Theoretical Computer Science 263 (2001) 17–29

Theoretical Computer Science

www.elsevier.com/locate/tcs

On point covers of c-oriented polygons ☆

Frank Nielsen

Sony Computer Science Laboratories, Takanawa Muse Building, 3-14-13 Higashi Gotanda, Tokyo 141-0022, Japan

Accepted April 2000

Abstract

Let $\mathscr{S}$ be any family of n *c-oriented* polygons of the two-dimensional Euclidean plane $\mathbb{E}^2$, i.e., bounded intersection of halfplanes whose normal directions of edges belong to a fixed collection of c distinct directions. Let $\phi(\mathscr{S})$ denote the packing number of $\mathscr{S}$, that is the maximum number of pairwise disjoint objects of $\mathscr{S}$. Let $\tau(\mathscr{S})$ be the transversal number of $\mathscr{S}$, that is the minimum number of points required so that each object contains at least one of those points. We prove that $\tau(\mathscr{S}) \leqslant \mathfrak{G}(2,c)\phi(\mathscr{S})\log_2^{c-1}(\phi(\mathscr{S})+1)$, where $\mathfrak{G}(2,c)$ is the Gallai number of pairwise intersecting c-oriented polygons. Our bound collapses to $\tau(\mathscr{S}) = \mathrm{O}(\mathfrak{G}(2,c)\phi(\mathscr{S}))$ if objects are more or less of the same size. We describe a $t(n,c) + \mathrm{O}(nc \log \phi(\mathscr{S}))$-time algorithm with linear storage that computes such a 0-transversal, where $t(n,c)$ is the time required to pierce pairwise intersecting c-oriented polygons. We provide linear-time algorithms $t(n,c) = \Theta(nc)$ for α-fat c-oriented polytopes, translates or homothets of $\mathbb{E}^d$ proving that $\mathfrak{G}(2,c) = \mathrm{O}(\alpha)^d$, $\mathfrak{G}(2,c) \leqslant d^d$ and $\mathfrak{G}(2,c) \leqslant (3d^{3/2})^d$ respectively. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: Computational geometry; Output-sensitive algorithms; Precision-sensitive heuristics; Transversal and packing numbers

1. Introduction and preliminary notations

Let $\mathscr{S} = \{P_1, \ldots, P_n\}$ be a set of n d-dimensional geometric objects of the Euclidean space $\mathbb{E}^d$ and $\mathscr{P}$ be a point set. We say that $\mathscr{P}$ is a 0-*transversal* of $\mathscr{S}$ if and only if every object P_i is pierced by $\mathscr{P}$, i.e., $P_i \cap \mathscr{P} \neq \emptyset$. $\mathscr{P}$ is said to be a *covering* or *stabbing* point set. The transversal number of $\mathscr{S}$ is defined as the minimum size of any 0-transversal of $\mathscr{S}$. Finding the minimum k so that $\mathscr{S}$ can be *pierced* (i.e., stabbed) by k points has been shown to be NP-complete [13] as soon as $d \geqslant 2$. Even in one-dimensional case, this problem remains NP-complete for non-convex instances [17].

☆ Part of this work has been done at École Polytechnique (France) and INRIA Sophia-Antipolis, Project PRISME (France).

E-mail address: nielsen@csl.sony.co.jp (F. Nielsen).

PII: S0304-3975(00)00227-9

The geometric covering/piercing problem is also referred in the literature as the *set covering problem* (SCP), or dually as the *hitting set problem* (HSP), where it is transformed into an optimization problem by means of matrix formulations. In this section, we will briefly sum up previously known complexity results (Section 1.1) and heuristics on abstract sets (Section 1.2). Then we will emphasize on its geometric counterpart (Section 1.3). Finally to conclude the first section, we will introduce Gallai numbers (Section 1.4). Section 2 defines c-oriented and b-stripped polytopes and give some of their fundamental properties. In Section 3, we give an output-sensitive precision-sensitive algorithm for the geometric SCP of c-oriented polygons and study its time complexity as well as its performance ratio. Section 4 describes how the algorithm can be slightly modified in order to compute independent sets (i.e., pairwise non-intersecting objects) of precision-sensitive size (although the counterpart on abstract graph is $\Omega(n^{1/4})$ hard to approximate in polynomial time). The *packing number* of $\mathscr{S}$ is defined as the maximal size of any independent set of $\mathscr{S}$. Finally, Section 5 summarizes up-to-date results on the geometric SCP/HSP.

