IMFM
Institute of Mathematics, Physics and Mechanics
Jadranska 19, 1000 Ljubljana, Slovenia

Preprint series
Vol. 48 (2010), 1127
ISSN 2232-2094

THE CLASS COVER PROBLEM WITH BOXES

S. Bereg S. Cabello
J. M. Díaz-Báñez P. Pérez-Lantero
C. Seara I. Ventura

Ljubljana, August 19, 2010

The Class Cover Problem with Boxes

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

August 13, 2010

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 open 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* [9, 13, 21] 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

*Department of Computer Science, University of Texas at Dallas, USA. Partially supported by project 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 MEC MTM2009-08652. {dbanez,pplantero,iventura}@us.es.

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

Preprint series, IMFM, ISSN 2232-2094, no. 1127, August 19, 2010

in at least one of the balls) and with the constraint that balls are centered at blue points. Cannon and Cowen [9] 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 [16]. By solving the Class Cover problem, a simple classifier can be stated; see [21].

In this paper we study a non-constrained version of the Class Cover problem in the plane, named the *Boxes Class Cover problem*, in which axis-aligned open rectangles (or boxes) are considered as the covering objects (see 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 axis-aligned open rectangles (or boxes) such that every point in B is covered by at least one rectangle of $\mathcal{H}$.*

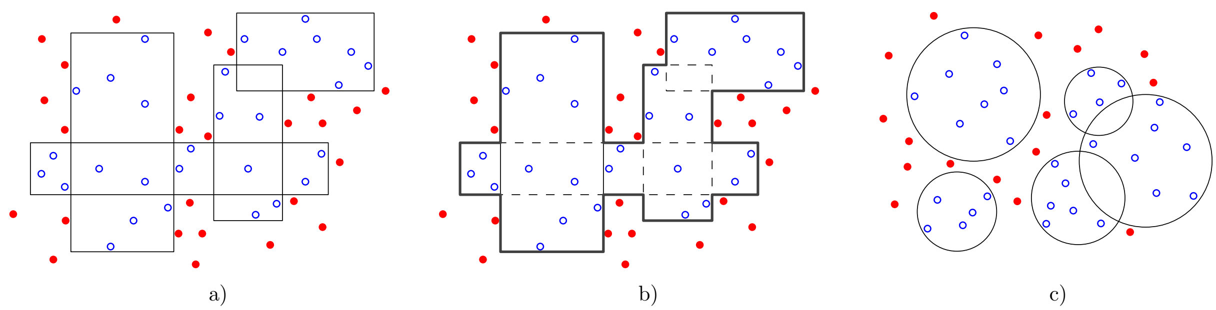


Figure 1: a) Covering the blue class with boxes. b) A solution to the BCC-problem induces a rectilinear polygon separating B from R. c) Covering the blue class with disks.

The problem of covering with disks (instead of boxes), not necessarily centered at blue points, is similar to the BCC-problem (see Figure 1 c)). This version of the class cover problem is NP-hard [6], and as we will see in Section 2.3, it admits a constant-factor approximation algorithm using the techniques of [7, 23].

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 rectangles and squares 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* [24], 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 a rectilinear polygon (see Figure 1 b)). Indeed, the use of rectangles is usual in the description of a point set [2, 19].

Preprint series, IMFM, ISSN 2232-2094, no. 1127, August 19, 2010

Our contributions. (1) We prove the NP-hardness of the BCC-problem by a reduction from the *Rectilinear Polygon Covering problem* [12, 22]. 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 [7, 10, 17, 25], 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 [15]. Moreover, using results from Clarkson and Varadarajan [10] 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 [5], and show the existence of an $O(1)$-approximation algorithm.

Outline of the paper. In Section 2.1 we state a first approach to our problem. In Section 2.2 we prove that the BCC-problem is NP-hard. In Section 2.3 we review related results concerning range spaces and ε-nets, most of which are relevant to give approximation algorithms to our problem. In Section 3 we study the BCC-problem when we restrict the boxes to strips or half-strips. In Section 4 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 5, we state the conclusions and further research.

2 The BCC-problem

In this section we first show a simple exponential algorithm for the BCC-problem. Second, we prove the hardness of the problem, and finally, we provide approximation results.

2.1 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 enlarge 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 can not be enlarged) $\mathcal{H}^*$ 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.

How many maximal boxes can there be for r red points and b blue points? A simple bound is $O(r^4)$ since every side of a maximal box contains a red point or is at infinity. Next we show that $|\mathcal{H}^*| = O(r^2)$ and this bound is tight in the worst case.

