is supported in part by the National Science Foundation under grant CNS-0347460.

[◇]Prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes not withstanding any copyright notation thereon.

sensors¹. In this paper, we study a suite of problems related to connected coverage in the context of directional sensor networks. To our best knowledge, this is the first effort to address such problems.

We consider a scenario where directional sensors can be precisely deployed at any location within a target sensing area, and address the following two problems, *Connected Point-Coverage Deployment* and *Connected Region-Coverage Deployment*, which aim to use a minimum number of directional sensors to form a connected network to cover a set of point-locations and the entire target sensing area, respectively.

Connected Point-Coverage Deployment (CPD) is NP-hard as its subproblem of covering a set of points with a minimum number of sectors, termed Geometric Sector Cover (GSC), is NP-hard². Therefore, our efforts focus on designing polynomial-time solutions with provable approximation ratios³ for CPD. We first present two approximation algorithms for GSC. The first algorithm works for sectors with arbitrary angle and has approximation ratio $\log|P| + 1$, where P is the given set of points to cover. The second algorithm works for sectors whose angle is no less than $\frac{\pi}{3}$ and has approximation ratio 9. We then develop a general framework for CPD using either algorithm for GSC as a subroutine, and prove that the approximation ratio of the framework is $\sigma + O(1)$, where σ is the approximation ratio of the GSC subroutine utilized. While these approximation ratios may be large in some circumstances, recall that they are worst case values. Simulation results show that the solutions provided by our framework using either subroutine are far better than the derived approximation ratios suggest, indicating that our algorithms work well in real applications.

Connected Region-Coverage Deployment (CRD) is closely related to the *Covering* problem in Computational Geometry. Assuming that a 2D area A is covered by a set C of instances of a geometric body K , then the *covering density* of this coverage is defined as $\frac{|C| \times \|K\|}{\|A\|}$, where $\|A\|$ and $\|K\|$ stand for the areas of A and K , respectively. We present two efficient patterns with bounded covering density for CRD. We also analyze the performance bound of covering densities for our solutions with respect to any *non-crossing* deployment patterns

¹Note that the directional sensing capability is orthogonal to a sensor node's communication capability, where sensors still communicate omni-directionally.

²GSC is the general case of the well known NP-hard problem of Geometric Disk Cover [8].

³A solution solving a minimization problem is an x -approximation solution (or has approximation ratio x), if the result provided by the solution is no more than x times the optimal result [9].

(see Definition 5.3 in Section V for a formal definition of non-crossing deployment pattern), provided that the communication radius of sensors is appropriately sized related to their sensing radius. Finally, we evaluate these two deployment patterns using directional sensors with different configurations of communication radius, sensing radius and sensing angle.

The contributions of our paper are summarized as follows: (1) We present two approximation algorithms for GSC as sub-routines, and develop a general framework with approximation ratio $\sigma + O(1)$ for CPD, where σ is the approximation ratio of the selected GSC subroutine. (2) We present two efficient deployment patterns for CRD and analyze their covering densities for directional sensors with different communication radius, sensing radius and sensing angle. (3) All of the algorithms presented in this paper do not impose any constraint on the relative order between sensing radius and communication radius, and hence are suitable for a wide range of applications.

The remainder of this paper is organized as follows. Section II reviews related work. Section III describes the network model of directional sensor networks, and formally defines the problems. In Section IV, we present two approximation algorithms for GSC and a general framework for CPD, and analyze the approximation ratios of these solutions. Section V gives two deployment patterns with bounded covering density for CRD and evaluates their performance. Section VI concludes the paper with some future research directions.

II. Related Work

Extensive research on connected coverage has been done for omni-directional sensor networks (see [10] for a survey). For example, [2] presented a strip-based deployment pattern to achieve connected region-coverage for sensors whose communication radius r_c is less than or equal to their sensing radius r_s . In [1], an efficient deployment pattern was given for connected region-coverage, provided $r_c \leq \sqrt{3}r_s$. Recently, [3] proved that the result in [2] is asymptotically optimal, and described a general deployment solution to connected region-coverage for sensors using arbitrary r_c and r_s . In terms of connected point-coverage, a $(4\pi/\sqrt{3})$ -approximation algorithm is given in [11] under the assumption that $r_s = r_c$. We are not aware of any work that addresses connected coverage in the context of directional sensor networks.

Recently, a few research efforts have been devoted to directional sensor networks. Reference [12] presented algorithms for randomly deployed directional sensor networks to identify a minimal set of directions to cover the maximal number of point-locations. In [13], several scheduling algorithms were presented to divide a randomly deployed sensor network into subsets to alternatively cover a set of point-locations so as to prolong network lifetime. In [14], the authors analyzed the probability that a point-location is sensing by a randomly deployed directional sensor networks(not effectively sensed). However, the problems, objectives, and algorithmic approaches presented in this paper are completely different from these earlier research efforts.

Last, we review the work on the covering problem in the field of Computational Geometry, which does not impose any connectivity constraint. The problems of covering a set of points using a minimum number of a given geometric body are generally NP-hard, and approximation algorithms had been proposed for disks [8], [15], orthogonal rectangles [16], and fat convex bodies [17] (see [18] for a formal definition of fat convex body). In terms of region coverage (covering a vast 2D square region with a given geometric shape), for a very long time the only shape with known optimal covering density was the disk. Only recently did [18] prove that the optimal covering density of any fat convex body K is $\frac{\|K\|}{\|H(K)\|}$, where $H(K)$ is the hexagon with maximum area inscribed in K . However, covering a set of points or a region using sectors is significantly harder than using disks or fat convex bodies. We are not aware of any existing work on these issues.

III. Network Model and Problem Formulation

We consider stationary, directional sensors, whose sensing range is a sector centered at each sensor with a sensing radius r_s and a sensing angle α ($0 < \alpha < \pi$). Moreover, we assume both sensors and data sinks communicate omni-directionally, represented by a disk communication range centered at each sensor or data sink with communication radius r_c .