1.1. Abstract set aspects

Let $\mathscr{V} = \{S_i \mid i \in I\}$ be a collection of $v = |\mathscr{V}| = |I|$ subsets of $2^{\mathscr{S}}$ for a set $\mathscr{S}$ of n elements. We want to find a minimal covering collection, i.e., a subset $I' \subseteq I$ of indices such that $\mathscr{S} = \bigcup_{i \in I'} S_i$ with $|I'|$ as small as possible. In other words, we want to minimize $e_v^{\mathrm{T}} \times x = |I'|$ subject to $Ax \geqslant e_n$ for x a $\{0,1\}^v$-vector, e_k a k-dimensional vector of 1's and A a $(v \times n)$-binary matrix, each column of which is the incidence vector of one of the sets I_i, $1 \leqslant i \leqslant v$. The above formulation gives an *integer linear program*. Using these notations, the set cover system is said *ε-dense* (*δ-super-dense*) if $|S_i| \geqslant \varepsilon |\mathscr{S}|$ (resp. $|S_i| \geqslant |\mathscr{S}| - \mathrm{o}(m^{\delta})$), for all $i \in I$. The SCP remains Max SNP-hard even on dense cases but can be solved polynomially in super-dense cases [3, 20].

1.2. Heuristics on abstract sets

Chvátal [6, 26] gave a quadratic-time greedy algorithm to find a cover set of size $\Delta(\mathscr{S})$ such that $\Delta(\mathscr{S}) \leqslant \tau(\mathscr{S})(\log k - \log\log k + 0.78)$, where k is the maximum column sum ($k \leqslant n$) of A. Notice that SCP can be solved in polynomial time whenever $k \leqslant 2$ (using a maximum matching algorithm in a bipartite graph) and that k-SCP have better heuristics using semi-local optimization [9]. An algorithm is *precision-sensitive* if its performance ratio does not depend on the input size but rather on the size of any optimal solution (greedy algorithm is not precision-sensitive). Hochbaum [15] proposed a cubic-time algorithm with a cover set of size at most $\tau(S)f$, where f is the maximum row sum of incidence matrix A using an analytical Russian method and a linear program relaxation. Interestingly, Feige [11] showed that no polynomial-time algorithm can approximate the optimal solution within a factor of $(1-\varepsilon)\log|\mathscr{S}|$, unless $\mathrm{NP} \subseteq \mathrm{DTIME}[n^{\log\log n}]$, where $1 > \varepsilon > 0$. Considering A as an hypergraph, we have $\tau(\mathscr{S}) \leqslant 11\lambda(\mathscr{S})^2(\lambda(\mathscr{S}) + \phi(\mathscr{S}) + 3)\binom{\lambda(\mathscr{S})+\phi(\mathscr{S})}{\phi(\mathscr{S})}$, where $\lambda(\mathscr{S})$ is the maximum l

so that the incidence matrix A has as a submatrix the incidence matrix of the complete graph $\mathscr{K}_l$ (clique of size l) [7].

1.3. Heuristics on geometric instances

One major drawback from the computational geometrical point of view is that these methods do not consider geometrical objects nor their shapes but require matrix A. One way to proceed is to consider from the whole arrangement of the constant-size descriptive objects of the d-dimensional Euclidean space $\mathbb{E}^d$ all the sets induced by k-faces, $0 \leqslant k \leqslant d$. We label each k-face with the set of objects fully containing it. A label is said to be *maximal* if it is not included in another one. We remove non-maximal labels and obtain a so-called *Sperner system*, still possibly of size $O(n^d)$ [22]. Hochbaum and Maass [16] considered the case of geometrical objects and gave a *polynomial-time approximation scheme*. Their algorithm allows us to consider sets of congruent star-shaped centrally symmetric objects T, or dually covering sets of points with star-shaped translates $T^* = T$. In that context, piercing families of c-oriented translates of a given polytope is of particular interest since it corresponds to covering a set of points by a minimum number of congruent copies of a given polytope. Brönninman and Goodrich [5] investigate these problems using the concept of the *Vapnik–Červonenkis dimension*. They obtain precision-sensitive set covers if the VC-dimension is bounded as it is usually the case when considering geometric scenes. Their evolutionary algorithm still relies on the fact that matrix A is computed beforehand. Recently, Efrat et al. [10] studied dynamic data-structures for fat objects and obtain efficient piercing algorithms under the fatness assumptions in dimension 2 and 3. As a byproduct, we have $\tau(\mathscr{S}) = O(\phi(\mathscr{S}))$ for fat objects in arbitrary fixed dimension. The case of isothetic boxes in arbitrary dimension has been studied in [18, 12, 19, 23]. It is worth noting that for d-dimensional boxes, we have the following inequality:

$$\tau(\mathscr{S}) \leqslant \phi(\mathscr{S}) \log_2^{d-1} \phi(\mathscr{S}) + d - \tfrac{1}{2}\phi(\mathscr{S}) \log_2^{d-2} \phi(\mathscr{S}).$$

