Minimum Ply Covering of Points with Convex Shapes

Ahmad Biniaz*　　Zhikai Lin†

Abstract

Introduced by Biedl, Biniaz, and Lubiw (CCCG 2019), the *minimum ply covering* of a point set P with a set S of geometric objects in the plane asks for a subset S' of S that covers all points of P while minimizing the maximum number of overlapping objects at any point in the plane (not only at points of P). This problem is NP-hard and cannot be approximated by a factor better than 2. Biedl et al. studied this problem for objects that are unit squares or unit disks. They present 2-approximation algorithms that run in polynomial time when the optimum objective value is bounded by a constant. We generalize this result and obtain a 2-approximation algorithm for any fixed-size convex shape. The new algorithm also runs in polynomial time if the optimum objective value is bounded.

1 Introduction

The problem of covering clients with antennas has been well studied in wireless networks [1, 3, 4, 5, 7, 9, 11]. Covering clients by placing new antennas can cause interference (this happens when more than one antenna cover the same region). Covering clients and—at the same time—reducing interference is a big challenge in wireless networks. In this paper we study a geometric problem that addresses this issue.

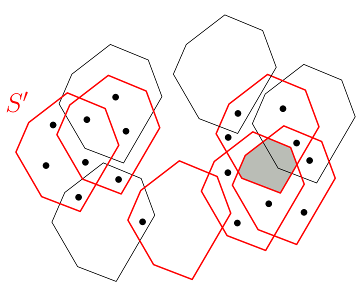

Figure 1: The ply of S' (shown in red) is 3.

Let P be a set of points and S be a set of geometric objects, both in the plane; each element of P represents a client and each object in S represents a coverage region of an antenna. We want to find a subset S' of S that covers all points in P and minimizes the maximum number of overlapping objects at any point in the plane. The *ply* of S' is the maximum number of overlapping objects of S' over all points of the plane. In other words,

$$ply(S') = \max_{p\in\mathbb{R}^2} |\{O \in S' : p \in O\}|.$$

See Figure 1 for an illustration. The term ply was used earlier by Eppstein and Goodrich [6]. With this definition, our goal is to find a subset of S, with minimum ply, that covers P. This problem is introduced by Biedl et al. [2], and it is known as the *minimum ply covering* (MPC). We denote an instance of the MPC problem by (P, S). The MPC problem has the same flavor as the geometric *minimum membership set cover* (MMSC) problem which asks for a subset S' of S that covers all points of P and minimizes the maximum number of overlapping objects only at points of P. Notice that the MPC problem minimizes the maximum number of overlapping objects over all points of the plane.

Erlebach and van Leeuwen [7] showed that the geometric MMSC problem is NP-hard for axis-aligned unit squares and unit disks, and it cannot be approximated by a factor better than 2 in polynomial time. According to Biedl et al. [2] the MPC problem is also NP-hard for axis-aligned unit squares and unit disk, and it cannot be approximated by a ratio better than 2. They presented factor-2 approximation algorithms for the MPC problem with unit squares and unit disks. Their algorithms run in linear time if the optimal ply is bounded by a constant.

In this paper we study the MPC problem for general convex shapes. Let C be an arbitrary convex polygon in the plane. The objects in S are translations of C. We present an algorithm that finds a subset S' of S, with ply at most 2ℓ, that covers all points of P, where ℓ is the optimal ply. In other word, we present a 2-approximation algorithm for the problem instance (P, S). The following theorem summarizes our result.

Theorem 1 *There exists a 2-approximation algorithm for the minimum ply covering of points with fixed-size convex polygons that runs in polynomial-time when the optimal objective value is bounded by a constant.*

Our algorithm is a generalization of the algorithm of Biedl et al. [2]. We first give an overview of their algorithm, and then we show how to extend it to work for any convex shape.