Preprint series, IMFM, ISSN 2232-2094, no. 1127, August 19, 2010

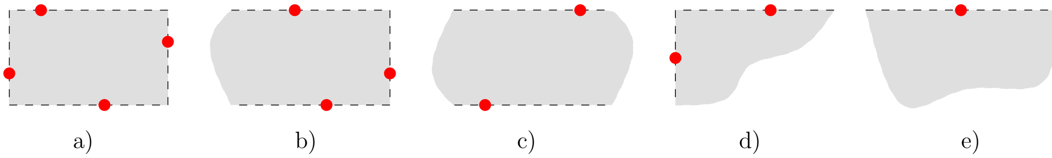


Figure 2: Boxes in $\mathcal{H}^*$: a) rectangle, b) half-strip, c) strip, d) quadrant, e) half-plane.

Let p and q be two red points of S so that q is below the horizontal line through p. Let H_{pq} denote the box with opposite vertices p and q. If H_{pq} is R-empty, then it can be extended horizontally until its left and right sides bump into red points or reach infinity, and then H_{pq} becomes a member of $\mathcal{H}^*$ (see Figure 3 a)). Additionally, we consider H_{pq} a vertical half-line with top-most point p. Then, there are at most $O(r)$ boxes in $\mathcal{H}^*$ whose top side contains the red point p, and thus at most $O(r^2)$ boxes whose top side contains a red point. Analogously, there are $O(r^2)$ boxes in $\mathcal{H}^*$ whose left side (resp. right side, bottom side) contains a red point. Since every box in $\mathcal{H}^*$ has at least one red point on its boundary, we can conclude that $|\mathcal{H}^*|$ is $O(r^2)$.

On the other hand, there are point configurations for which $|\mathcal{H}^*|$ is $\Omega(r^2)$. For example, let r be an even number and consider the sets S_1 and S_2 of red points with $r/2$ points each, separated by a horizontal line as illustrated in Figure 3 b). Assume that the blue points are anywhere. There are $\frac{r}{2} - 1$ different boxes in $\mathcal{H}^*$ for every two consecutive red points in S_1, and thus $(\frac{r}{2} - 1)^2 = \Omega(r^2)$ in total.

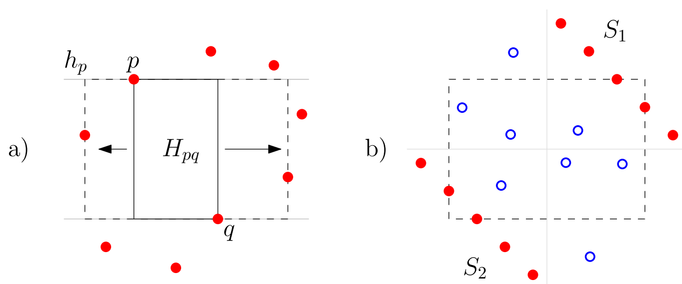


Figure 3: a) Finding boxes in $\mathcal{H}^*$ whose top sides contain the red point p. b) An example with $\Omega(r^2)$ boxes in $\mathcal{H}^*$.

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

If points have distinct x-coordinates (resp. y-coordinates) the upper bound $r + 1$ for $|\mathcal{H}|$ is easy. We prove now that $|\mathcal{H}| \leq 2r + 2$ for any set $S = R \cup B$, possibly with degeneracies.

Let $p, q \in R$, let H_p^- (resp. H_p^+) be the maximum-height box of $\mathcal{H}^*$ whose top (resp. bottom) side contains p, and let 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:

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

Now, 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. Since $|\mathcal{W}| \le 2r+2$, it suffices to show that $\mathcal{W}$ is a cover of $\mathbb{R}^2 \setminus R$ (thus it covers B). Let u be any point from $\mathbb{R}^2 \setminus R$. There are two cases to cover u:

Case 1: There is a red point p on the line $x = x_u$. Assume that p is closest to u. If $A_p = \{H_p^-, H_p^+\}$ then u is covered by a box of A_p. Otherwise, there is a sequence $p_1, p_2, \ldots, p_k$ of red points such that $p_1 = p$, $S_{p_i p_{i+1}} \in \mathcal{W}$ $(1 \le i < k)$, and u is covered by one of the two elements of A_{p_k}.

Case 2: There is no red point on the line $x = x_u$. Let u' (resp. u'') be the orthogonal projection of u on the left (resp. right) side of the vertical strip of $\mathcal{H}^*$ covering u. If u' is not a red point, then u' is covered by *Case 1* and u is covered by the same box as u'. Analogously if u'' is not a red point. Otherwise, if u' and u'' are red points, then $S_{u'u''}$ is in $\mathcal{W}$ and covers u.

For the tightness of the bound, consider the configuration of points depicted in Figure 4. 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 can not be covered by the same R-empty open box. □

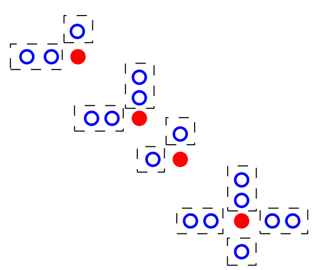


Figure 4: A case in which exactly $2r+2$ R-empty open boxes are needed to cover B.

