The Class Cover Problem with Boxes

S. Bereg* S. Cabello† J. M. Díaz-Báñez‡ P. Pérez-Lantero§ C. Seara¶
I. Ventura‡

January 21, 2012

Abstract

In this paper we study the following problem: Given sets R and B of r red and b blue points respectively in the plane, find a minimum-cardinality set $\mathcal{H}$ of axis-aligned rectangles (boxes) so that every point in B is covered by at least one rectangle of $\mathcal{H}$, and no rectangle of $\mathcal{H}$ contains a point of R. We prove the NP-hardness of the stated problem, and give either exact or approximate algorithms depending on the type of rectangles considered. If the covering boxes are vertical or horizontal strips we give an efficient algorithm that runs in $O(r \log r + b \log b + \sqrt{r}b)$ time. For covering with oriented half-strips an optimal $O((r+b)\log(\min\{r,b\}))$-time algorithm is shown. We prove that the problem remains NP-hard if the covering boxes are half-strips oriented in any of the four orientations, and show that there exists an $O(1)$-approximation algorithm. We also give an NP-hardness proof if the covering boxes are squares. In this situation, we show that there exists an $O(1)$-approximation algorithm.

1 Introduction

Let R and B be sets of red and blue points respectively in the plane. Let $S = R \cup B$, $|R| = r$, $|B| = b$, and $n = r + b$. We say that R and B are the red and blue classes, respectively, and that S is a bicolored point set. The x- and y-coordinates of the point p are denoted by x_p and y_p, respectively. Given $X, Y \subset \mathbb{R}^2$, we say that X is Y-empty if X does not contain elements from Y.

A classical problem in Data Mining and classification problems is the Class Cover problem [10, 13, 22] which is as follows: given a bicolored set of points $S = R \cup B$ find a minimum-cardinality set of R-empty balls which covers the blue class (i.e., every point in B is contained in at least one of the balls) and with the constraint that balls are centered at blue points. Cannon and Cowen [10]

*Department of Computer Science, University of Texas at Dallas, USA. Partially supported by project FEDER MEC MTM2009-08652. besp@utdallas.edu.

†Department of Mathematics, IMFM, and Department of Mathematics, FMF, University of Ljubljana, Slovenia. Partially supported by the Slovenian Research Agency, program P1-0297. sergio.cabello@fmf.uni-lj.si.

‡Departamento de Matemática Aplicada II, Universidad de Sevilla, Spain. Partially supported by project FEDER MEC MTM2009-08652 and ESF EUROCORES programme EuroGIGA - ComPoSe IP04 - MICINN Project EUI-EURC-2011-4306. {dbanez,iventura}@us.es.

§Escuela de Ingeniería Civil en Informática, Universidad de Valparaíso, Chile. Partially supported by project FEDER MEC MTM2009-08652 and grant FONDECYT 11110069. pablo.perez@uv.cl

¶Departament de Matemàtica Aplicada II, Universitat Politècnica de Catalunya, Spain. Supported by projects MTM2009-07242 and Gen. Cat. DGR2009GR1040. carlos.seara@upc.edu.

showed that the Class Cover problem using balls is NP-hard in general, and presented an $(1+\ln n)$-approximation algorithm for general metric spaces. For points in $\mathbb{R}^d$ with the Euclidean norm, they gave a polynomial-time approximation scheme (PTAS).

One of the basic objectives in Data Mining is to identify (classify) members between two different classes of data [17]. By solving the Class Cover problem, a simple classifier can be stated; see [22].

In this paper we study a non-constrained version of the Class Cover problem in the plane, called the Boxes Class Cover problem, in which axis-aligned rectangles (i.e. boxes) are considered as the covering objects (Figure 1 a)). Our problem can be formulated as follows:

> The Boxes Class Cover problem (BCC problem): *Given the set $S = R \cup B$, find a minimum-cardinality set $\mathcal{H}$ of R-empty open boxes such that every point in B is covered by at least one box of $\mathcal{H}$.*

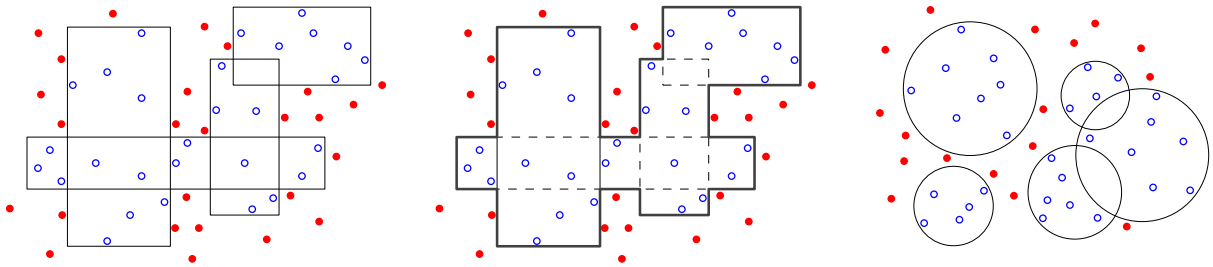

Figure 1: a) Covering the blue class with boxes. b) A solution to the BCC problem induces a rectilinear polygon separating B from R. c) Covering the blue class with disks.

The problem of covering with disks (instead of boxes), not necessarily centered at blue points, is similar to the BCC problem (Figure 1 c)). This version of the Class Cover problem is NP-hard [8], and as we will see in Section 4, it admits a constant-factor approximation algorithm using the techniques of [9, 24].

General position (i.e. no two points are in the same vertical or horizontal line) is not assumed in this paper. It is not hard to see that if a point set is perturbed a bit to be in general position, then we might obtain a different solution to the BCC problem. Thus, we assume by default that points from S may have equal coordinates. Consequently, we will use the lexicographic order, that is, each time we sort points by x-coordinate, ties are broken by using the y-coordinate. All boxes considered in this paper are axis-parallel and open, unless explicitly stated otherwise.

Another motivation for the BCC problem is the so-called Red-Blue Geometric Separation problem [25], where the goal is to compute a simple polygon with fewest vertices as possible separating the red points from the blue points. This problem is motivated by applications in scientific computation, visualization and computer graphics [1]. A solution to the BCC problem provides a geometric separation between the two classes with rectilinear polygons (Figure 1 b)). Indeed, the use of rectangles is usual in the description of a point set [3, 20].

Our contributions. (1) We prove the NP-hardness of the BCC problem by a reduction from the Rectilinear Polygon Covering problem [12, 23]. We present an algorithm that runs in $b \cdot r^{O(\min\{r,b\})}$ time and thus has good performance if r or b is small. We review the theory of ε-nets, which has strong applications to the Class Cover problem [9, 11, 18, 27], and show that our problem admits an $O(\log c)$-approximation, where c is the size of an optimal covering.

(2) Due to the NP-hardness, we study some variants of our problem in which specific types of boxes are used as covering objects. Firstly, if the covering rectangles are axis-parallel strips we prove that the problem is polynomially solvable and give an exact algorithm running in $O(r \log r + b \log b + \sqrt{r}b)$ time. If the boxes are half-strips oriented in the same direction, we present an algorithm that solves the problem in $O((r+b)\log(\min\{r,b\}))$ time. However, if the covering boxes are half-strips in any of the four possible orientations, we prove that the problem remains NP-hard by a reduction from the 3-SAT problem [16]. Moreover, using results from Clarkson and Varadarajan [11] we show that in this case there exists an $O(1)$-approximation algorithm.