*School of Computer Science, University of Windsor, ahmad.biniaz@gmail.com

†School of Computer Science, University of Windsor, lin12v@uwinsdor.ca

2 Algorithm of Biedl, Biniaz, and Lubiw

We describe their 2-approximation algorithm for unit squares. The main idea of their unit disks algorithm is similar to that of unit squares. Let S be a set of axis-aligned squares of side length 1. Recall that P is a set of points in the plane. To solve the instance (P, S), the algorithm partitions the plane into horizontal slabs of height 2. Let $H_1, H_2, \ldots$ denote these slabs from bottom to top. Let P_j be the points of P in H_j and let S_j be the squares of S that intersect H_j, as in Figure 2. Every square intersects at most two (neighboring) slabs and thus it can appear in at most two sets S_j. The idea is to first solve the MPC problem for each slab H_j optimally, i.e., to solve (P_j, S_j) instances. Let S'_j be an optimal solution for slab H_j. Then take S' as the union of all solutions S'_j. The set S' is a 2-approximate solution for the original problem because every square can appear in at most two S'_j.

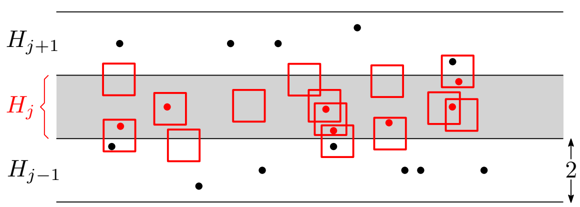

Figure 2: Partitioning the plane into slabs. Red points belong to P_j and red squares belong to S_j.

Assume that the optimal ply is at most ℓ. To solve the (P_j, S_j) instance, partition H_j into vertical strips by vertical lines through the leftmost and rightmost points of all squares.[1] Let t_1, t_2, ..., t_k denote these strips from left to right. The following observation plays an important role in the design of the algorithm: *if S^*_j is a solution of (P_j, S_j) with ply at most ℓ, then each strip t_i is intersected by at most 3ℓ squares of S^*_j.*[2] This observation is used to construct a directed acyclic graph G such that any path from the source to the destination in G corresponds to a solution of (P_j, S_j). The graph G is constructed as follows.

For every strip t_i, define a vertex set V_i as follows. Consider every subset $Q \subseteq S_j$ containing at most 3ℓ squares that intersect t_i. Add a vertex $v_i(Q)$ to V_i if (i) the ply of Q is at most ℓ, and (ii) the squares in Q cover all points of P_j that lie in t_i. Notice that no square intersects the strips t_1 and t_k. Thus the set V_1 has exactly one vertex $v_1(\emptyset)$ which is called the "source", and the set V_k has exactly one vertex $v_k(\emptyset)$ which is called the "sink". The vertex set of G is the union of all vertex sets V_i.

[1] In case of squares, the vertical line through the leftmost (resp. rightmost) point is essentially the line through the left (resp. right) side of square.

[2] This number is at most 8ℓ for unit disks [2].

The edges of G are defined base on the following observation. Imagine we scan an optimal solution S^*_j from left to right. While moving from a strip t_i to t_{i+1} either one square stops at their boundary, or one square starts at their boundary, or the squares that intersect t_{i+1} are the same as those intersect t_i. Based on this, we add a directed edge from every vertex $v_i(Q)$ in V_i to every vertex $v_{i+1}(Q')$ in V_{i+1} if one of the following conditions hold

1. $Q' = Q$ as in Figure 3(a), or
2. $Q' = Q \setminus \{q\}$, where q is the square whose right side is on the left boundary of t_{i+1} as in Figure 3(b), or
3. $Q' = Q \cup \{q\}$, where q is the square whose left side is on the left boundary of t_{i+1} as in Figure 3(c).