The above discussion lets us to design the following exponential algorithm to report an exact solution for the BCC-problem:

1. Compute the set $\mathcal{H}^*$ in $O(r^2)$ time,
2. Find the smallest $k \in \{1, \ldots, \min\{2r+2, b\}\}$ such that there exists a subset of $\mathcal{H}^*$ of size k covering B.

There are $(O(r^2))^{\min\{2r+2,b\}} = r^{O(\min\{r,b\})}$ subsets of $\mathcal{H}^*$ to be checked, and the checking of a subset can be done in $O(b \cdot \min\{2r+2, b\}) = O(\min\{rb, b^2\})$ time. The overall time complexity is $r^{O(\min\{r,b\})} \cdot \min\{rb, b^2\} = b \cdot r^{O(\min\{r,b\})}$.

In general, the time complexity of this algorithm is exponential. However, if r or b is $O(1)$, then the time complexity is polynomial.

2.2 Hardness

In this subsection we prove that the BCC-problem is NP-hard. The proof is based on 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 axis-aligned rectangles whose union is exactly P* (see Figure 5). For a general class of rectilinear polygons with holes, Masek proved that the RPC-problem is NP-hard [22]. Culberson and Reckhow used a clever reduction from the 3-SAT-problem [15] to show that the RPC-problem is also NP-hard for polygons without holes [12].

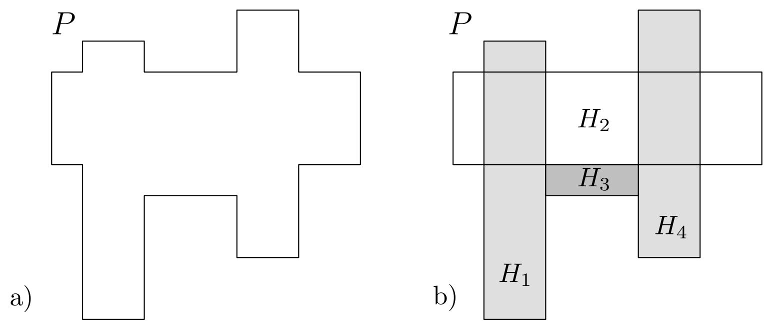


Figure 5: a) A rectilinear polygon P. b) An optimal covering of P with four rectangles.

Theorem 2.2 *The BCC-problem is NP-hard.*

Proof. Let P be an instance for 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 in every vertex of $G \setminus P$ (resp. $G \cap P$) (see Figure 6).

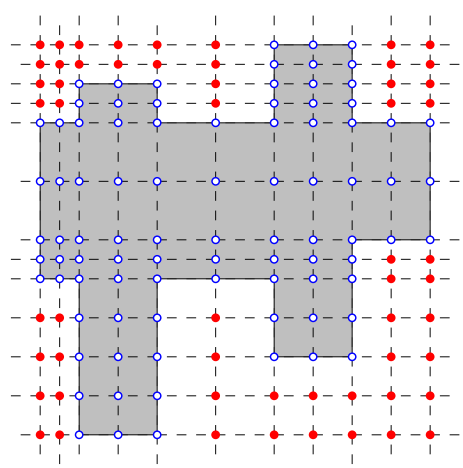


Figure 6: The reduction from the RPC-problem to the BCC-problem.

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 can not be enlarged in order to contain more blue points. Let H'_i, $1 \leq i \leq k$, be the smallest bounding box of $H_i \cap B$ with $H'_i \subseteq P$. If some H'_i is not contained in P, then it must contain at least one cell of $G \setminus P$ with at least one red vertex, say u, and then H_i covers u, a contradiction.

To verify that $\mathcal{H}' = \bigcup_{i=1}^{k} H'_i$ covers P we proceed as follows: Let c be a cell of $G \cap P$. 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 open 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 open box covering them. Therefore, the BCC-problem is equivalent to finding a *Minimum Clique Partition* in G_S [15]. The *Partition Into Cliques Problem* is strongly NP-complete [15], 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.

2.3 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 [7].

Given a range space $(X, \mathcal{R})$, the *set cover* problem asks for the minimum-cardinality subset of $\mathcal{R}$ that covers X [15]. 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 $\mathcal{R}$ [15]. 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 $(1 + \ln |X|)$ [14, 15]. 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 7). As a consequence, we get the following result:

Statement 2.3 *The BCC-problem has an $O(\log b)$-approximation algorithm if we cover with either boxes or axis-aligned squares.*