Moreover, efficient $O(dn \log \phi(\mathscr{S}))$-time $O(dn)$-space output-sensitive algorithms have been designed to compute such a 0-transversal in that case. In this paper, we generalize the methodology we used for boxes to c-oriented polygons. Observe that for general objects, there is no relationship between $\tau(\mathscr{S})$ and $\phi(\mathscr{S})$. For example, let us take $\mathscr{S}$ as n pairwise intersecting segments in non-degenerate position then, it is clear that $\tau(\mathscr{S}) = \lceil n/2 \rceil$ and $\phi(\mathscr{S}) = 1$. This observation can be extended to polytopes by transforming each segment to a polytope obtained as the Minkowski sum and a tiny d-dimensional cube so that the overall resulting set of polytopes has the same intersection graph (see Fig. 1). Therefore, it is of particular interest to focus on the restricted case of c-oriented polygons for which we obtain the first (to our knowledge) non-trivial upper bounds.

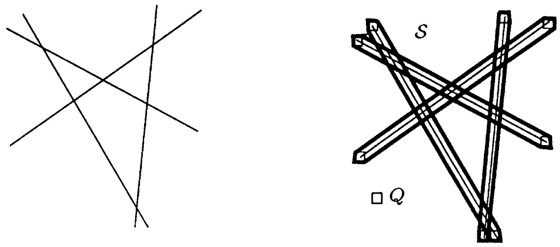

Fig. 1. A family $\mathscr{S}$ of n pairwise intersecting polytopes ($\phi(\mathscr{S})=1$) that requires $\tau(\mathscr{S})=\lceil n/2 \rceil$ points to pierce them.

1.4. Introducing Gallai numbers

Let $\prod^q$ denotes the q-pierceability property, i.e., being pierceable by q points. We define Gallai numbers as follows:

Definition 1. If every subset of $\mathscr{S}$ ($\mathscr{S} \in \mathscr{C}$) of size p is $\prod^q$ (let $\prod^q_p$ denote that property) then $\mathscr{S}$ is pierceable with $\mathfrak{G}(p,q,\mathscr{C})$ points, where $\mathfrak{G}(p,q,\mathscr{C})$ is the Gallai number of class $\mathscr{C}$ for the $\prod^q$ property of its p-subsets.

For example $\mathfrak{G}(2,1,\mathscr{D}^2)=4$, where $\mathscr{D}^2$ is the class of planar disks. [1] In the sequel, $\mathfrak{G}(2,c)$ will denote the Gallai number of pairwise intersecting c-oriented polytopes.

2. Properties of c-oriented polytopes of $\mathbb{E}^d$

An object $P \in \mathscr{S}$ is said to be c-oriented if it can be expressed as the intersection of translated halfspaces of a given family $\mathscr{H}=\{H_1,\ldots,H_c\}$ of c halfspaces. We denote the corresponding family of c bounding hyperplanes by $\partial\mathscr{H}$: $\partial\mathscr{H}=\{\partial\mathscr{H}_1,\ldots,\partial\mathscr{H}_c\} =\{h_1,\ldots,h_c\}$. A geometric object P is a *c-oriented polytope* if it is both c-oriented and bounded. Next, we introduce *b-stripped* objects as polytopes defined by the intersection of at most b strips, where a *strip* is the intersection of two halfspaces whose corresponding bounding hyperplanes are translates of each other. Thus, d-boxes are $2d$-oriented but d-stripped. A strip S is described by means of a d-dimensional vector a and two real values $\alpha \leqslant \beta$ such that $S=\{x \in \mathbb{E}^d \mid \alpha \leqslant a \cdot x \leqslant \beta\}$ (see Fig. 2). Each c-oriented polytope $P_i \in \mathscr{S}$ is the product of $b \leqslant c$ strips: $P_i=\prod_{j\in\{1,\ldots,b\}} strip(a_j,\alpha_{i,j},\beta_{i,j})$. The intersection of two b-stripped polytopes $P \cap P'$ is also a b-stripped polytopes. Note that $P \cap P'$ may be empty even if the cartesian product of its strips is not empty, i.e.,

$$\prod_{j\in\{1,\ldots,b\}} [\max\{\alpha_j,\alpha'_j\},\min\{\beta_j,\beta'_j\}] \neq \emptyset.$$

[1] For congruent pairwise intersecting disks, three points suffice. Surprisingly, it has been shown by Grünbaum that $\mathfrak{G}(2,1,\mathscr{D}^d)>1.003^d$, where $\mathscr{D}^d$ is the family of d-dimensional balls.