(3) We prove that the version in which the covering boxes are axis-aligned squares is NP-hard by a reduction from the problem of covering a rectilinear polygon with holes, represented as a zero-one matrix, with the minimum number of squares [6], and show the existence of an $O(1)$-approximation algorithm.

Outline of the paper. In Section 2 we state a first approach to our problem. In Section 3 we prove that the BCC problem is NP-hard. In Section 4 we review related results concerning range spaces and ε-nets, most of which are relevant to give approximation algorithms to our problem. In Section 5 we study the BCC problem when we restrict the boxes to strips or half-strips. In Section 6 we consider the version of the BCC problem in which the boxes are axis-aligned squares, and we prove its NP-hardness. Finally, in Section 7, we state the conclusions and further research.

2 A simple approach

Observe that any solution $\mathcal{H} = \{H_1, H_2, \ldots, H_k\}$ of the BCC problem is a cover of B, and we can expand every $H_i \in \mathcal{H}$ so that the sides of H_i pass through red points or reach infinity. From this observation, we can consider the set of *maximal* boxes (they cannot be expanded) $\mathcal{H}^*$ of all the R-empty open boxes whose sides pass through red points or are at infinity. Thus, any solution of the BCC problem will be a subset of $\mathcal{H}^*$. Such types of boxes are depicted in Figure 2, up to symmetry. It can be seen that $|\mathcal{H}^*|$ is $O(r^2)$ and also that this bound is tight in the worst case [2, 5, 7].

Figure 2: Boxes in $\mathcal{H}^*$: a) rectangle, b) half-strip, c) strip, d) quadrant, e) half-plane.

Lemma 2.1 *The number of boxes in an optimal solution to the BCC problem is upper bounded by* $\min\{2r+2, b\}$. *Furthermore, this bound is tight.*

Proof. Let $\mathcal{H}$ be an optimal solution of the BCC problem. Since every blue point is covered by a box from $\mathcal{H}$ then $|\mathcal{H}| \leq b$. The equality holds when the elements of S are on a line ℓ and their colors alternate along ℓ. We now prove $|\mathcal{H}| \leq 2r+2$ for any set $S = R \cup B$. For given points $p, q \in R$, let H_p^- (resp. H_p^+) be the maximum-height box of $\mathcal{H}^*$ whose top (resp. bottom) side contains p, and S_{pq} be the vertical strip containing both p and q on its boundary. Associate with each red point p the following set of boxes:

$$A_p = \begin{cases} \{S_{pq}\} & \text{if there exists } q \in R \text{ such that } x_p < x_q,\ y_p = y_q, \text{ and } S_{pq} \in \mathcal{H}^* \\ \{H_p^-, H_p^+\} & \text{otherwise.} \end{cases}$$

Let $\mathcal{W} = \left(\bigcup_{p \in R} A_p\right) \cup \{H_1, H_2\}$, where H_1 is the half-plane in $\mathcal{H}^*$ with right boundary, and H_2 the one with left boundary. It is not hard to see that $\mathcal{W}$ has at most $2r+2$ boxes and covers B. In fact, $\mathcal{W}$ covers $\mathbb{R}^2 \setminus R$. For the tightness of this bound, consider the configuration of points depicted in Figure 3. Notice that there are $2r+2$ groups of blue points, each located on a vertical or horizontal line passing through a red point, so that every two blue points belonging to different groups cannot be covered by the same R-empty box. □

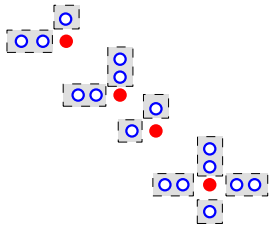

Figure 3: A case in which exactly $2r+2$ R-empty boxes are needed to cover B.

The above discussion lets us design the following exponential algorithm to report an exact solution for the BCC problem: First compute the set $\mathcal{H}^*$ in $O(r^2)$ time, and after that test all subsets of $\mathcal{H}^*$ of size $1, \ldots, \min\{2r+2, b\}$ in this order until finding a covering of B. There are $(O(r^2))^{\min\{2r+2,b\}} = r^{O(\min\{r,b\})}$ subsets of $\mathcal{H}^*$ to be tested, and the test of each subset can be done in $O(b \cdot \min\{2r+2, b\}) = O(\min\{rb, b^2\})$ time. The overall time complexity is $r^{O(\min\{r,b\})} \cdot \min\{rb, b^2\} = b \cdot r^{O(\min\{r,b\})}$, which is exponential in general. However, if r or b is $O(1)$, then it is polynomial.

3 Hardness

We prove that the BCC problem is NP-hard by using a reduction from the Rectilinear Polygon Covering problem (RPC problem) which is as follows: *Given a rectilinear polygon P, find a minimum cardinality set of closed boxes whose union is exactly P* (Figure 4 a), b)). For a general class of rectilinear polygons with holes the RPC problem is NP-hard [23], and it remains NP-hard for polygons without holes [12].

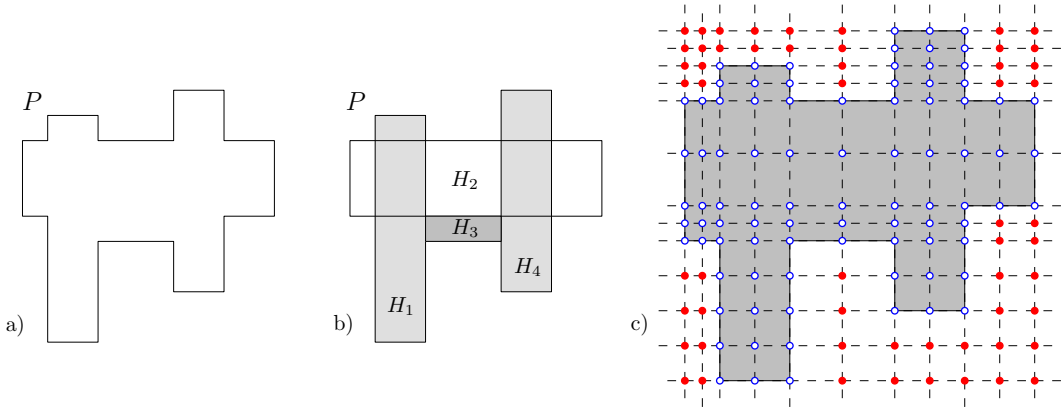

Figure 4: a) A rectilinear polygon P. b) An optimal covering of P with four rectangles. c) The reduction from the RPC problem to the BCC problem

Theorem 3.1 *The BCC problem is NP-hard.*

Proof. Let P be an instance of the RPC problem. Let A_1 be the set of all distinct axis-parallel lines containing an edge of P. For every two consecutive vertical (resp. horizontal) lines in A_1, draw the vertical (resp. horizontal) mid line between them. Let A_2 be the set of these additional lines. Let G be the grid defined by $A_1 \cup A_2$. Put a red (resp. blue) point at each vertex of $G \setminus P$ (resp. $G \cap P$) (Figure 4 c)).