Throughout the paper, we use $s(r_c, r_s, \alpha)$ to represent *directional sensors*, and use $s(r, \alpha)$ to represent a *sector* with radius r and angle α . Furthermore, let X be a set, and $|X|$ stands for the cardinality of X . Given two points a and b and a 2D area A , $\|ab\|$ represents the Euclidean distance between a and b , and $\|A\|$ represents the area of A .

Using these assumptions and notations, the problems we study in this paper are formally defined as follows:

- *Connected Point-Coverage Deployment (CPD)*: Given r_c, r_s, α and a set of points P in a 2D area. CPD seeks to deploy a minimum number of directional sensors $s(r_c, r_s, \alpha)$ to form a connected network to cover P .
- *Connected Region-Coverage Deployment (CRD)*: Given r_c, r_s, α and a vast 2D square area A . CRD seeks to find a pattern with minimum covering density to deploy directional sensors $s(r_c, r_s, \alpha)$ to form a connected network to cover A .

IV. Connected Point-Coverage Deployment (CPD)

This section presents a general framework with provable approximation ratio for CPD. In a nutshell, our framework consists of two phases. Phase one solves the Geometric Sector Cover problem and covers all of the given point-locations with directional sensors; phase two places additional sensors to connect all of the sensors deployed in phase one.

A Geometric Sector Cover Problem (GSC)

One approach to GSC is to combine $\lceil \frac{2\pi}{\alpha} \rceil$ sectors together into a complete disk, and then use the known Polynomial Time Approximation Scheme (PTAS) [8] for Geometric Disk Cover to cover all of the given points. Such a scheme yields a $(\lceil \frac{2\pi}{\alpha} \rceil + \epsilon)$ -approximation algorithm for GSC. However, this

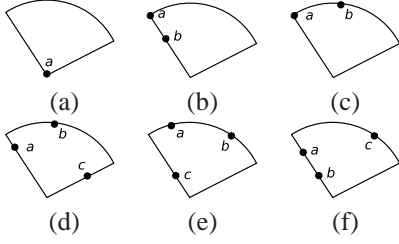


Fig. 1. Examples of sectors anchored by one, two and three points

naïve approach is not particularly useful because (1) when α is small, the resulting approximation ratio can be arbitrarily large, and (2) when α is large, the algorithm requires a long running time to achieve a reasonable approximation ratio. Therefore, we propose two polynomial-time approximation algorithms with provable approximation ratios for GSC. The first algorithm is suitable for sectors with arbitrary angle. The second algorithm is designed for sectors whose angle is no less than $\frac{\pi}{3}$.

A.1 Greedy Algorithm for GSC

The key idea of the first algorithm is to solve GSC using a greedy algorithm analogous to a greedy algorithm for the Set Cover problem [19]. We start with the following definition.

Definition 4.1: [Anchored Sector]: A sector $s(r, \alpha)$ is an anchored sector if it satisfies one of the following three conditions: (1) $s(r, \alpha)$ covers one point a , and a coincides with the center of $s(r, \alpha)$, as shown in Figure 1(a); (2) $s(r, \alpha)$ covers two or more points, and there exist two points a and b located on the same edge (arc) of $s(r, \alpha)$, and a coincides with one of the end points of this edge (arc). Moreover, there are no other points located on the other edge (arc), as shown in Figures 1(b) and 1(c); (3) $s(r, \alpha)$ covers three or more points, and there exist three points a , b , and c such that a , b , and c are on different edges (arc), as shown in Figure 1(d); or a and b are on the same edge (arc), while c is on one of the other edges (arc), as shown in Figures 1(e) and 1(f). ■

Algorithm 1 presents a greedy algorithm for GSC. Considering different point combinations and all of the possible cases in Definition 4.1, the number of sectors anchored by one, two or three points in P is at most $\binom{|P|}{1}$, $\binom{|P|}{2}\binom{|P|}{1}\binom{|P|}{1}$ and $\binom{|P|}{3}\binom{|P|}{1}\binom{|P|}{1} + 2 \times \binom{|P|}{2}\binom{|P|}{1}\binom{|P|}{1}$, respectively. Therefore, in each execution of step 5 in Algorithm 1, we only need to consider at most $O(P^3)$ sectors and the time complexity of Algorithm 1 is $O(|P|^4)$, which results from repeatedly finding the sector that covers the most uncovered points. To derive the approximation ratio of Algorithm 1, we first prove the following lemma.

Lemma 4.2: Let P be a set of points. There exists an optimal set C of sectors $s(r, \alpha)$ with minimum cardinality that covers P , where each sector in C is anchored by one, two, or three points in P .

Proof: Let OPT_{GSC} be an optimal set of sectors $s(r, \alpha)$ that covers P . We perform a *transformation* on each sector in OPT_{GSC} to produce the set C where each sector in C is anchored by one, two, or three points in P , and the cardinality of C is the same as that of OPT_{GSC} . For an arbitrary unanchored

Algorithm 1 Greedy Algorithm for GSC

```

1: INPUT: Sector  $s(r, \alpha)$  and a set of points  $P$ .
2: OUTPUT: A set  $S$  of sectors  $s(r, \alpha)$  that cover  $P$ .
3:  $R \leftarrow P$ ;  $S \leftarrow \emptyset$  (empty set);  $i = 1$ ;
4: while  $R$  is not empty do
5:   Find a sector  $s_i(r, \alpha)$  anchored by one, two, or three points
     in  $R$  that covers a maximum number of points in  $R$ . Let
      $R_i$  be the set of points of  $R$  that are covered by  $s_i(r, \alpha)$ ;
6:    $S = S \cup \{s_i(r, \alpha)\}$ ;  $R = R \setminus R_i$ ;  $i = i + 1$ ;
7: end while
8: Output  $S$ ;
```

sector $s_i(r, \alpha)$ in OPT_{GSC} , let o be the center of $s_i(r, \alpha)$, and o' be the middle point of the arc in $s_i(r, \alpha)$.

When $s_i(r, \alpha)$ only covers one or two points in P , we can move $s_i(r, \alpha)$ and let $s(r_s, \alpha)$ be anchored by the points it covers, as shown in Figures 1(a), 1(b) or 1(c).