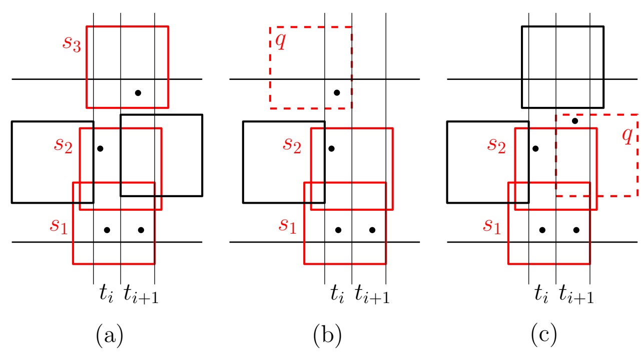

Figure 3: Constructing edges of G where (a) $Q = Q' = \{s_1, s_2, s_3\}$ (b) $Q = \{s_1, s_2, q\}$ and $Q' = \{s_1, s_2\}$ (c) $Q = \{s_1, s_2\}$ and $Q' = \{s_1, s_2, q\}$.

Let δ be any path from the source $v_1(\emptyset)$ to the sink $v_k(\emptyset)$. The union of all sets Q corresponding to the vertices of δ is a solution of (P_j, S_j). The running time of this algorithm for one slab H_j is $O\left((\ell + |P_j|) \cdot |S_j|^{3\ell+1}\right)$, and for all slabs is $O\left((\ell + n) \cdot (2m)^{3\ell+1}\right)$ where $n = |P|$ and $m = |S|$; see section 3.1 for more details. If ℓ is bounded by a constant then the running time is polynomial. The main ingredient to achieve this running time is the fact that the number of squares of any optimal solution S^*_j that intersect any strip t_i is bounded by a constant multiple of ℓ. We are going to obtain a similar fact for all convex shapes, and then extend the algorithm to work for any convex shape.

3 Minimum ply covering with convex shapes

Let P be a set of n points in the plane, and let S be a set of m objects that are translations of the same convex polygon C, as in Figure 1. We show how to find a subset S' of S, with ply at most 2ℓ, that covers all points of P, where ℓ is the optimal ply. In other words, we present a 2-approximation algorithm for the problem instance

(P, S). The algorithm takes polynomial time when ℓ is a constant.

Before proceeding to the algorithm we introduce some terminology. A pair of rectangles (r, R) is called *homothetic* if they are parallel and have the same aspect ratio (r and R need not be axis-parallel). A homothetic pair (r, R) is an *approximating pair* for C if $r \subseteq C \subseteq R$, that is, r is enclosed in C and C is enclosed in R; see Figure 4. Let $\lambda(r, R)$ be the smallest ratio of the length of R to the length of r, over all convex shapes. Pólya and Szegö [12] showed that for every convex shape there exists an approximating pair (r, R) with $\lambda(r, R) \leqslant 3$. Schwarzkopf et al. [13] and Lassak [10] improved this upper bound to 2.[3] For any convex polygon C, an approximating pair of ratio at most 2, can be computed in $O(\log^2 |C|)$ time if the vertices of C are given as a sorted array [13]. The upper bound 2 for $\lambda(r, R)$ is the best possible because for a triangle the length of smallest enclosing rectangle is at least 2 times the length of its largest enclosed homothetic rectangle.

Let (r, R) be an approximating pair for our convex polygon C such that $\lambda(r, R) \leqslant 2$. For simplicity we assume that $\lambda(r, R) = 2$ (this can be achieved by enlarging R or by shrinking r). After a suitable rotation and scaling assume that the longer side of R is vertical and its length is 1. Let α denote the length of the smaller side of R after scaling, as in Figure 4. In this setting the side lengths of r are $1/2$ and $\alpha/2$.

As before, we partition the plane into horizontal slabs of height 2, and then for every slab H_j we solve the problem instance (P_j, S_j) optimally. To solve this instance we partition H_j into vertical strips $t_1, \dots, t_k$ by vertical lines through the leftmost and the rightmost points of every object in S_j. To construct the corresponding directed acyclic graph G we use the following lemma. This lemma, which is our main technical result, uses the concept of approximating pair of rectangles.