Let S be the above set of red and blue points. Clearly, any optimal covering of P can be scaled up slightly to obtain an optimal covering for the BCC problem on S with the same cardinality (because it covers the edges and the interior of P). Conversely, any optimal covering $\mathcal{H}$ for the BCC problem on S can be adjusted to be an optimal covering for P with the same cardinality: Let $\mathcal{H} = \{H_1, H_2, \ldots, H_k\}$ be an optimal covering for the BCC problem on S. We assume that each $H_i \in \mathcal{H}$ is maximal, i.e., it cannot be expanded in order to contain more blue points. Let H'_i, $1 \leq i \leq k$, be the smallest closed bounding box of $H_i \cap B$. If some H'_i is not contained in P, then it must contain at least one cell of G not contained in P with at least one red vertex, say u, and then H_i covers u, a contradiction. To verify $\mathcal{H}' = \bigcup_{i=1}^{k} H'_i$ covers P we proceed as follows: Let c be a cell of G contained in P. By construction, it holds that: (i) c has exactly two adjacent edges on lines of A_1 and two adjacent edges on lines of A_2, and (ii) any maximal box $H_i \in \mathcal{H}$ covering a blue vertex v of c whose two edges lie on lines of A_1, covers c. Hence, $\mathcal{H}'$ is an optimal covering of P. □

Remark. Let $G_S = (V, E)$ be the graph in which V is equal to B, and there is an edge in E between two blue points p and q if and only if the minimum closed box containing both p and q is R-empty. The blue points covered by an R-empty box are pairwise adjacent and form a clique in G_S. Conversely, the smallest closed bounding box of the points of a clique in G_S is R-empty, and thus there exists an R-empty box covering them. Therefore, the BCC problem is equivalent to finding a minimum clique partition in G_S [16]. The Partition Into Cliques problem is strongly NP-complete [16], and the NP-hardness of the BCC problem implies that it remains NP-complete if the input graph is a graph G_S, where S is a bicolored point set.

4 Approximation algorithms

A finite[1] range space $(X, \mathcal{R})$ is a pair consisting of an underlying finite set X of objects and a finite collection $\mathcal{R}$ of subsets of X called ranges. Given the (primal) range space $(X, \mathcal{R})$, its dual range space is $(\mathcal{R}, X^*)$ where $X^* = \{\mathcal{R}_x \mid x \in X\}$ and $\mathcal{R}_x$ is the set of all ranges in $\mathcal{R}$ that contains x [9].

Given a range space $(X, \mathcal{R})$, the Set Cover problem asks for the minimum-cardinality subset of $\mathcal{R}$ that covers X [16]. The dual of the Set Cover problem is the Hitting Set problem: to find a minimum subset $P \subseteq X$ such that P intersects with each range in $\mathcal{R}$ [16]. A set cover in the primal range space is a hitting set in its dual, and vice versa. The Set Cover problem is NP-hard and the best known approximation factor of a polynomial-time algorithm is $(1 + \ln |X|)$ [15, 16]. The algorithm follows the greedy approach: while there are elements in X not covered, add to the solution the set of $\mathcal{R}$ that covers the maximum number of non-covered elements in X.

The BCC problem is an instance of the Set Cover problem in the range space $(B, \mathcal{H}^*)$. The greedy approach above gives the same logarithmic factor of approximation for the BCC problem, even if we modify its definition by restricting the covering boxes to axis-aligned squares (Figure 5). As a consequence, we get the following result:

[1] A range space can be infinite, but for the purpose of our problem it will be finite.

Statement 4.1 *The BCC problem has an $O(\log b)$-approximation algorithm if we cover with either boxes or axis-aligned squares.*

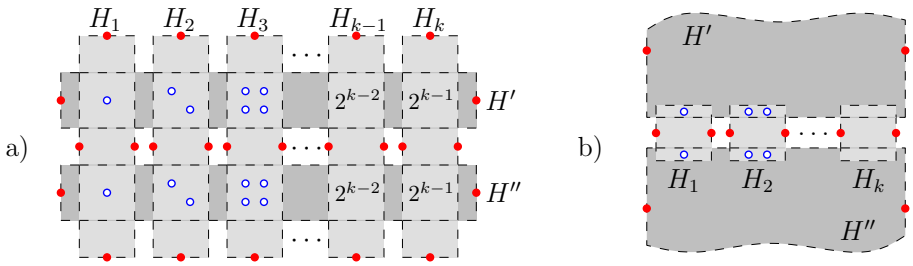

Figure 5: The greedy method gives a logarithmic factor of approximation for both boxes and squares. In a) (resp. b)), each of the intersections of the boxes (resp. squares) H' and H'' with the box (resp. square) H_i $(1 \leq i \leq k)$ contains 2^{i-1} blue points. The greedy method reports $\{H_1, H_2, \ldots, H_k\}$ instead of $\{H', H''\}$.

We now show there exists an approximation algorithm with a smaller approximation ratio, which uses the so-called ε-nets [18] and the VC-dimension [27] of $(X, \mathcal{R})$ and its dual.

In terms of our problem, an ε-net, $0 \leq \varepsilon \leq 1$, is a subset $B' \subseteq B$ such that any box in $\mathcal{H}^*$ containing $\varepsilon|B|$ points, covers an element of B'. In the dual range space, an ε-net is a subset $H \subseteq \mathcal{H}^*$ covering all points p of B such that p is covered by at least $\varepsilon|\mathcal{H}^*|$ boxes of $\mathcal{H}^*$.

The VC-dimension of $(X, \mathcal{R})$ is the maximum cardinality of a subset $Y \subseteq X$ such that any subset of Y is the intersection of Y with some range in $\mathcal{R}$. If the VC-dimension of the primal space is d, then the VC-dimension of the dual range space is at most 2^{d+1} [9, 27]. It is not hard to see that any subset $P \subseteq B$ with at least five points has a subset $P' \subset P$ that cannot be separated with a box in $\mathcal{H}^*$ from $P \setminus P'$. Then the VC-dimension of our range space $(B, \mathcal{H}^*)$ is at most four and is thus constant.

Brönnimann and Goodrich [9] and Even et al. [14] gave techniques using ε-nets to find approximate solutions of the Hitting Set problem for range spaces. The technique proposed in [14] is based on LP-relaxation, and can be interpreted as a simplification of the one of Brönnimann and Goodrich [9]. We recall only the machinery of Brönnimann and Goodrich [9] which works for range spaces of finite VC-dimension. It reports a hitting set whose size is within a factor of $O(\log c)$ from the optimal size c, and is based on the fact that, for every range space with finite VC-dimension d, there exists an ε-net of size $O(\frac{d}{\varepsilon} \log \frac{d}{\varepsilon})$ [18]. In general, if the range space has a constant VC-dimension, and there exists an ε-net of size $O(\frac{1}{\varepsilon}\varphi(\frac{1}{\varepsilon}))$, their method computes a hitting set of size $O(\varphi(c)c)$, where c is the size of an optimal set. Therefore, since both our range space $(B, \mathcal{H}^*)$ and its dual have a constant VC-dimension, Brönnimann and Goodrich's technique can be applied to obtain in the dual range space a hitting set of size at most $O(\log c)$ times the optimal size c, which induces a solution $\mathcal{H}$ (a set cover) of the BCC problem with the same size. Thus we obtain the following result:

Statement 4.2 *The BCC problem has an $O(\log c)$-approximation algorithm, where c is the size of the optimal covering.*

Finding ε-nets of size $o(\frac{1}{\varepsilon} \log \frac{1}{\varepsilon})$ for the dual of the range space $(B, \mathcal{H}^*)$ seems to be a challenge [5]. With such ε-nets we would obtain a smaller approximation factor for the BCC problem. Recently, Pach and Tardos [26] have proved that there exist dual ranges spaces of VC-dimension 2, induced by points and box ranges, whose sizes are $\Omega(\frac{1}{\varepsilon} \log \frac{1}{\varepsilon})$.