When $s_i(r, \alpha)$ covers three or more points, we first move $s_i(r, \alpha)$ along the line oo' in the $\overrightarrow{o'o}$ direction until one or more points touch the arc. We consider the two cases separately.

- If more than one point touches the arc, we fix the center of $s_i(r, \alpha)$ at the current location and rotate $s_i(r, \alpha)$ clockwise or counterclockwise until at least one point touches the edge. Then $s_i(r, \alpha)$ is anchored, as shown in Figure 1(c) or 1(e).

- If only one point touches the arc, we still fix the center of $s_i(r, \alpha)$ at the current location and rotate $s_i(r, \alpha)$ clockwise or counterclockwise until one or more points touch the edge. (1) If more than one point touches the edge, then $s_i(r, \alpha)$ is anchored, as shown in Figure 1(f); (2) If only one point touches the edge, we have one point x on the arc and one point y on one of the edges. Then we keep x on the arc and y on the edge, and move $s_i(r, \alpha)$ to make x approach the edge where y resides until one of the following two cases happens. Case 1: we move $s_i(r, \alpha)$ until x coincides with the end point of the edge where y resides, then $s_i(r, \alpha)$ is anchored, as shown in Figure 1(b). Case 2: we move $s_i(r, \alpha)$ until another point z touches the arc or edge, then $s_i(r, \alpha)$ is anchored as shown in Figures 1(d), 1(e) or 1(f).

Note that in the transformation, $s_i(r, \alpha)$ always covers all of the points it originally covers. Therefore, after performing the transformation on each sector in OPT_{GSC} , we obtain an optimal set C that covers P where each sector in C is anchored by one, two, or three points in P . ■

The next theorem, which follows directly from the proof of Lemma 4.2, states the approximation ratio of Algorithm 1.

Theorem 4.3: Let P be a set of points. Algorithm 1 is an $(\log|P| + 1)$ -approximation algorithm for GSC.

Proof: Let S_P be the set of sectors that are anchored by one, two or three points in P . Let S be the set of sectors that result from running Algorithm 1 on P . By Lemma 4.2, there exists an optimal set C ($C \subseteq S_P$) of anchored sectors $s(r, \alpha)$ that covers P . According to the known approximation ratio of a

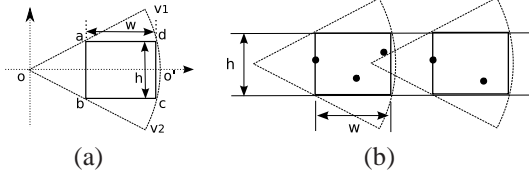


Fig. 2. Strip-based algorithm for GSC. $h = r \times \sin(\frac{\alpha}{2})$, $w = \frac{r}{2} \sqrt{4 - \sin^2(\frac{\alpha}{2})} - \cos(\frac{\alpha}{2})$. (a) Find an inscribed rectangle $abcd$ in a sector $s(r, \alpha)$. o is the center of the sector; o' is the middle point of the arc; v_1 and v_2 are the two end points of the arc. a and b are the middle points of ov_1 and ov_2 ; ad and bc are parallel to oo' . (b) Step 8 of Algorithm 2 which covers the points in a strip using such inscribed rectangles.

greedy algorithm for the Set Cover problem [19]⁴, we have $|S| \leq (\log|P|+1)|C|$. Therefore, Algorithm 1 is a $(\log|P|+1)$ -approximation algorithm for GSC. ■

A.2 Strip-Based Algorithm for GSC Using Large Sectors

In practice, it is expected that the sensing angle of directional sensors is not small. We thus give a constant approximation algorithm for GSC using sectors with angle $\frac{\pi}{3} \leq \alpha < \pi$.

The key idea of this algorithm is to (1) divide the entire sensing area into strips of fixed height $r \times \sin(\frac{\alpha}{2})$, and (2) cover the points in each strip from left to right using a rectangle inscribed in the sector with height $r \times \sin(\frac{\alpha}{2})$ and width $\frac{r}{2} \sqrt{4 - \sin^2(\frac{\alpha}{2})} - \cos(\frac{\alpha}{2})$. Figure 2(a) depicts how to compute such an inscribed rectangle. The complete algorithm is presented in Algorithm 2.

Algorithm 2 Strip-Based Algorithm for GSC

- 1: **INPUT:** Sectors $s(r, \alpha)$ ($\frac{\pi}{3} \leq \alpha < \pi$) and point set P .
 - 2: **OUTPUT:** A set S of sectors $s(r, \alpha)$ that covers P .
 - 3: $S \leftarrow \emptyset$; $i \leftarrow 1$;
 - 4: Divide the sensing area into strips with height $r \times \sin(\frac{\alpha}{2})$;
 - 5: **for** each strip **do**
 - 6: Let $x(p)$ be the x -coordinates of point p . Sort the points in the strip according to their x -coordinates and obtain $P_s = \{p_1, p_2, \dots, p_m\}$, where $x(p_1) \leq x(p_2) \leq \dots \leq x(p_m)$;
 - 7: **while** P_s is not empty **do**
 - 8: Pick the first point p_j in P_s , and place a sector $s_i(r, \alpha)$ such that its inscribed rectangle covers the points with x -coordinates from $x(p_j)$ to $x(p_j) + \frac{r}{2} \sqrt{4 - \sin^2(\frac{\alpha}{2})} - \cos(\frac{\alpha}{2})$, as shown in Figure 2(b);
 - 9: Remove the points covered by $s_i(r, \alpha)$ from P_s ;
 - 10: $S = S \cup \{s_i(r, \alpha)\}$; $i = i + 1$;
 - 11: **end while**
 - 12: **end for**
 - 13: Output S ;
-

The time complexity of Algorithm 2 is $O(|P| \log|P|)$, which results from sorting the points in each strip. The following

⁴In [19], the authors proved that the greedy algorithm, which has an approximation of ratio $\log|P| + 1$, is the best approximation algorithm for Set Cover problem, and no polynomial-time algorithm can achieve a better approximation ratio unless $P = NP$.

theorem states the approximation ratio.