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

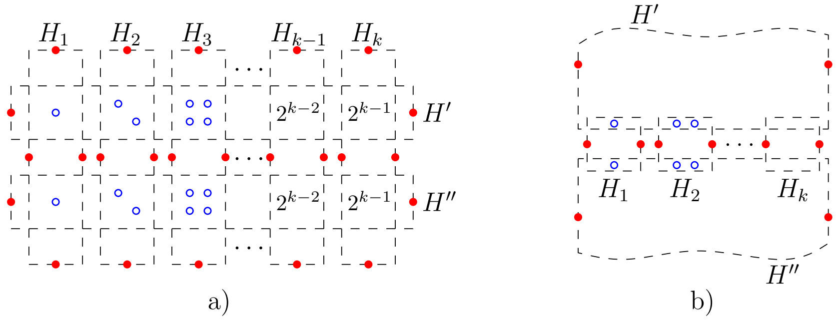


Figure 7: The greedy method gives a logarithmic factor of approximation for both boxes and squares. In a) (resp. b)), each of the intersections of the boxes (resp. squares) H' and H'' with the box (resp. square) H_i ($1 \leq i \leq k$) contains 2^{i-1} blue points. The greedy method reports $\{H_1, H_2, \ldots, H_k\}$ instead of $\{H', H''\}$.

Brönnimann and Goodrich [7] gave a general approach in order to find an approximate hitting set for range spaces, in such a way that when the problem is solved in the dual range space, it gives a set cover in the primal one. Their method is based on finding, as candidate hitting sets, small-size subsets called ε-nets, and it works for range spaces with finite VC-dimension [7, 17, 25]. 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 stated as 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} [7, 25]. Note that the VC-dimension of our range space $(B, \mathcal{H}^*)$ is at most four (i.e. in any subset $P \subseteq B$ with at least five points, there is subset $P' \subset P$ that can not be separated with a box in $\mathcal{H}^*$ from $P \setminus P'$).

For range spaces with constant VC-dimension, the Brönnimann and Goodrich's method reports a hitting set of size at most a factor of $O(\log c)$ from the optimal size c. This result is based on the fact that, for every range space with finite VC-dimension d, there exists an ε-net of size $O(\frac{d}{\varepsilon} \log \frac{d}{\varepsilon})$ [17]. In general, if the range space has constant VC-dimension, and there exists an ε-net of size $O(\frac{1}{\varepsilon}\varphi(\frac{1}{\varepsilon}))$, their method finds a hitting set of size $O(\varphi(c)c)$, where c is the size of an optimal set.

Therefore, since our range space $(B, \mathcal{H}^*)$ has constant VC-dimension (also the dual), the Brönnimann and Goodrich's technique can be applied to obtain in the dual range space a hitting set of size at most a factor of $O(\log c)$ from the optimal size c, which induces a solution $\mathcal{H}$ (a set cover) for the BCC-problem with the same size. Thus we obtain the following result:

Statement 2.4 *The BCC-problem has an $O(\log c)$-approximation algorithm, where c is the size of the optimal covering.*

Remark. Aronov et al. [4] proved the existence of ε-nets of size $O(\frac{1}{\varepsilon} \log \log \frac{1}{\varepsilon})$ for range spaces of points and box ranges. They stated as further research to find, for the dual range space, ε-nets of size less than $O(\frac{1}{\varepsilon} \log \frac{1}{\varepsilon})$. They also mentioned that in [8] the authors claimed, without proof, their existence. Such bound would give an $O(\log\log c)$-approximation algorithm to the BCC-problem.

Matoušek et al. [23] proved the existence of ε-nets of size $O(\frac{1}{\varepsilon})$ for point range spaces and disk ranges. Due to this, techniques from Brönnimann and Goodrich [7] provide an $O(1)$-approximation for the class cover problem with disks. Clarkson and Varadarajan [10] showed that if the geometric range space[2] $(X, \mathcal{R})$ has the property that, given a random subset $R' \subset \mathcal{R}$ 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 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 [10, Theorem 2.2]. If $\mathcal{R}$ is a family of pseudo-disks[3], then the trapezoidization of the complement of any subset $R' \subset \mathcal{R}$ has complexity $O(|R'|)$ and thus the dual range space has ε-nets of size $O(\frac{1}{\varepsilon})$ [10]. Since a set of axis-aligned squares is a family of pseudo-disks, by using the techniques in [7, 10], the following result is obtained:

Statement 2.5 *The BCC-problem has an $O(1)$-approximation algorithm if the covering boxes are restricted to axis-aligned squares.*

3 Solving particular cases

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

3.1 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.

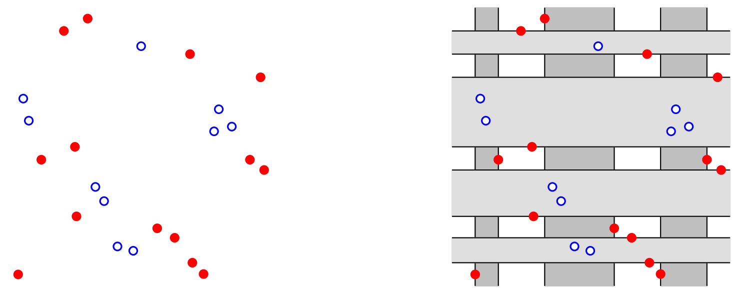
There exists a solution for the BCC-problem if and only if every blue point can be covered by an axis-parallel line avoiding red points. This can be tested in linear time. Suppose that the BCC-problem has a solution. If a blue point and a red point lie on the same vertical (resp. horizontal) line then 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 8), and whose set of edges E is defined as follows: put an edge between

[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.