Matoušek et al. [24] proved the existence of ε-nets of size $O(\frac{1}{\varepsilon})$ for the dual of any range space consisting of points and disk ranges. Due to this, techniques from Brönnimann and Goodrich [9] provide an $O(1)$-approximation for the Class Cover problem with disks. Clarkson and Varadarajan [11] showed that if the geometric range space[2] $(X, \mathcal{R})$ has the property that, given a any subset $R' \subseteq \mathcal{R}$ and a nondecreasing function $f(\cdot)$, there is a decomposition of the complement of the union of the elements of R' into an expected number of at most $f(|R'|)$ regions, then a cover of size $O(f(c))$ can be found in polynomial time, where c is the size of an optimal cover. This result is based on the fact that, with the above conditions, there are ε-nets of size $O(f(\frac{1}{\varepsilon}))$ for the dual range space [11, Theorem 2.2]. If $\mathcal{R}$ is a family of pseudo-disks[3], then for any subset $R' \subseteq \mathcal{R}$ the trapezoidal decomposition of the complement of the union of the elements of R' has complexity $O(|R'|)$ and thus the dual range space has ε-nets of size $O(\frac{1}{\varepsilon})$ [11]. Since a set of axis-aligned squares is a family of pseudo-disks, by using the techniques in [9, 11], the following result is obtained:

Statement 4.3 *The BCC problem has an $O(1)$-approximation algorithm if the covering boxes are restricted to axis-aligned squares.*

5 Solving particular cases

In this section we study the BCC problem for some special cases. Namely, we consider only certain boxes of $\mathcal{H}^*$ having at most three points on their boundary.

5.1 Covering with horizontal and vertical strips

In this subsection we solve the BCC problem by using only horizontal and vertical strips and also axis-aligned half-planes (which we also call strips for simplicity) as covering objects; see Figure 2 c) and e). We assume that R, B and S are sorted by x- and y-coordinate, which can be achieved in $O(r \log r + b \log b)$ time. Then, the strips of $\mathcal{H}^*$ can be computed in linear time.

There exists a solution for the BCC problem if and only if every blue point can be covered by an axis-parallel line avoiding red points. This can be tested in linear time. Suppose that the BCC problem has a solution. If a blue point and a red point lie on the same vertical (resp. horizontal) line then the blue point can be covered by only one strip in $\mathcal{H}^*$. We add all such strips to the solution and remove the blue points they cover. This can be done in linear time. Each of the remaining blue points is covered by two strips in $\mathcal{H}^*$. We show how to solve this problem optimally.

Consider the graph $G = (V, E)$ whose set of vertices is the set of strips that cover at least one blue point (Figure 6), and whose set of edges E is defined as follows: put an edge between the strips H_1 and H_2 if and only if $H_1 \cap H_2$ contains a blue point. The graph G is bipartite, has $O(r)$ vertices and $O(b)$ edges, and can be constructed in $O(r + b)$ time.

Since each blue point is covered by exactly two strips, the problem is reduced to finding a minimum mertex cover [16] in G. However, because of König's theorem, the Vertex Cover problem for bipartite graphs is equivalent to the Maximum Matching problem, and thus it can be solved in $O(\sqrt{|V|}|E|) = O(\sqrt{r}b)$ time [19]. Thus, the following result is obtained:

[2] A range space $(X, \mathcal{R})$ is *geometric* if X is a set of geometric objects, generally points, and $\mathcal{R}$ is a set of geometric ranges such as half-spaces, boxes, convex polygons, balls, etc.

[3] A family of Jordan regions (i.e. regions bounded by closed Jordan curves) is a family of pseudo-disks if the boundaries of any pair of regions intersect at most twice.

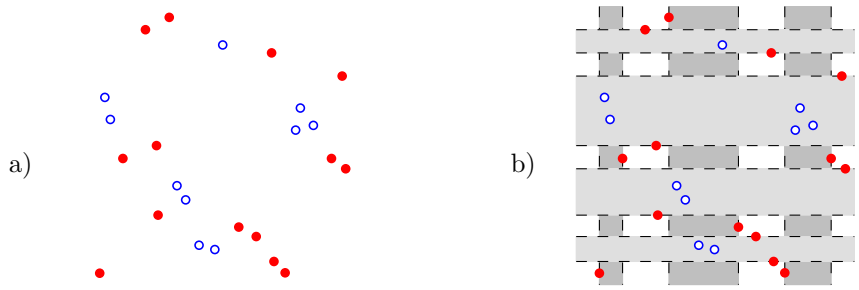

Figure 6: a) A set of red and blue points. b) Strips covering at least one blue point.

Theorem 5.1 *The BCC problem can be solved in* $O(r \log r + b \log b + \sqrt{r}b)$ *time if we use only axis-aligned strips as covering objects.*

5.2 Covering with oriented half-strips

In this subsection we solve the BCC problem by considering only half-strips oriented in a given direction, say top-bottom half-strips. A box of $\mathcal{H}^*$ is a half-strip if it contains at most three points on its boundary (Figure 2 b), c), d), and e)), and is top-bottom if either it contains a red point on its top side or it is a vertical strip. Next, we give an optimal $O((r+b)\log(\min\{r,b\}))$-time algorithm.

Consider the structure of rays that is obtained by drawing a bottom-top red ray starting at each red point as depicted in Figure 7. For a given blue point p, let s_p be the maximum-length horizontal segment passing through p whose interior does not intersect any red ray.

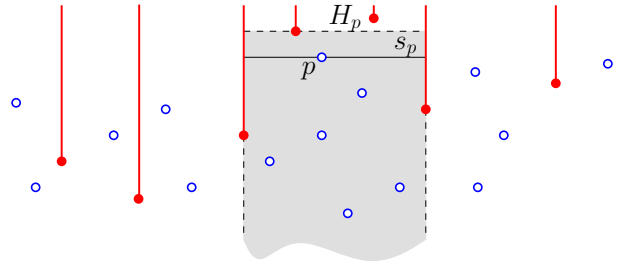

Figure 7: The structure of rays.

Sketch of the algorithm and correctness. Every time, we select the highest blue point p not yet covered, and include in the solution the top-bottom half-strip $H_p \in \mathcal{H}^*$ whose top side is s_p translated upwards until it touches a red point or reaches infinity. In other words, H_p is the top-bottom half-strip in $\mathcal{H}^*$ covering p and the maximum number of other blue points. The algorithm ends when all blue points are covered. The correctness follows from the fact that, if p is a blue point not yet covered with maximum y-coordinate, then H_p is so that, for any other non-covered blue point p' which is not in H_p, p and p' cannot be covered with the same top-bottom half-strip.