Theorem 4.4: Let P be a point set, and $s(r, \alpha)$ ($\frac{\pi}{3} \leq \alpha < \pi$) be a sector. Then the performance ratio of Algorithm 2 is 9.

Proof: Let OPT_{GSC} be an optimal set of sectors $s(r, \alpha)$ ($\frac{\pi}{3} \leq \alpha < \pi$) that covers P . For any sector $s_i(r, \alpha)$ in OPT_{GSC} , since $\frac{\pi}{3} \leq \alpha < \pi$, the longest distance between any two points in $s_i(r, \alpha)$ is at most $2r \times \sin(\frac{\alpha}{2})$, and each inscribed rectangle has length $\frac{r}{2} \sqrt{4 - \sin^2(\frac{\alpha}{2})} - \cos(\frac{\alpha}{2})$, therefore in each strip Algorithm 2 uses at most $\lceil \frac{4 \times \sin(\frac{\alpha}{2})}{(\frac{r}{2} \sqrt{4 - \sin^2(\frac{\alpha}{2})} - \cos(\frac{\alpha}{2}))} \rceil \leq 3$ sectors to cover the points that are covered by $s_i(r, \alpha)$. Moreover, since each strip has height $r \times \sin(\frac{\alpha}{2})$, $s_i(r, \alpha)$ spans at most 3 strips. As a result, Algorithm 2 uses at most 9 sectors to cover all of the points that are covered by $s_i(r, \alpha)$, which provides the performance ratio of 9. ■

B. Algorithm for Connected Point-Coverage Deployment

Regarding the two approximation algorithms for GSC described in previous section, the strip-based algorithm, although it has a better running time and a constant approximation ratio, can only be applied to sectors with large angles. On the other hand, the greedy algorithm actually achieves smaller approximation ratios when $|P| < e^8$. Therefore, this section presents a general framework for CPD, which can use either approximation algorithm for GSC as a subroutine. Such a design allows users to choose a proper subroutine and balance the trade-offs between quality and performance. The complete framework is given as Algorithm 3.

Algorithm 3 A framework for CPD

- 1: **INPUT:** Sensor $s(r_c, r_s, \alpha)$ and a set P of points.
 - 2: **OUTPUT:** A set S of connected sensors that covers P .
 - 3: $S \leftarrow \emptyset$; $S_1 \leftarrow \emptyset$; $S_2 \leftarrow \emptyset$;
 - 4: Execute a σ -approximation algorithm for GSC on P using sector $s(r_s, \alpha)$, and put the resulting sensors in S_1 ;
 - 5: Create a complete graph $C = (S_1, E)$, where each edge $s_i s_j \in E$ has weight $\lceil \frac{\|s_i s_j\|}{r_c} \rceil - 1$;
 - 6: Find a Minimum Spanning Tree T of C . For each edge $s_i s_j$ in T , evenly place $\lceil \frac{\|s_i s_j\|}{r_c} \rceil - 1$ sensors on the straight line between s_i and s_j , and put all of the sensors in S_2 ;
 - 7: Output $S = S_1 \cup S_2$;
-

Let d be the maximum distance between any two points in P . The time complexity of Algorithm 3 is $O(T_{GSC} + |P|^3 + |P| \times \frac{d}{r_c})$, where T_{GSC} is the time complexity of the GSC subroutine, $O(|P|^3)$ results from computing a Minimum Spanning Tree, and $|P| \times \frac{d}{r_c}$ is from Step 6 in Algorithm 3.

We now derive the approximation ratio of Algorithm 3. Let σ , P , S_1 , S_2 , C , and T be as defined in Algorithm 3. Let $OPT_{CPD} = \{os_1, os_2, \dots, os_o\}$ be an optimal set of sensors $s(r_c, r_s, \alpha)$ for CPD on P . We have the following lemma.

Lemma 4.5: $|S_1| \leq \sigma \times |OPT_{CPD}|$ ■

We define $\theta = (\lceil \frac{4\sqrt{2}r_s}{r_c} \rceil)^2 + 2\lceil \frac{2r_s}{r_c} \rceil + 6$. The next lemma is the key result towards deriving the approximation ratio of Algorithm 3.

Lemma 4.6: $|S_2| \leq \theta \times |OPT_{CPD}|$

Proof: Let C_{opt} be a complete graph over OPT_{CPD} , where each edge $os_x os_y$ in C_{opt} carries weight $\|os_x os_y\|$. Since the edge weights in C_{opt} satisfy the triangle inequality, there exists a minimum spanning tree $T_{opt} = (OPT_{CPD}, E_{opt})$ (rooted at os_{root} , $os_{root} \in OPT_{CPD}$) of C_{opt} , where each sensor in T_{opt} has at most 5 neighbors. We show that by using T_{opt} as a backbone, we can find a spanning tree $T_C = (S_1, E_C)$ of C whose total weight is no more than $\theta \times |OPT_{CPD}|$.

To facilitate the creation of T_C , we associate each sensor s_i in T_C with one sensor os_x in T_{opt} , where s_i and os_x cover certain common points in P . For simplicity, we term each sensor os_x in T_{opt} as a *leader* sensor, and term the sensors in T_C that are associated with os_x as os_x 's *follower* sensors. In this proof, we consider the worst case where each leader sensor is associated with some follower sensors. We define the leading zone of os_x as a circle centered at os_x with radius $2r_s$. Clearly, all of the follower sensors associated with os_x are within os_x 's leading zone. We create T_C in two steps.

In the first step, we connect the follower sensors associated with the same leader sensor. Specifically, for each leader sensor os_x , we create a square just large enough to encapsulate its leading zone, then evenly divide the square into $(\lceil \frac{4\sqrt{2}r_s}{r_c} \rceil)^2$ square cells labeled as (left to right, top to bottom) C_1, \dots, C_n , $n = (\lceil \frac{4\sqrt{2}r_s}{r_c} \rceil)^2$. Figure 3(a) depicts an example of dividing a leading zone into 16 cells. Denote the square cells containing follower sensors as $C_{i1}, C_{i2}, \dots, C_{im}$, where $i1 < i2 < \dots < im$. Note that all of the follower sensors within the same cell are connected since the diagonal of each cell is less than or equal to r_c . We first find an arbitrary spanning tree in each cell and add the edges to E_C . Then, for all of the cell pairs C_{ij} and C_{ij+1} , we select two follower sensors s_x and s_y (one from each cell) and add edge $s_x s_y$ ($s_x s_y \in C$) to E_C . As a result, os_x 's follower sensors are connected. We charge the total weight of added edges to os_x , which is at most $im - i1 \leq (\lceil \frac{4\sqrt{2}r_s}{r_c} \rceil)^2$.