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.



a) b)

Figure 8: a) A set of red and blue points. b) Strips covering at least one blue point.

Since each blue point is covered by exactly two strips, the problem is reduced to finding a *Minimum Vertex Cover* [15] 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 [18]. Thus, the following result is obtained:

Theorem 3.1 *The BCC-problem can be solved in $O(r \log r + b \log b + \sqrt{r}b)$ time if we only use axis-aligned strips as covering objects.*

3.2 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 9. 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. Let p_l (resp. p_r) be the red point such that the left (resp. right) endpoint of s_p is located in the ray corresponding to p_l (resp. p_r). We say that p_l (resp. p_r) is the left (resp. right) red neighbor of p (Figure 9).

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 whose top side is s_p translated upwards until it touches a red point or reaches the infinite. 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 of this algorithm 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' can not be covered with the same top-bottom half-strip. This is so because every top-bottom half-strip which covers both p and p', contains at least one of the two red neighbors of p.

Preprint series, IMFM, ISSN 2232-2094, no. 1127, August 19, 2010

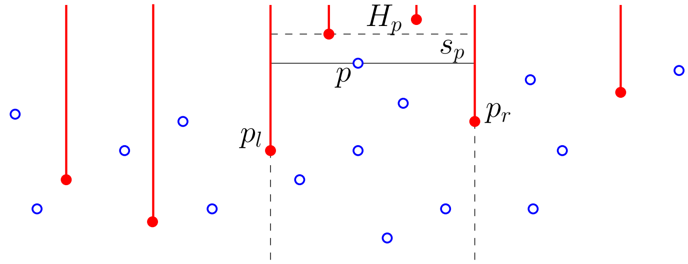


Figure 9: The ray structure.

Algorithm. We first preprocess S to obtain the decreasing y-coordinate order of the elements of B, and build two balanced binary search trees T_B and T_R containing the blue and the red points sorted by x-coordinate, respectively. The first tree allows the deletion of elements. In the second tree, each node v is labeled with the element of minimum y-coordinate in the subtree rooted at v. This labeling permits us to obtain the red neighbors for a given blue point p and to determine the top side of H_p, both in $O(\log r)$ time. This preprocess takes $O(r \log r + b \log b)$ time in total. If there are a blue point p and a red point q such that $x_p = x_q$ and $y_p > y_q$, then there is no solution to our problem. It can be checked in $O(r + b)$ time by simultaneous in-order traversals on T_R and T_B. Now, we do the following for each blue point p in the decreasing y-coordinate order which is not still covered (i.e., p is in T_B): find the left and the right red neighbors p_l and p_r of p, determine H_p and include it in the solution, in $O(\log b + k_p)$ time find the k_p blue points in T_B covered by H_p (i.e., those points p' in T_B such that $x_{p_l} < x_{p'} < x_{p_r}$) and remove them from T_B in $O(k_p \log b)$ time. The total time complexity is $O(r \log r + b \log b) + \sum_{p \in B} O(\log b + \log r + k_p \log b) = O(r \log r + b \log b + b \log r) = O(r \log r + b \log b)$.

We show now how to reduce the asymptotic time complexity of the algorithm above. Suppose that $r < b$. We prune the set of blue points as follows. Sort the red points by x-coordinate to obtain the ordered sequence X_R. 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 by the above algorithm. They can be computed in $O(b \log r)$ time using binary search in X_R. The overall time complexity is $O((r + b) \log r)$. 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 3.2 *The BCC-problem can be solved in $O((r + b) \log(\min\{r, b\}))$ time if we only use half-strips in one direction as covering objects.*

Next we show that for $\min\{r, b\} = \Omega(r + b)$ the above algorithm is optimal in the algebraic computation 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$(X) = \max_{1 \leq i < n}\{x_{\pi_{i+1}} - x_{\pi_i}\}$ [20]. Arkin et al. [3] 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 computation 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 3.3 *The BCC-problem has an $\Omega(n \log n)$ lower bound in the algebraic computation tree model if we only use 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 in 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 in 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 $\text{MAX-GAP}\{x_1, \ldots, x_n, 0, \varepsilon, 2\varepsilon, \ldots, n\varepsilon\} < \varepsilon$ if and only if the minimum number of covering half-strips (or strips) is exactly n (Figure 10). □