Using balanced trees, combined with one-dimensional range search techniques, the algorithm can be done in $O(r \log r + b \log b)$ time. Within the same time complexity the algorithm can decide whether a solution exists.

We now show how to reduce the asymptotic time complexity. Suppose $r < b$. We prune the set of blue points so that obtaining a set of at most $2r+1$ blue points. It is as follows. For every vertical strip between two consecutive red points and for every vertical line containing a red point, we store

only the highest blue point. It suffices to cover only these blue points with the strips generated by the algorithm. Those blue points can be found in $O(b \log r)$ time by using a binary search in the sequence obtained by sorting the elements of R by x-coordinate. Then the above algorithm executes in $O(r \log r)$ time. We overall time complexity is thus $O((r+b)\log r)$. We can proceed analogously if $b \geq r$ in order to reduce the time complexity to $O((r+b)\log b)$ (by pruning red points in the strips defined by blue points). In general, we can choose which variant to apply depending on the minority color, and finally obtain an algorithm running in $O(\min\{(r+b)\log r, (r+b)\log b\}) = O((r+b)\log(\min\{r,b\}))$ time. Thus, the following result is obtained:

Theorem 5.2 *The BCC problem can be solved in $O((r+b)\log(\min\{r,b\}))$ time if we use only half-strips in one direction as covering objects.*

We next show that for $\min\{r,b\} = \Omega(r+b)$ the above algorithm is optimal in the algebraic decision tree model. Given a set $X = \{x_1, \ldots, x_n\}$ of n numbers, denote as $x_{\pi_1} \leq \cdots \leq x_{\pi_n}$ the sorted sequence of these numbers. The maximum gap of X is defined as MAX-GAP$(X) = \max_{1 \leq i < n}\{x_{\pi_{i+1}} - x_{\pi_i}\}$ [21]. Arkin et al. [4] proved that, given a set $X = \{x_1, \ldots, x_n\}$ of n real numbers and a positive real number ε, the problem of deciding whether

$$\text{MAX-GAP}\{x_1, \ldots, x_n, 0, \varepsilon, 2\varepsilon, \ldots, n\varepsilon\} < \varepsilon$$

has an $\Omega(n \log n)$ lower bound in the algebraic decision tree model. (Note that this problem can be solved in linear time with the floor function, which is not an algebraic operation.) By a reduction from this new version of MAX-GAP, we show that our algorithm is optimal.

Theorem 5.3 *The BCC problem has an $\Omega(n \log n)$ lower bound in the algebraic decision tree model if we use only half-strips (or strips) in one direction.*

Proof. Let $X = \{x_1, \ldots, x_n\}$ and $\varepsilon > 0$ be an instance of the above MAX-GAP problem. Assume that $0 \leq x_i \leq n\varepsilon$, for $i = 1, \ldots, n$, because otherwise the max gap would be greater than or equal to ε. We do the following construction: Put red points at the coordinates $(0,0), (\varepsilon, 0), (2\varepsilon, 0), \ldots, (n\varepsilon, 0)$. Let R be the set of these $n+1$ red points. Put blue points at the coordinates $(x_1, 1), (x_2, 1), \ldots, (x_n, 1)$, and let B be the set of these n blue points. In order to have the max gap smaller than ε, each of the open intervals $(0, \varepsilon), (\varepsilon, 2\varepsilon), \ldots, ((n-1)\varepsilon, n\varepsilon)$ has to be pierced by one of the x_i's. Now, solve the BCC problem for R and B with half-strips (or strips) in the top-bottom direction. It follows that MAX-GAP$\{x_1, \ldots, x_n, 0, \varepsilon, 2\varepsilon, \ldots, n\varepsilon\} < \varepsilon$ if and only if the minimum number of covering half-strips (or strips) is exactly n (Figure 8). □

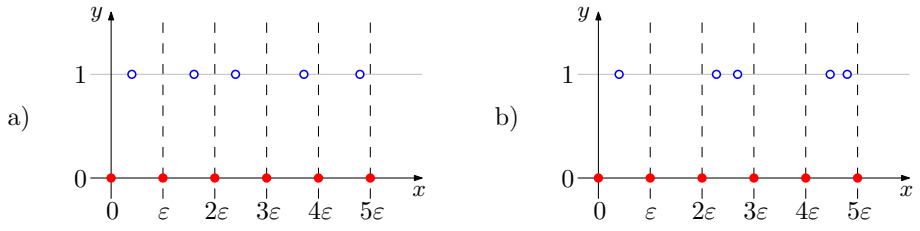

Figure 8: Reduction from the MAX-GAP problem to our problem.

5.3 Covering with half-strips

In this subsection we study the BCC problem when the covering boxes are half-strips oriented in any of the four possible directions. We call this version the Half-Strip Class Cover problem (HSCC problem). First we show that this variant is also NP-hard, and after that we give a constant-factor approximation algorithm due to results in [9, 11].

Notice that a solution to the HSCC problem does not exist if and only if there are two segments with red endpoints, one vertical and one horizontal, such that their intersection is a blue point. This can be checked by using similar arguments as in Subsections 5.1 and 5.2.

Theorem 5.4 *The HSCC problem is NP-hard.*

Proof. To prove the NP-hardness we use a reduction from the 3-SAT problem [16]. An instance of the 3-SAT problem is a logic formula $\mathcal{F}$ of t boolean variables $x_1, \ldots, x_t$ given by m conjunctive clauses $C_1, \ldots, C_m$, where each clause contains exactly three literals (i.e., a variable or its negation). The 3-SAT problem asks for a value assignment to the variables which makes the formula satisfiable, and its NP-hardness is well known [16].

Given $\mathcal{F}$, an instance of the HSCC problem is constructed in the following way. Let α be a set of t pairwise-disjoint vertical strips of equal width such that the i-th strip α_i, from left to right, represents the variable x_i. Similarly, let β be a set of $t+m$ pairwise-disjoint horizontal strips of equal width. The clause C_j is represented by the $(t+j)$-th strip β_{t+j} from bottom to top. Consecutive strips in α and β are well separated. Let δ_i be a mid line partitioning the strip α_i into two equal parts (Figure 9). We say that the part of the interior of α_i that is to the right (resp. to the left) of δ_i is the true (resp. false) part of α_i.

For each variable x_i $(1 \leq i \leq t)$ we put at $\alpha_i \cap \beta_i$ a set V_i of red and blue points as follows (Figure 9). We add red points in the intersections of δ_i and the boundary of β_i; a blue point p in the center of $\alpha_i \cap \beta_i$ (p is on δ_i); two red points q and q' in the interior of β_i such that q is on the left boundary of α_i and $y_q > y_p$, and q' is on the right boundary of α_i and $y_{q'} < y_p$. Moreover, we add two blue points p' and p'' in the interior of $\alpha_i \cap \beta_i$ such that p' is in the false part of α_i and $y_{p'} < y_{q'}$, and p'' is in the true part of α_i and $y_{p''} > y_q$.