In the second step, we connect the follower sensors in different leading zones. For the ease of description, we use '*leading zone of os_x* ' as a shorthand for 'all of the follower sensors associated with os_x ,' and define the operation of connecting two leading zones as adding an edge in C between two follower sensors (one from each leading zone) to E_C . We conduct a breadth-first search on T_{opt} , label the root os_{root} as layer 1, and label the children of leader sensors in layer i as layer $i + 1$. Assuming T_{opt} has totally n layers, we connect the leading zones bottom-up starting from layer $n - 1$.

Each leader sensor os_y in layer $n - 1$ has at most five children in T_{opt} , denoted as $os_{y1}, os_{y2}, \dots, os_{y5}$. As depicted in Figure 3(b), we first connect the leading zones of each leader sensor pair os_{yi} and os_{yi+1} , then we connect the leading zones of os_{y5} and os_y . As a result, we add two edges for the leading zones of $os_{y2}, os_{y3}, os_{y4}$ and os_{y5} , and add one edge for the leading zones of os_{y1} and os_y . We term the leading zones with two added edges '*saturated* leading zones,' and the leading zones with one added edge '*unsaturated* leading zones.' To count the total weight of the added edges, we charge the weight of

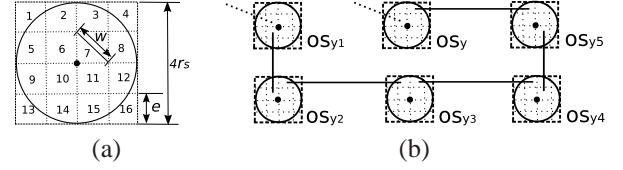


Fig. 3. (a) Divide the leading zone into 16 cells. $e = 4r_s / \lceil \frac{4\sqrt{2}r_s}{r_c} \rceil$, $w = \sqrt{2}e^2 \leq r_c$. (b) Connect the leading zones. The leading zones of $os_{y2}, os_{y3}, os_{y4}$ and os_{y5} are saturated, while the leading zones of os_{y1}, os_y are unsaturated.

the added edges to the leader sensors. Specifically, for each edge connecting leading zones of os_{yi} and os_{yi+1} , we charge $\lceil \frac{2r_s}{r_c} \rceil$ to os_{yi} and os_{yi+1} , and charge 1 to each node on the shortest path between os_{yi} and os_{yi+1} in T_{opt} , including os_{yi} and os_{yi+1} .

Assuming that we have connected the leading zones in each subtree of layer $i - 1$ leader sensors, we now consider the leader sensors in layer i . Each leader sensor os_z in layer i has at most five children $os_{z1}, os_{z2}, \dots, os_{z5}$, and all of the leading zones in the subtree of os_{zi} are connected. For each node pair os_{zi} and os_{zi+1} , we connect their subtrees by connecting two unsaturated leading zones (one from each subtree). Then we connect the subtrees of os_{z5} and os_z in the same way. As a result, all of the leading zones of os_z and its descendants are connected, while two leading zones remain *unsaturated*. Again, we charge the total weight of the added edges to the leader sensors as we did for leader sensors in layer $n - 1$.

We repeat the above step for each layer until reaching the root os_{root} , which results in a spanning tree T_C of C . By combining the charges to each leader sensor in T_{opt} in both steps, a leader sensor os_w will be charged at most $(\lceil \frac{4\sqrt{2}r_s}{r_c} \rceil)^2 + 2\lceil \frac{2r_s}{r_c} \rceil + 6$. Therefore, the total weight of T_C is at most $\theta \times |OPT_{CPD}|$. As a result, $|S_2|$, which is the total weight of T , is no more than $\theta \times |OPT_{CPD}|$. ■

The following theorem, which follows directly from Lemma 4.5 and Lemma 4.6, describes the approximation ratio of the presented framework.

Theorem 4.7: Let P be a set of points, and let $s(r_c, r_s, \alpha)$ be a directional sensor. The framework described in Algorithm 3 for CPD has an approximation ratio of $\sigma + \theta$, where σ is the approximation ratio of the selected GSC subroutine, and $\theta = (\lceil \frac{4\sqrt{2}r_s}{r_c} \rceil)^2 + 2\lceil \frac{2r_s}{r_c} \rceil + 6$. ■

C Simulation Result

We evaluate the presented framework using different sub-routines with simulations. We first consider a special case where $\alpha = \frac{\pi}{3}$, $r_c = r_s$ and the point-locations are scattered on the rectangular lattices, as shown in Figures 4(a). In this case, each sensor's sensing range can cover at most 4 point-locations. Therefore, the optimal solution to CPD, which is shown as dotted sectors with solid center in Figures 4(a), can be derived manually. We thus compare the performance of the presented framework using different subroutines against the optimal solution, and the results are shown in Figures 4(b). On the average, the greedy and strip-based GSC subroutines

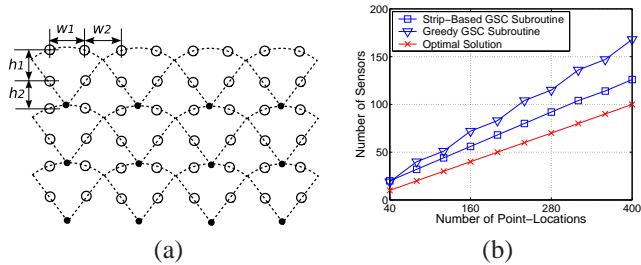


Fig. 4. Hollow circles stand for the point-locations. Dotted sectors with solid circle stand for the optimal sensors that cover all of the points. (a) Point-locations scattered on rectangular lattices. $r_c = r_s$, $\alpha = \pi/3$. $w_1 = w_2 = r_s/2$, $h_1 = \frac{r_s}{2}\sqrt{4 - \sin^2(\alpha/2)} - \cos(\alpha/2)$, $h_2 = r_s - h_1$. (b) Performance of the presented framework for the rectangular lattices case.