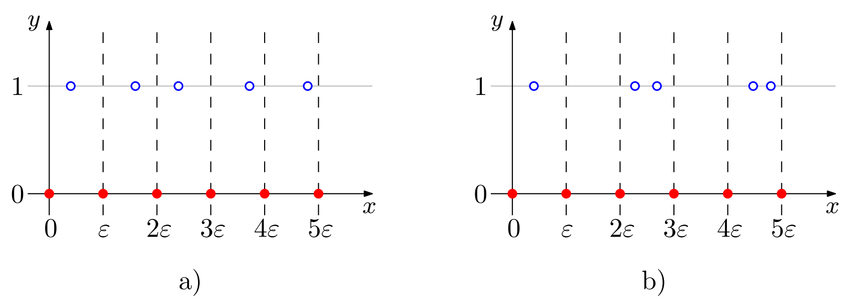


Figure 10: Construction of the reduction from the MAX-GAP to our problem.

3.3 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 [7, 10].

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 3.1 and 3.2.

Theorem 3.4 *The HSCC-problem is NP-hard.*

Proof. To prove the NP-hardness we use a reduction from the 3-SAT-problem [15]. 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 [15].

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 11). 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 in $\alpha_i \cap \beta_i$ a set V_i of red and blue points as follows (Figure 11). 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 $y_{p''} > y_q$.

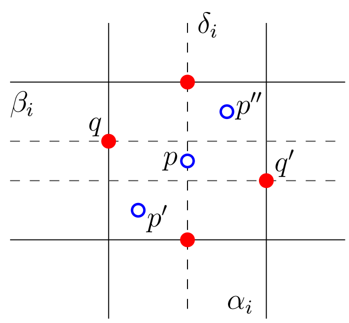


Figure 11: Set of bicolored points V_i for x_i in the reduction from the 3-SAT-problem.

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 12). 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 between ℓ_3 and the left boundary of β_k, above ℓ_2' and close to ℓ_2'; and the last one on ℓ_2' and between the right boundary of β_k and ℓ_3'.

Now we add blue points. Put a blue point in the intersection of ℓ_1' and ℓ_3, and another in the intersection of ℓ_3' and the bottom boundary of β_{t+j}. If x_i is not negated in C_j, then put in 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 in 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 in 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 on the bottom boundary.

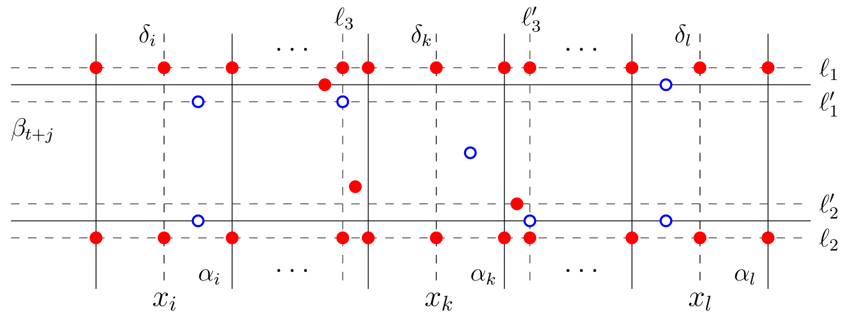


Figure 12: The set W_j of red and blue points for the clause $C_j = (x_i \vee x_k \vee \neg x_l)$.

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 can not 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 13 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 13 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).

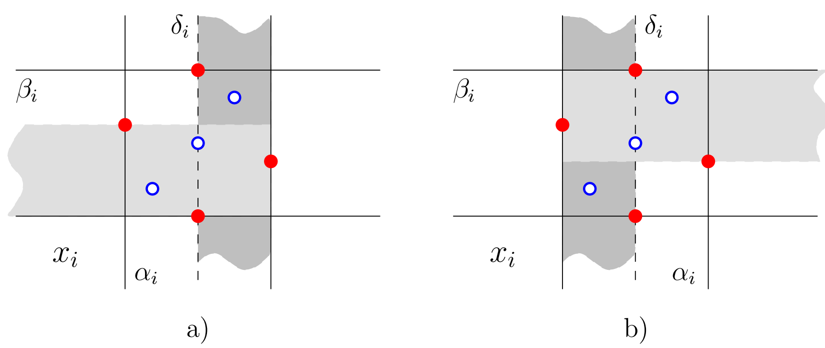


Figure 13: The two ways of optimally covering the blue points associated to a variable x_i. a) x_i is equal to true, b) x_i is equal to false.

For each clause C_j $(1 \leq j \leq m)$ that involves the variables x_i, x_k, and x_l $(1 \leq i < k < l \leq m)$ we 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, at least three half-strips are needed.