Lemma 2 *Let $S_j^* \subseteq S_j$ be any solution with ply at most ℓ for the problem instance (P_j, S_j). Then any strip t_i is intersected by at most 12ℓ objects in S_j^*.*

Proof. After a suitable translation assume that H_j has y-range $[0, 2]$, and that the y-axis lies in t_i, as in Figure 4. Consider any object C in S_j, and let (r, R) be its approximating pair. We refer to the bottom-left corner of r as the *representative point* of C, and denote it by c. Let h and w be the distances from c to the bottom and left sides of R, respectively. Then the distances from c to the top and right sides of R are $1 - h$ and $\alpha - w$, as in Figure 4. Consider the rectangle F with bottom-left corner $(w - \alpha, h - 1)$ and top-right corner $(w, 2 + h)$. The length of F is 3 and its width is α. Cover F by 12 instances of r, say $r_1, r_2, \dots, r_{12}$. Denote the top-right corner of each r_k by p_k; these corners are marked by green points in Figure 4.

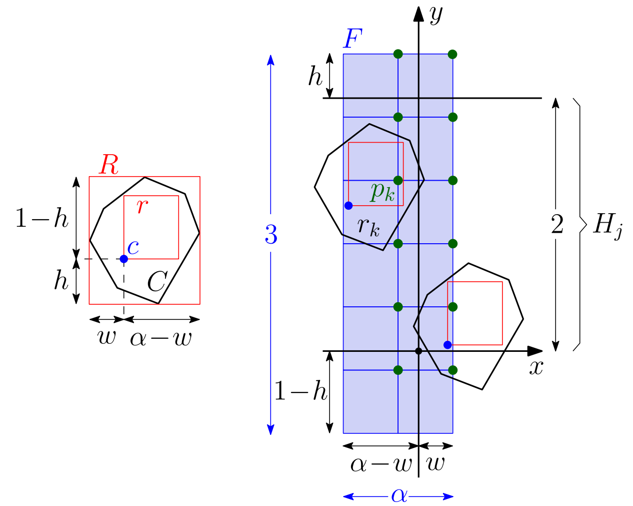

Figure 4: Illustration of the proof of Lemma 2.

Assume that C intersects the strip t_i. Then C intersects the y-axis because vertical strips are defined by vertical lines through leftmost and rightmost points of objects in S_j. In this setting, our definition of h, w, and F imply that the representative c of C must lie in rectangle F. Since F is covered by instances of r, the point c must lie in one of these instances, say r_k. In this case the enclosed rectangle r of C contains p_k, and so does C. Thus, each object in S_j that intersects t_i contains at least one of the points $p_1, \dots, p_{12}$. Since S_j^* has ply at most ℓ, each point p_k lies in at most ℓ objects of S_j^*. Therefore, at most 12ℓ objects of S_j^* intersect t_i. □

We use Lemma 2 to construct a directed acyclic graph G, analogous to that of [2]. The main difference between the two constructions is in the definition for vertex set V_i of each strip t_i: for every subset Q of at most 12ℓ squares that intersect t_i we introduce a vertex $v_i(Q)$ if (i) the ply of Q is at most ℓ, and (ii) its squares cover all points in t_i. The edges of G are defined as before. Any path from the source to the sink in G corresponds to a solution of (P_j, S_j)—this claim, which is proved in [2] for squares and circles, holds for any convex shape and in particular for C. This is the end of the algorithm and its correctness proof.

3.1 Time complexity

The running time analysis is analogous to that of [2] for squares, and thus we keep it short. Set $n_j = |P_j|$ and $m_j = |S_j|$. Then the number of strips is $k = 2m_j + 1$. The number of vertices in every set V_i is $O\left(m_j^{12\ell}\right)$. Therefore the total number of vertices of G is at most $k \cdot O\left(m_j^{12\ell}\right) = O\left(m_j^{12\ell+1}\right)$. Since every vertex has at most three outgoing edges, the number of edges of G is also

[3] A similar ratio is also obtained for pairs of ellipses that approximate convex shapes [8].