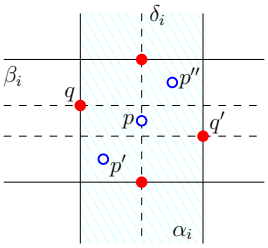

Figure 9: Set of bicolored points V_i for x_i in the reduction from the 3-SAT problem.

For each clause C_j $(1 \leq j \leq m)$ we add a set W_j of bicolored points in the following way. Suppose that C_j involves the variables x_i, x_k, and x_l $(1 \leq i < k < l \leq t)$. Let ℓ_1 and ℓ'_1 (resp. ℓ_2 and ℓ'_2) be two horizontal lines that are close to the top (resp. bottom) boundary of β_{t+j} such that ℓ_1 (resp. ℓ_2) is outside β_{t+j} and ℓ'_1 (resp. ℓ'_2) is inside (Figure 10). Let ℓ_3 and ℓ'_3 be two vertical lines lying outside α_k and such that ℓ_3 and ℓ'_3 are close to the left and right boundaries of α_k, respectively.

Put red points at the intersections of the lines ℓ_1 and ℓ_2 with δ_i, ℓ_3, δ_k, ℓ'_3, δ_l, and the boundaries of α_i, α_k, and α_l. Add three more red points, one on the top boundary of β_{t+j}, to the left of ℓ_3 and

close to ℓ_3; another one between ℓ_3 and the left boundary of α_k, above ℓ'_2 and close to ℓ'_2; and the last one on ℓ'_2 and between the right boundary of α_k and ℓ'_3.

Now we add blue points. Put a blue point at the intersection of ℓ'_1 and ℓ_3, and another one at the intersection of ℓ'_3 and the bottom boundary of β_{t+j}. If x_i is not negated in C_j, then put at the true part of α_i (otherwise, in the false part) two blue points, the first one on ℓ'_1 and the second on the bottom boundary of β_{t+j}. If x_k is not negated in C_j, then put one blue point at the center of the intersection of β_{t+j} and the true part of α_k (otherwise in the false part). Finally, if x_l is not negated in C_j, then put at the true part of α_l (otherwise in the false part) two more blue points, one on the top boundary of β_{t+j} and another one on the bottom boundary.

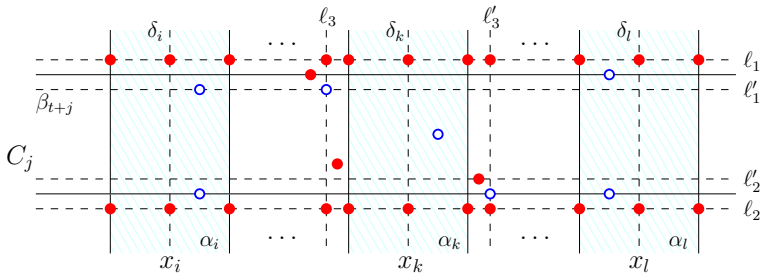

Figure 10: The set W_j of red and blue points for the clause $C_j = (x_i \vee x_k \vee \neg x_l)$.

Let $S = \bigcup_{i=1}^{t} V_i \cup \bigcup_{j=1}^{m} W_j$ be the instance of the HSCC problem. We say that two blue points in S are *independent* if they cannot be covered with the same half-strip. Notice that for each variable x_i the blue points in V_i are independent from the others blue points in S except with those that are in α_i, and also that at least two half-strips are needed to cover them. Moreover, blue points in the false part of α_i are independent from blue points in the true part. There are essentially two ways of covering the blue points in V_i with two half-strips. The first one with a right-left half-strip covering the two lowest blue points in V_i and a vertical strip covering the true part of α_i (Figure 11 a)), and the second one with a vertical strip covering the false part of α_i and a left-right half-strip that covers the upper two blue points of V_i (Figure 11 b)). We say that the first way is a true covering of V_i (i.e., x_i is true), and that the second one is a false covering of V_i (i.e., x_i is false).

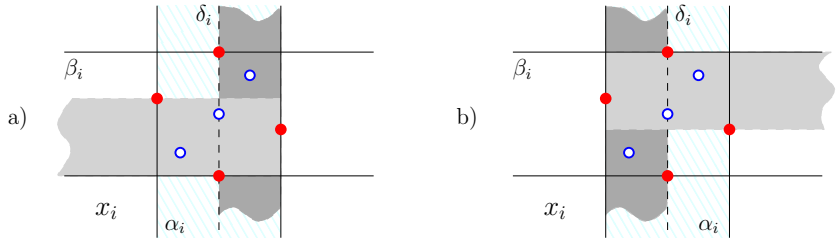

Figure 11: The two ways of optimally covering the blue points associated with a variable x_i. a) x_i is equal to true, b) x_i is equal to false.

For each clause C_j $(1 \leq j \leq m)$ that involves the variables x_i, x_k, and x_l $(1 \leq i < k < l \leq m)$, observe that if at least one variable, say x_i, is such that the covering of V_i covers the blue points in $W_j \cap \alpha_i$ (i.e., the value of x_i, corresponding to the covering of V_i, makes C_j true), then exactly two half-strips are sufficient and needed to cover $W_j \setminus \alpha_i$. Otherwise, exactly three half-strips are sufficient and needed to cover W_j. To see this, note that the blue points in $W_j \setminus (\alpha_i \cup \alpha_k \cup \alpha_l)$ (those in lines ℓ_3 and ℓ'_3; see Figure 10) are independent not only between them but also with all blue points not in W_j. Then at least two half-strips are needed to cover W_j, which are sufficient if

the covering of V_i covers $W_j \cap \alpha_i$. Refer to Figure 12.

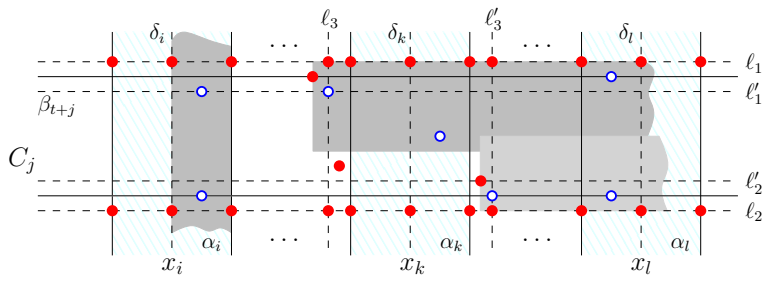

Figure 12: If $W_j \cap \alpha_i$ is covered by the covering of V_i then exactly two half-strips (the horizontal ones) are sufficient and needed to cover $W_j \setminus \alpha_i$.