Due to the above observations we claim that $\mathcal{F}$ is satisfiable if and only if the blue points in S can be covered with $2t + 2m$ half-strips. 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

Preprint series, IMFM, ISSN 2232-2094, no. 1127, August 19, 2010

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, we can not use less than $2t + 2m$ half-strips to cover blue points in S, thus if we use exactly $2t + 2m$, then we have to use two per each variable, and two 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 [10], 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 3.5 *There exists 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 [10]) an $(\frac{\varepsilon}{2})$-net N_v of size $O(\frac{2}{\varepsilon})$ 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 $O(\frac{2}{\varepsilon})$. We claim that $N_v \cup N_h$ is an ε-net of size $O(\frac{4}{\varepsilon}) = O(\frac{1}{\varepsilon})$ 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 [7] and Clarkson and Varadarajan [10, Theorem 3.2] that there exists a polynomial-time $O(1)$-approximation algorithm for the HSCC-problem. □

4 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. [5] 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 left(s) the left endpoint of s. Let t be the largest integer less than or equal to left(s). We say that s is *lattice* if left$(s) = t$. Otherwise, we say that we *adjust* s, or that s is *adjusted*, if we shift s so that either left$(s) = t$ or left$(s) = t + 1$. Given $X \subset \mathbb{R}$, let $\mathrm{m}(X)$ denote the Lebesgue measure of X.

Preprint series, IMFM, ISSN 2232-2094, no. 1127, August 19, 2010

Lemma 4.1 *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 $\mathrm{m}([0, N] \setminus U)$ 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 do the following for $j = 0, 1, \ldots, N-1$: Denote by I_j the subset of intervals $s \in I$ such that $j < \mathrm{left}(s) < j+1$. Let c_j be the condition that there is an interval $s_1 \in I$ such that either j belongs to the interior of s_1 or $j = \mathrm{left}(s_1)$. We consider two actions:
Action 1: Adjust all intervals $s_2 \in I_j$ so that $\mathrm{left}(s_2) = j+1$.
Action 2: Given $s_3 \in I_j$, adjust both s_3 and all intervals $s_2 \in I_j \setminus \{s_3\}$ so that $\mathrm{left}(s_3) = j$ and $\mathrm{left}(s_2) = j+1$.
If c_j holds, then we apply *Action 1*. Otherwise, let $s_3 \in I_j$ be the interval minimizing $\mathrm{left}(s_3)$. We have that the open interval $(j, \mathrm{left}(s_3))$ is contained in $[0, N] \setminus U$, and we apply *Action 2*.

We show now that this process is correct. Consider an iteration j of the process. Note that all intervals $s \in I$ such that $\mathrm{left}(s) \le j$ are lattice. Moreover, each non-lattice interval is adjusted, and every time it is done, $\mathrm{m}([0, N] \setminus U)$ does not increase. For a given value of j we have that: If c_j holds, then s_1 is lattice because $\mathrm{left}(s_1) \le j$. Thus, since s_1 has integer length, we obtain that $[j, j+1] \subseteq s_1$. If c_j does not hold, let s be the interval of I such that $j < \mathrm{left}(s)$ and $\mathrm{left}(s)$ is minimized. We have that $(j, \mathrm{left}(s)) \subseteq [0, N] \setminus U$. As a consequence, $\mathrm{left}(s) < j+1$ because $\mathrm{left}(s) - j = \mathrm{m}((j, \mathrm{left}(s))) \le \mathrm{m}([0, N] \setminus U) < 1$. Thus, $s \in I_j$ and *Action 2* is applied by considering $s_3 = s$. We obtain that $[j, j+1] \subseteq s_3$. Hence, we have that $[j, j+1] \subseteq U$ for $j = 0, \ldots, N-1$. Therefore, $U = [0, N]$ and the result follows. □

Theorem 4.2 *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 14 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 in every vertex of G that is in the interior (resp. boundary) of P (Figure 14 b)). Let S be the resulting bicolored point set.

Any covering set of P is a covering set of S, and conversely, the covering squares of S can be enlarged/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 enlarge 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 4.1 in order to shift elements of $\mathcal{Q}$, horizontally or vertically, in such a way $\mathcal{Q}$ becomes a covering set of P. It is 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. 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 4.1 in order to shift horizontally squares of $\mathcal{Q}_\ell$ so that $\mathcal{Q}_\ell$ covers $P \cap \ell$.