$O\left(m_j^{12\ell+1}\right)$. By an initial sorting of the points of P_j and the objects of S_j with respect to the y-axis, conditions (i) and (ii) can be verified in $O\left(|C| \cdot (\ell + n_j)\right)$ time for each subset Q, where $|C|$ is the number of vertices of C. Therefore, it takes $O(|C| \cdot (\ell + n_j) \cdot m_j^{12\ell+1})$ time to construct G. A path from the source to the sink in G can be found in time linear in the size of G. Thus, the total running time to solve the problem instance (P_j, S_j) is $O(|C| \cdot (\ell + n_j) \cdot m_j^{12\ell+1})$. Since every point of P belongs to one slab and every object of S belongs to at most two slabs, the running time of the entire algorithm—for all slabs—is $O(|C| \cdot (\ell + n) \cdot (2m)^{12\ell+1})$, which is polynomial when ℓ is bounded by a constant.

4 Conclusion

We generalized the 2-approximation algorithm of Biedl et al. [2] for the MPC problem to work for any convex shape. A natural question is to verify if there are polynomial-time O(1)-approximation algorithms for the MPC problem when the objective value is not necessarily a constant.

References

[1] M. Basappa, R. Acharyya, and G. K. Das. Unit disk cover problem in 2D. *Journal of Discrete Algorithms*, 33:193–201, 2015.

[2] T. Biedl, A. Biniaz, and A. Lubiw. Minimum ply covering of points with disks and squares. In *Proceedings of the 31st Canadian Conference on Computational Geometry (CCCG)*, pages 226–235, 2019.

[3] A. Biniaz, P. Liu, A. Maheshwari, and M. H. M. Smid. Approximation algorithms for the unit disk cover problem in 2D and 3D. *Comput. Geom.*, 60:8–18, 2017. Also in CCCG'15.

[4] P. Carmi, M. J. Katz, and N. Lev-Tov. Covering points by unit disks of fixed location. In *Proceedings of the 18th International Symposium on Algorithms and Computation (ISAAC)*, pages 644–655, 2007.

[5] G. K. Das, R. Fraser, A. López-Ortiz, and B. G. Nickerson. On the discrete unit disk cover problem. *Int. J. Comput. Geometry Appl.*, 22(5):407–420, 2012. Also in WALCOM'11.

[6] D. Eppstein and M. T. Goodrich. Studying (non-planar) road networks through an algorithmic lens. In *Proceedings of the 16th ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems, ACM-GIS*, 2008.

[7] T. Erlebach and E. J. van Leeuwen. Approximating geometric coverage problems. In *Proceedings of the 19th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1267–1276, 2008.

[8] F. John. Extremum problems with inequalities as subsidiary conditions. *Studies and Essays Presented to R. Courant on his 60th Birthday, January 8, 1948, Interscience, New York*, pages 187–204, 1948.

[9] F. Kuhn, P. von Rickenbach, R. Wattenhofer, E. Welzl, and A. Zollinger. Interference in cellular networks: The minimum membership set cover problem. In *Proceedings of the 11th International Computing and Combinatorics Conference (COCOON)*, pages 188–198, 2005.

[10] M. Lassak. Approximation of convex bodies by rectangles. *Geometriae Dedicata*, 47:111–117, 1993.

[11] N. H. Mustafa and S. Ray. Improved results on geometric hitting set problems. *Discrete & Computational Geometry*, 44(4):883–895, 2010.

[12] G. Pólya and G. Szegő. *Isoperimetric Inequalities in Mathematical Physics*. Annals of Mathematics Studies 27, Princeton University Press, 1951.

[13] O. Schwarzkopf, U. Fuchs, G. Rote, and E. Welzl. Approximation of convex figures by pairs of rectangles. *Comput. Geom.*, 10(2):77–87, 1998. Also in STACS'90.