Let $\mathcal{H}$ be an optimal solution of the HSCC problem. Due to the above observations, we claim that $\mathcal{F}$ is satisfiable if and only if $|\mathcal{H}| = 2t + 2m$. In fact, if $\mathcal{F}$ is satisfiable, then for each variable x_i we cover V_i with a true covering if x_i is true, and otherwise with a false covering. Each clause C_j (with variables x_i, x_k, and x_l) is true, then with two half-strips we can cover the blue points in W_j not covered by the coverings of V_i, V_k, and V_l. We use $2t$ half-strips for the variables and $2m$ for the clauses, thus $2t + 2m$ in total. Inversely, $|\mathcal{H}|$ cannot be less than $2t + 2m$, thus if $|\mathcal{H}|$ is exactly $2t + 2m$, then we must use two half-strips for each variable, and two half-strips for each clause, implying that $\mathcal{F}$ is satisfiable if we assign the value true to each variable x_i, if V_i has a true covering, and the value false otherwise. Hence, the theorem follows. □

Given the NP-hardness of the HSCC problem, we are interested in approximation algorithms. Let $\mathcal{H}_S$ be the set of all half-strips in $\mathcal{H}^*$. By using results from Clarkson and Varadarajan [11], we prove that the dual of the range space $(B, \mathcal{H}_s)$ has ε-nets of size $O(\frac{1}{\varepsilon})$, implying an $O(1)$-approximation algorithm to the HSCC problem.

Theorem 5.5 *There is a polynomial-time $O(1)$-approximation algorithm for the HSCC problem.*

Proof. Let $\mathcal{H}_S$ be the set of all half-strips in $\mathcal{H}^*$, and partition $\mathcal{H}_S$ into the subsets $\mathcal{H}_{S_v}$ and $\mathcal{H}_{S_h}$ of all vertical and horizontal half-strips, respectively. Given $\varepsilon > 0$, the dual of the range space $(B, \mathcal{H}_{S_v})$ has (by results from Clarkson and Varadarajan [11]) an $(\frac{\varepsilon}{2})$-net N_v of size $O(\frac{1}{\varepsilon})$ because $\mathcal{H}_{S_v}$ is a family of pseudo-disks. Analogously, the dual of the range space $(B, \mathcal{H}_{S_h})$ has an $(\frac{\varepsilon}{2})$-net N_h of size $O(\frac{1}{\varepsilon})$. We claim that $N_v \cup N_h$ is an ε-net of size $O(\frac{1}{\varepsilon})$ for the dual of $(B, \mathcal{H}_S)$. In fact, if p is a blue point covered by $\varepsilon|\mathcal{H}_S|$ half-strips, then at least $\frac{\varepsilon}{2}|\mathcal{H}_S|$ of them are either vertical or horizontal. Thus, since N_v and N_h are $(\frac{\varepsilon}{2})$-nets, p is covered by a half-strip in $N_v \cup N_h$. It follows from the results from Brönnimann and Goodrich [9] and Clarkson and Varadarajan [11, Theorem 3.2] that there exists a polynomial-time $O(1)$-approximation algorithm for the HSCC problem. □

6 Covering with squares

In this section we study the variant of the BCC problem in which axis-aligned squares are used, instead of general boxes (rectangles), as covering objects. We call this version the Square Class Cover problem (SCC problem).

Aupperle et al. [6] studied the problem of covering a rectilinear polygon with the minimum number of axis-aligned squares. The input polygons were represented as a bit-map, that is, a zero-one

matrix in which the 1's represent points inside the polygon, and the 0's points outside it. They proved that the problem (equivalent to covering the 1's of the matrix with the minimum number of squares) is NP-hard if the input polygon contains holes. By using a reduction from this problem, we prove that the SCC problem is NP-hard. Before presenting our NP-hardness proof, we state and prove the following useful lemma:

Let $s \subset \mathbb{R}$ be a closed interval. We denote by $\text{left}(s)$ the left endpoint of s. Let t be the largest integer less than or equal to $\text{left}(s)$. We say that s is *lattice* if $\text{left}(s) = t$. Otherwise, we say that we *adjust* s, or that s is *adjusted*, if we shift s so that either $\text{left}(s) = t$ or $\text{left}(s) = t + 1$. Given $X \subset \mathbb{R}$, let $\text{m}(X)$ denote the Lebesgue measure of X.

Lemma 6.1 *Let N be a positive integer number, and I be a finite set of closed intervals so that each of them is contained in the interval $[0, N]$ and has integer length. Let U be the union of the elements of I. If $\text{m}([0, N] \setminus U)$ is less than one, then all non-lattice elements of I can be adjusted in such a way U becomes equal to $[0, N]$.*

Proof. We can perform what follows for $j = 0, 1, \ldots, N-1$: Let $I_j = \{s \in I \mid j < \text{left}(s) < j + 1\}$ and $s_j = \arg\min_{s \in I_j} \text{left}(s)$. If $[j, j + 1] \subset U$ then adjust all intervals $s \in I_j$ so that $\text{left}(s) = j + 1$. Otherwise, adjust both s_j and all intervals $s \in I_j \setminus \{s_j\}$ so that $\text{left}(s_j) = j$ and $\text{left}(s) = j + 1$. Induction can be used to prove for all $j \in \{0, 1, \ldots, N-1\}$ that after the above processing, $[0, j+1] \subseteq U$ and all intervals $s \in I$ such that $\text{left}(s) \leq j$ are lattice . Observe that $\text{m}([0, N] \setminus U)$ never increases after any interval is adjusted. The result thus follows. □

Theorem 6.2 *The SCC problem is NP-hard.*

Proof. Let P be a rectilinear polygon with holes represented in a $N \times N$ zero-one matrix. We reduce P to an instance S of the SCC problem as follows. Let M be an integer number greater than N. We can consider that the vertices of P are lattice points in $[0, N] \times [0, N]$ (Figure 13 a)). We subdivide the square $[0, N] \times [0, N]$ into a regular grid G of cell size $\frac{1}{M}$, and put a blue (resp. red) point at each vertex of G that is in the interior (resp. boundary) of P (Figure 13 b)). Let S be the resulting bicolored point set.

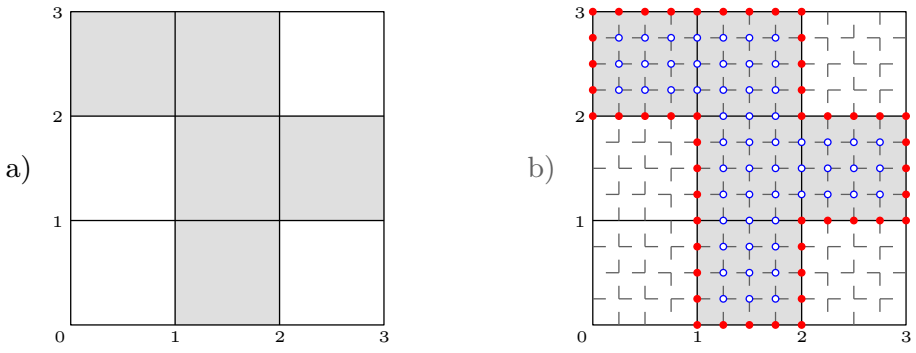

Figure 13: The reduction from the problem of covering a rectilinear polygon with the minimum number of axis-aligned squares, to the SCC problem. a) A rectilinear polygon represented in a 3×3 zero-one matrix, whose vertices are considered lattice points in $[0, 3] \times [0, 3]$. b) The set of red and blue points generated from the polygon.