Preprint series, IMFM, ISSN 2232-2094, no. 1127, August 19, 2010

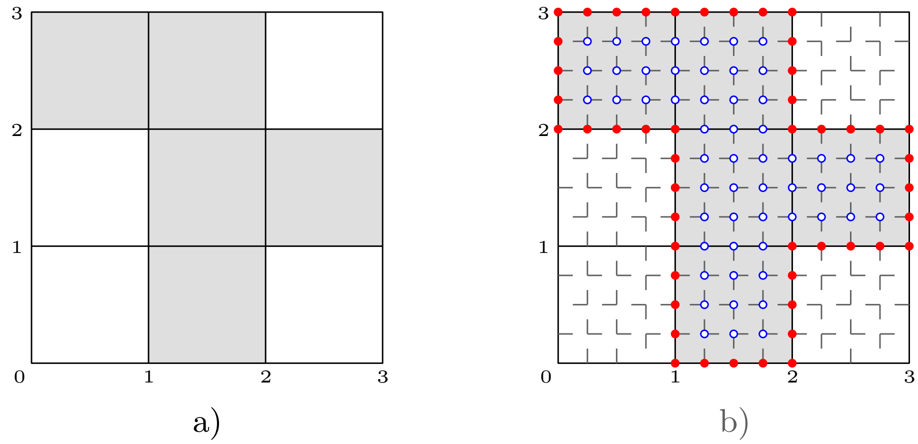


Figure 14: The reduction from the problem of covering a rectilinear polygon with the minimum number of axis-aligned squares, to the SCC-problem. a) A rectilinear polygon represented in a 3×3 zero-one matrix, whose vertices are considered lattice points in $[0, 3] \times [0, 3]$. b) The set of red and blue points generated from the polygon.

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 only add blue points over the lattice vertices of the interior of P and at the centers of the pixels of P, and red points over the lattice points of the boundary of P.

We have shown in Section 2.3 that there exists an $O(1)$-approximation algorithm for the SCC-problem because a set of squares is a set of pseudo-disks [10] (Statement 2.5).

5 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 (see Subsections 3.1 and 3.2). 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. *Proceedings of the fifth annual ACM-SIAM symposium on Discrete algorithms*, pp. 24–33, January 23-25, 1994, Arlington, Virginia, United States.

[2] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Paghavan. Automatic subspace clustering of high dimensional data for data mining applications. *ACM SIGMOD International Conference on Management of Data*, 1998.

[3] E.M. Arkin, F. Hurtado, J.S.B. Mitchell, C. Seara, and S.S. Skiena. Some lower bounds on geometric separability problems. *International Journal of Computational Geometry and Applications*, Vol. 16, No. 1, pp. 1–26, 2006.

[4] B. Aronov, E. Ezra, and M. Sharir. Small-size ε-nets for axis-parallel rectangles and boxes. *SIAM J. Comput.* Vol. 39, No. 7, pp. 3248–3282, 2010.

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

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

[7] H. Brönnimann and M.T. Goodrich. Almost optimal set covers in finite VC-dimension. *Discrete and Computational Geometry*, Vol. 14, pp. 463–479, 1995.

[8] H. Brönnimann and J. Lenchner. Fast almost-linear-sized nets for boxes in the plane. *In Proc. 14th Annu. Fall Workshop Comput. Geom.*, pp. 36-38, 2004.

[9] A.H. Cannon and L.J. Cowen. Approximation algorithms for the class cover problem. *Annals of Mathematics and Artificial Intelligence*, Vol. 40, pp. 215–223, 2004.

[10] K.L. Clarkson and K. Varadarajan. Improved approximation algorithms for geometric set cover. *Discrete and Computational Geometry*, Vol. 37, pp. 43–58, 2007.

[11] T.H. Cormen, C.E. Leiserson, and R.L. Rivest. *Introduction to Algorithms*. MIT Press, McGraw-Hill, 2001.

[12] J.C. Culberson and R.A. Reckhow. Covering polygons is hard. *J. Algorithms*, Vol. 17, pp. 2-44, 1994.

[13] J.G. Devinney and C. Priebe. The class cover problem and its applications in pattern recognition. *The Johns Hopkins University*, 2003.

[14] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of ACM*, Vol. 45, No. 4, pp. 634-652, 1998.

[15] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman & Co., New York, 1979.

[16] H. Hand, H. Mannila, and P. Smyth. *Principles of data mining*. The MIT Press, 2001.

[17] D. Haussler and E. Welzl. ε-nets and simplex range queries. *Discrete Computational Geometry*, Vol. 2, pp. 127–151, 1987.

[18] J.E. Hopcroft and R.M. Karp. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SICOMP*, Vol. 2, pp. 225-231, 1973.

[19] L.V.S. Lakshmanan, R.T. Ng, C.X. Wang, X. Zhou, and T.J. Johnson. The generalized MDL approach for summarization. *International Conference on Very Large Data Bases*, 2002.

[20] D.T. Lee and Y.F. Wu. Geometric complexity of some location problem. *Algorithmica*, Vol. 1, pp. 193-211, 1986.

[21] D. Marchette. Class cover catch digraphs. *Wiley Interdisciplinary Reviews: Computational Statistics*, Vol. 2, No. 2, pp. 171–177, 2010.

[22] W.J. Masek. Some NP-complete set covering problems. Manuscript, MIT, Cambridge, MA, 1979.

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

[24] J.S.B. Mitchell. Approximation algorithms for geometric separation problems. *Technical Report*, Dept. of Applied Math. and Statistics, SUNY at Stony Brook, July 1993.

[25] V.N. Vapnik and A.Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Appl.*, Vol. 16, pp. 264–280, 1971.