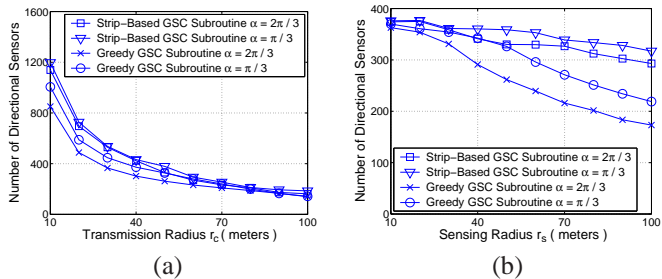


Fig. 5. (1) $r_s = 50m$, r_c increase from 10m to 100m. (2) $r_c = 70m$, r_s increase from 10m to 100m.

only use 1.75 times and 1.43 times more sensors than does the optimal solution, respectively. We repeat this experiment for other cases where point-locations are scattered according to other patterns, such as triangular lattice, and find that both greedy and strip-based GSC subroutines use at most 4 times more sensors than does the optimal solution.

We next consider the case where 200 point-locations are randomly distributed in a $1000 \times 1000 m^2$ 2D area. We use directional sensors with different configurations of r_s , r_c and α , and run 50 simulations for each configuration. The average results are illustrated in Figure 5. As we can observe from the figures, when the value of r_s , r_c or α increases, the number of sensors computed by both the greedy GSC subroutine and the strip-based GSC subroutine drops accordingly. Moreover, on the average, the strip-based GSC subroutine uses 29.3% and 25% more sensors than does the greedy GSC subroutine when α equals $\frac{2\pi}{3}$ and $\frac{\pi}{3}$, respectively.

In summary, the presented CPD framework can achieve connected point-coverage using directional sensors with different r_c , r_s and α . Moreover, the results provided by the presented framework using either the greedy GSC subroutine or the strip-based GSC subroutine are far better than the corresponding approximation ratio suggested, indicating that our algorithms work well in real applications.

V. Connected Region-Coverage Deployment (CRD)

This section gives two deployment patterns for CRD that work for directional sensors with arbitrary r_c , r_s and α ($0 < \alpha < \pi$). Recall that the objective of CRD is to find deployment

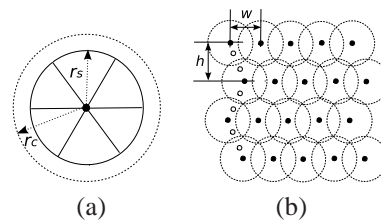


Fig. 6. (a) Combine six directional sensors $s(r_c, r_s, \frac{\pi}{3})$ together as an omnidirectional sensor. (b)[3] Optimal deployment pattern for omnidirectional sensors using r_c and r_s . $w = \min(r_c, \sqrt{3}r_s)$, $h = r_s + \sqrt{r_s^2 - \frac{w^2}{4}}$. When $d = \sqrt{h^2 + (\frac{w}{2})^2} > r_c$, $\lceil \frac{d}{r_c} \rceil$ relay sensors, shown as hollow circles, need to be deployed between each pair of horizontal strips of sensors to guarantee network connectivity.

patterns with minimum covering density to place directional sensors as a connected network to cover a 2D square area A . We follow the assumption in [2], [3] and assume that A is sufficiently large so that the coverage waste beyond the boundary can be omitted. Notice that for a limited region of particular shape, such as limited disk, triangle and rectangle regions, it is possible to derive specific and more efficient deployment solutions, but this is beyond the scope of this paper.

A. Disk-Based Deployment Pattern (DDP)

The first deployment scheme combines $\lceil \frac{2\pi}{\alpha} \rceil$ directional sensors $s(r_c, r_s, \alpha)$ together as a complete omnidirectional sensor using sensing radius r_s and communication radius r_c , as depicted in Figure 6(a). Then, these combined sensors are deployed according to the optimal deployment pattern described in [3] for omnidirectional sensors, as shown in Figure 6(b). Note that when $\frac{r_c}{r_s} < \sqrt{3}$, additional sensors, termed *relay sensors*, shown as hollow nodes in Figure 6(b), need to be deployed to guarantee network connectivity. Since the number of relay sensors is vanishingly small in comparison to the total number of sensors when A is sufficiently large, the overlapping areas introduced by relay sensors can be ignored [3].

Although the disk-based scheme does not make full use of the sector sensing range, it provides some insights into efficient deployment patterns, and can be used as a benchmark to evaluate other (more effective) solutions.

B. Strip-Based Deployment Patterns (SDP)

To describe the strip-based deployment pattern, we first give the following definition.

Definition 5.1: [*Tiling Body*]: Given a geometric body K and a large 2D area A , a set T_K of the instances of K is a *tiling* with K in A , if all of the instances in T_K are mutually disjoint, and the union of these instances in T_K equals A . Further, K is called a *Tiling Body* if a tiling with K in A exists. ■

The key idea of the strip-based deployment pattern is to first find a convex tiling body inscribed in the sector sensing range, and then deploy sensors as a connected network in horizontal strips so that the inscribed tiling bodies can mesh together and tile the 2D area A . As pointed out in [20], given a set of convex bodies (not necessarily congruent) that tile a 2D plane,

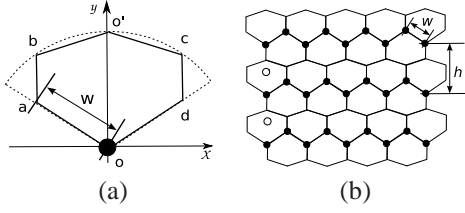


Fig. 7. (a) Find the inscribed p -hexagon in a sector $s(r_s, \alpha)$. (b) Strip-based deployment pattern using directional sensors $s(r_c, r_s, \alpha)$. $w \leq r_c$, $h = r_s(1 + \sqrt{r_s^2 - w^2} \times \sin^2(\frac{\alpha}{2}) - w \times \cos(\frac{\alpha}{2}))$. When $h > r_c$, $\lceil \frac{h}{r_c} \rceil$ relay sensors, shown as hollow circles, needs to be deployed between each pair of strips to guarantee network connectivity.

the average number of edges for these convex bodies is 6. We thus focus on finding a tiling body with no more than 6 edges. It is shown in [21] that each convex body contains a tiling body named p -hexagon, which is formally defined as follows.