Any covering set of P is a covering set of S, and conversely, the covering squares of S can be expanded/shifted to be a covering set of P. Namely, let $\mathcal{Q}$ be a covering set of S. First, we assume

that the squares of $\mathcal{Q}$ are closed and we expand them so that they do not contain red points in their interiors. Notice that the side length of each square in $\mathcal{Q}$ is now an integer number. After that, we use Lemma 6.1 in order to shift (horizontally and/or vertically) elements of $\mathcal{Q}$ so that $\mathcal{Q}$ becomes a covering set of P. Every square of $\mathcal{Q}$ is shifted at most once in each direction. It is done as follows:

Let ℓ be a horizontal line passing through points of S. Let $\mathcal{Q}_\ell$ denote the set of squares of $\mathcal{Q}$ intersected by ℓ, and let U_ℓ be their union. We say that a square of $\mathcal{Q}_\ell$ is *lattice* if the x coordinates of its vertices are all integer numbers. If $\mathcal{Q}_\ell$ does not cover $P \cap \ell$, then $(P \cap \ell) \setminus U_\ell$ consists of a set I_ℓ of pairwise-disjoint maximal-length segments. Moreover, the size of each segment in I_ℓ is at most $\frac{1}{M}$ because the distance between consecutive blue points in ℓ is equal to $\frac{1}{M}$. Since the side length of each square in $\mathcal{Q}_\ell$ is an integer number, the total size (or measure) of $(P \cap \ell) \setminus U_\ell$ is at most $\frac{N}{M} < 1$. Therefore, it is easy to see that we can use Lemma 6.1 in order to shift horizontally the non-lattice squares of $\mathcal{Q}_\ell$, so that $\mathcal{Q}_\ell$ covers $P \cap \ell$ and all elements of $\mathcal{Q}_\ell$ are lattice.

By repeating the above process for every horizontal line ℓ passing through points of S, and after that considering ℓ vertical and working analogously, the final set $\mathcal{Q}$ covers P. □

Notice that the SCC problem remains NP-hard if we restrict the squares to be centered at blue points. In fact, we can use the above reduction and add only blue points at the lattice vertices of the interior of P and at the centers of the pixels of P, and red points at the lattice points of the boundary of P.

We have shown in Section 4 that there exists an $O(1)$-approximation algorithm for the SCC problem because a set of squares is a set of pseudo-disks [11] (Statement 4.3).

7 Conclusions and further research

In this paper we have addressed the Class Cover problem with boxes. We proved its NP-hardness and explored some variants by restricting the covering boxes to have special shapes. The main results of this paper are the NP-hardness proofs and the exact algorithms when we cover with strips and top-bottom half-strips, respectively (Subsections 5.1 and 5.2). All the approximation algorithms for the NP-hard problems come from results on ε-nets, which were stated for a more general problem, and the factors of approximation given are asymptotic. The major open problem is to develop approximation algorithms whose approximation factors are either better than or equal, but not asymptotic, to the ones stated here.

A natural variant of the BCC problem to be considered in future research is to use only vertical half-strips as covering objects. At this point, we are unable to give either a polynomial-time exact algorithm or a hardness proof. We can prove that the problem of finding an optimal cover of B with R-empty vertical half-strips so that the top-bottom (resp. bottom-top) half-strips have pairwise-disjoint interiors, is a 2-approximation. This new problem can be solved in polynomial-time by using dynamic programming.

References

[1] P. K. Agarwal and S. Suri. Surface approximation and geometric partitions. In *Proc. of the 5th annual ACM-SIAM symposium on Discrete algorithms*, pages 24–33, 1994.

[2] A. Aggarwal and S. Suri. Fast algorithms for computing the largest empty rectangle. In *Symposium on Computational Geometry*, pages 278–290, 1987.

[3] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. *SIGMOD Rec.*, 27:94–105, 1998.

[4] E. M. Arkin, F. Hurtado, J. S. B. Mitchell, C. Seara, and S. S. Skiena. Some lower bounds on geometric separability problems. *Int. Journal of Computational Geometry and App.*, 16(1):1–26, 2006.

[5] B. Aronov, E. Ezra, and M. Sharir. Small-size ε-nets for axis-parallel rectangles and boxes. *SIAM J. Comput.*, 39(7):3248–3282, 2010.

[6] L. J. Aupperle, Corm, H. E., Keil, J.M., and J. O'Rourke. Covering orthogonal polygons with squares. In *Proc. 26th Annu. Allerton Conf. on Communications, Control and Computing*, pages 97–106, 1988.

[7] J. Backer and J. Keil. The mono- and bichromatic empty rectangle and square problems in all dimensions. In Alejandro López-Ortiz, editor, *LATIN*, pages 14–25. 2010.

[8] C. Bautista-Santiago, J. M. Díaz-Báñez, D. Lara, C. Peláez, and J. Urrutia. On covering a class with arbitrary disks. Manuscript.

[9] H. Brönnimann and M. T. Goodrich. Almost optimal set covers in finite VC-dimension. *Discrete Comput. Geom*, 14(4):463–479, 1995.

[10] A. H. Cannon and L. J. Cowen. Approximation algorithms for the class cover problem. *Annals of Mathematics and Artificial Intelligence*, 40(3-4):215–223, 2004.

[11] K.L. Clarkson and K. Varadarajan. Improved approximation algorithms for geometric set cover. *Discrete Comput. Geom.*, 37(1):43–58, 2007.

[12] J. C. Culberson and R. A. Reckhow. Covering polygons is hard. *J. Algorithms*, 17(1):2–44, 1994.

[13] J. G. Devinney. *The class cover problem and its applications in pattern recognition.* PhD thesis, 2003.

[14] G. Even, D. Rawitz, and S. Shahar. Hitting sets when the VC-dimension is small. *Inf. Process. Lett.*, 95(2):358–362, 2005.

[15] U. Feige. A threshold of ln n for approximating set cover. *J. ACM*, 45(4):634–652, 1998.

[16] M. R. Garey and D. S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA, 1990.

[17] D. J. Hand, P. Smyth, and H. Mannila. *Principles of data mining.* MIT Press, 2001.

[18] D. Haussler and E. Welzl. epsilon-nets and simplex range queries. *Discrete & Computational Geometry*, 2:127–151, 1987.

[19] J. E. Hopcroft and R. M. Karp. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM J. Comput.*, 2(4):225–231, 1973.

[20] L. V. S. Lakshmanan, R. T. Ng, C. X. Wang, X. Zhou, and T. Johnson. The generalized MDL approach for summarization. In *VLDB*, pages 766–777, 2002.

[21] D. T. Lee and Y.-F. Wu. Complexity of some laction problems. *Algorithmica*, 1(2):193–211, 1986.

[22] D. J. Marchette. Class cover catch digraphs. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(2):171–177, 2010.

[23] W. J. Masek. Some NP-complete set covering problems, 1979. Manuscript.

[24] J. Matoušek, R. Seidel, and E. Welzl. How to net a lot with a little: small ε-nets for disks and half-spaces. In *Proc. 6th Annual ACM Symposium on Computational Geometry*, pages 296–306, 1992.

[25] J. S. B. Mitchell. Approximation algorithms for geometric separation problems. Technical report, Dept. of Applied Math. and Statistics, State U. of New York at Stony Brook, 1993.

[26] J. Pach and G. Tardos. Tight lower bounds for the size of epsilon-nets. In *Proc. of the 27th annual ACM symposium on Computational geometry*, SoCG'11, pages 458–463, 2011.

[27] V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Appl.*, 16(2):264–280, 1971.