Definition 5.2: [p -Hexagon]: A p -hexagon is a hexagon with a pair of parallel opposite sides of equal length. ■

Under Definition 5.2, extreme cases are allowed, in which adjacent sides are collinear, and some sides, even the parallel ones, may be reduced to points. Thus, each triangle is a p -hexagon, and so is each quadrilateral and each pentagon that has a pair of parallel sides. However, since for any triangle, quadrilateral or pentagon that is inscribed in sector, we can easily find an inscribed hexagon that contains this shape and has larger area, we focus on inscribed hexagons. Since the work in [21] did not specifically consider convex sectors and the connectivity constraint, we define the following operation to find a common p -hexagon inscribed within convex sectors to guarantee connected region-coverage.

Finding a p -hexagon inscribed within a sector: As shown in Figure 7(a), for sector $s(r, \alpha)$, o and o' denote the center of the sector and the middle point of the arc, respectively. We place this sector in an xy -coordinate system where o coincides with the origin, and o' resides on the y -axis. We find two points a and d on the two edges of the sector where $\|oa\| = \|od\| \leq r_c$. Then we find two more points b and c on the arc such that both ab and cd are vertical to the x -axis. The hexagon $oabo'cd$ is a p -hexagon.

By using Lagrangian multiplier [22], we can find such a p -hexagon $oabo'cd$ ($\|oa\| \leq r_c$) with maximum area, and the corresponding deployment pattern using such p -hexagons to achieve connected region-coverage is shown in Figure 7(b).

C. Comparison with Non-Crossing Deployment Patterns

In general, the deployment patterns can be categorized as *crossing patterns* and *non-crossing patterns*, which are explained in following definition.

Definition 5.3: [*Non-crossing Deployment Pattern*]: Let K_1 and K_2 be two convex bodies. K_1 and K_2 are *non-crossing* if there exist two half planes L^+ and L^- such that $L^+ \cap K_1 \subset K_2$ and $L^- \cap K_2 \subset K_1$. Let A be a 2D area covered by a set of convex bodies following a deployment pattern. This deploy-

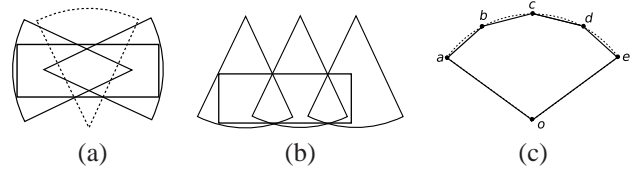


Fig. 8. (a) Crossing deployment pattern where the sector with dotted line crosses the other two sectors. (b) Non-crossing deployment pattern. (c) Maximum hexagon inscribed within a sector.

ment pattern is *non-crossing* if every pair of convex bodies in this covering is *non-crossing*. Figures 8(a) and 8(b) depict the difference between crossing and non-crossing patterns. ■

Finding the optimal pattern to deploy directional sensors, and proving its optimality is very difficult. This is so because rigorously analyzing the overlapping area induced by either the crossing sectors or connectivity requirement seems to be an intractable problem. We thus focus on analyzing the performance of DDP and SDP with arbitrary non-crossing deployment patterns in the cases where the connectivity requirement does not induce any overlapping area ($r_c \geq \sqrt{3}r_s$ for DDP, and $r_c \geq r_s$ for SDP). We start with the following theorem.

Theorem 5.4: [18] Let A be a large $l \times l$ 2D square area, and let K be a convex body. If A is covered by a large number of non-crossing instances of K , then the covering density of this coverage is no less than $\frac{\|K\|}{\|H(K)\|}$, where $H(K)$ is the hexagon with maximum area inscribed within K . ■

The next theorem, which is a standard result of Lagrangian multiplier, describes the maximum hexagon inscribed within a convex sector. Due to page limitations, the detailed proof is not provided.

Theorem 5.5: Let $s(r, \alpha)$ ($0 < \alpha < \pi$) be a sector, let o be the center of $s(r, \alpha)$, and let a and e be the two end points of the arc in $s(r, \alpha)$. The maximum hexagon inscribed within $s(r, \alpha)$ is $oabcde$ with area $2r^2 \sin(\frac{\alpha}{4})$, as shown in Figure 8(c), where b , c and d are evenly distributed on the arc. ■

SDP achieves the maximum covering density when $r_c \geq r_s$, as does DDP when $r_c \geq \sqrt{3}r_s$. On the other hand, Theorems 5.4 and 5.5 give the theoretical lower-bound of covering density for any non-crossing deployment patterns (not necessarily connected). We compare the maximum covering density of DDP and SDP with the theoretical lower-bound, and the results are given in Figure 9. When $r_c \geq r_s$, the covering density of SDP is very close to the theoretical lower-bound (5.7% more on the average). For DDP, when $r_c \geq \sqrt{3}r_s$, its covering density varies sharply when the sensing angle changes. This is so because when 2π is not divisible by α , there exists some overlap area in each combined disk. On the average, the covering density of DDP is 29.7% more than the theoretical lower-bound.

Using crossing sectors in a covering is very wasteful. Moreover, it is believed that the non-crossing requirement in Theorem 5.4 can be removed, provided the region needed to be covered is very large. Therefore, we conjecture the results presented in Figure 9 actually indicate the performance bound of the presented deployment patterns with respect to arbitrary deployment patterns, including the optimal solution.

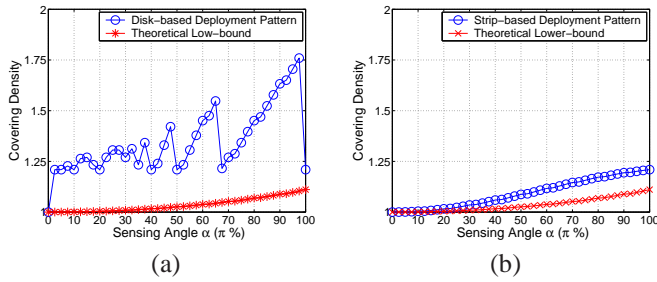


Fig. 9. Comparison with the theoretical lower-bound of non-crossing deployment patterns (a) Covering density of DDP when $r_c \geq \sqrt{3}r_s$. (b) Covering density of SDP when $r_c \geq r_s$.

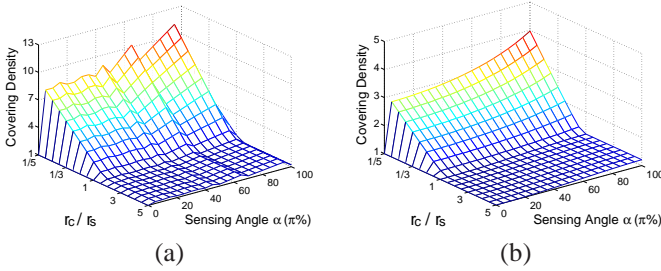


Fig. 10. Covering Density of presented deployment pattern for directional sensors using different r_c , r_s and α (a) Covering density of DDP. (b) Covering density of SDP.

D. Numerical Results

We further compare the covering density of DDP and SDP using directional sensors with different configurations of r_c , r_s and α . The numerical results are presented in Figure 10. We can see from these figures that SDP outperforms DDP in all of the configurations. Moreover, the covering density of both SDP and DDP decreases when the ratio $\frac{r_c}{r_s}$ decreases or when the sensing angle α increases.

VI. Conclusion and Future Work

In this paper, we studied several novel problems related to connected coverage in directional wireless sensor networks. Specifically, we presented two approximation algorithms for the GSC problem, and using these algorithms as the subroutine, developed a framework with an approximation ratio of $\sigma + O(1)$ for CPD, where σ is the approximation ratio of the selected GSC subroutine. We also presented two efficient patterns to deploy directional sensors for CRD. Future work includes finding better solutions to CPD and CRD, as well as investigating current problems with higher order connectivity or coverage requirement.

ACKNOWLEDGMENT: The authors would like to thank the anonymous reviewers for their time and valuable feedback. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government.

References

- [1] Y.-C. Wang, C.-C. Hu, and Y.-C. Tseng, "Efficient deployment algorithms for ensuring coverage and connectivity of wireless sensor networks," in *Proceedings of the First International Conference on Wireless Internet (WICON)*, 2005.
- [2] R. Iyengar, K. Kar, and S. Banerjee, "Low-coordination topologies for redundancy in sensor networks," in *Proceedings of ACM MobiHoc*, 2005.
- [3] X. Bai, S. Kumar, Z. Yun, D. Xuan, and T.-H. Lai, "Deploying wireless sensors to achieve both coverage and connectivity," in *Proceedings of ACM MobiHoc*, 2006.
- [4] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Computer Networks*, vol. 51, pp. 921–960, March 2007.
- [5] F. Qureshi and D. Terzopoulos, "Virtual vision and smart camera networks," in *Proceedings of Workshop on Distributed Smart Cameras (DSC 2006)*, 2006.
- [6] J. Djughash, S. Singh, G. Kantor, and W. Zhang, "Range-only slam for robots operating cooperatively with sensor networks," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2006.
- [7] R. Szwedczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler, "An analysis of a large scale habitat monitoring application," in *Proceedings of ACM SenSys*, 2004.
- [8] D. S. Hochbaum and W. Maass, "Approximation schemes for covering and packing problems in image processing and VLSI," *Journal of the ACM*, vol. 32, no. 1, pp. 130–136, 1985.
- [9] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, 1979.
- [10] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks," *Ad Hoc Networks*, no. doi:10.1016/j.adhoc.2007.05.003, in Press, 2007.
- [11] K. Kar and S. Banerjee, "Node placement for connected coverage in sensor networks," in *Proceedings of WiOpt*, 2003.
- [12] J. Ai and A. Abouzeid, "Coverage by directional sensors in randomly deployed wireless sensor networks," *Journal of Combinatorial Optimization*, vol. 11, no. 1, pp. 21–41, 2006.
- [13] Y. Cai, W. Lou, M. Li, and X.-Y. Li, "Target-oriented scheduling in directional sensor networks," in *Proceedings of IEEE INFOCOM*, 2007.
- [14] H. Ma and Y. Liu, "Some problems of directional sensor networks," *International Journal of Sensor Networks*, vol. 2, no. 1/2, pp. 44–52, 2007.
- [15] A. Srinivas, G. Zussman, and E. Modiano, "Mobile backbone networks construction and maintenance," in *Proceedings of ACM MobiHoc*, 2006.
- [16] T. Gonzalez, "Covering a set of points in multidimensional space," *Information Processing Letters*, vol. 30, no. 1, pp. 181–188, 1991.
- [17] K. L. Clarkson and K. Varadarajan, "Improved approximation algorithms for geometric set cover," in *Proceedings of ACM Annual Symposium on Computational Geometry*, 2005.
- [18] G. F. TÁ'oth, "Covering with fat convex discs," *Journal of Discrete and Computational Geometry*, vol. 34, no. 1, pp. 129–141, 2005.
- [19] U. Feige, "A threshold of $\ln N$ for approximating set cover," *Journal of the ACM (JACM)*, vol. 45, pp. 634–652, 1998.
- [20] R. Kershner, "The number of circles covering a set," *American Journal of Mathematics*, vol. 61, pp. 665–671, 1939.
- [21] W. Kuperberg, "Covering the plane with congruent copies of a convex body," *Bulletin of the London Mathematical Society*, vol. 21, pp. 82–86, 1989.
- [22] F. S. Hillier and G. J. Lieberman, *Introduction to Operations Research*. McGraw-Hill, 2002.