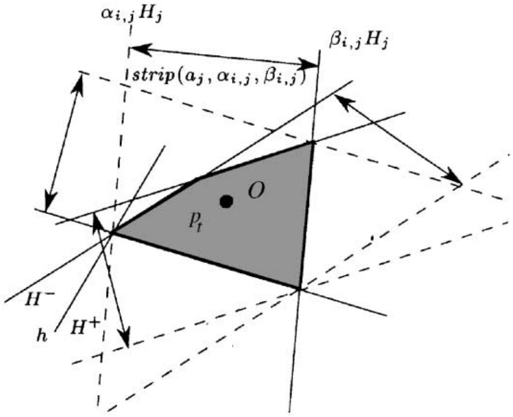

Fig. 2. A 4-oriented polytope P_i, the supporting hyperplanes of its facets and the four induced strips.

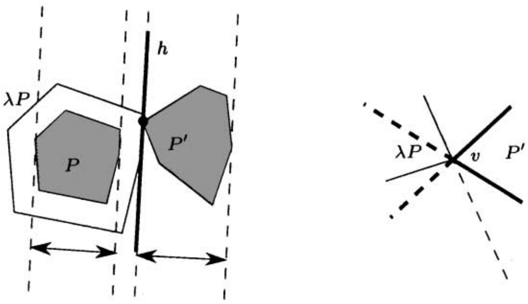

Fig. 3. $P \cap P' = \emptyset$ implies two disjoint strips. left: vertex–edge contact, right: vertex–vertex contact.

Clearly, the converse is true only for boxes (up to a projective transformation, see [25, p. 329]). We denote by $strip_l(P)$ the real interval $[\alpha_l, \beta_l]$. A *non-empty polytope* P' lies within P if and only if $strip_l(P') \subseteq strip_l(P)$ for all $l \in \{1, \ldots, b\}$.

Theorem 1. *Two non-empty c-oriented polygons P and P' have an empty intersection if and only if there exists $l \in \{1, \ldots, c\}$ such that $strip_l(P') \cap strip_l(P) = \emptyset$.*

The key point is to prove that disjoint non-empty c-oriented polygons have at least a pair of disjoint strips.

Proof. Assume that $P \cap P' = \emptyset$ and that the origin O is within P. Let λP be the smallest homothet of P with scaling factor λ such that $\lambda P \cap P' \neq \emptyset$. Clearly for $1 \leqslant \lambda' < \lambda$, we have $P \subseteq \lambda' P \subset \lambda P$. Since $P \cap P' = \emptyset$, we have $\lambda > 1$. λP and P' have a either an edge–vertex (Fig. 3, left) or vertex–vertex contact (Fig. 3, right). In the first case, let h be the line passing through that edge, then h separates P from P' and thus

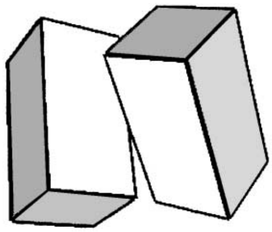

Fig. 4. Two non-empty 6-stripped polytopes of $\mathbb{E}^3$ which do not admit a pair of non-empty strips.

the corresponding strips of P and P' of same orientation are disjoint. In the second case, there is always one line passing through an edge incident at v that separate P from P'. $\square$

Here, we would like to point out the main differences between boxes (whenever $b = d$) and c-oriented polytopes. In the case of boxes, a set $\mathscr{S}$ of boxes has a non-empty intersection ($\bigcap \mathscr{S} \neq \emptyset$) iff $\forall i \in \{1, \ldots, d\}$ $strip_i(\mathscr{S}) \neq \emptyset$. In terms of Gallai numbers, we have $\mathfrak{G}(2,1,\mathscr{B}) = \mathfrak{G}(2,d) = 1$ for the class $\mathscr{B}$ of d-dimensional boxes. However, we have $\mathfrak{G}(2,1,\mathscr{H}) = \mathrm{O}(1)$ for the class $\mathscr{H}$ of translates or homothets. Theorem 1, which can be seen as a refinement of Farkas' lemma on the plane, is not anymore true for c-oriented polytopes (because of the possibly non-degenerate edge–edge contact, see Fig. 4).

3. An output-sensitive algorithm

In [23], an optimal $\Theta(n \log \phi(\mathscr{S}))$-time algorithm with linear storage is given for computing an optimal covering point set of size $\Delta(\mathscr{S}) = \tau(\mathscr{S}) = \phi(s)$ for a family of n intervals. The algorithm for piercing c-oriented polygons proceeds through two stages as described below:

Partitioning. Create a partition $\mathscr{Q}$ of $\mathscr{S}$ such that any set $Q \in \mathscr{Q}$ has the following weak *pairwise intersection property*:

$$\forall P, P', P'' \in Q, \quad (P \cap P' \neq \emptyset) \quad \text{and} \quad (P' \cap P'' \neq \emptyset) \;\Rightarrow\; (P \cap P'' \neq \emptyset).$$

Viewing Q as its intersection graph, means that Q has a trivial disjoint clique covering, such that for any clique $K \in Q$, we have $\forall i \in \{1, \ldots, c\}$, $strip_i(K) \neq \emptyset$.

In the following, each of the cliques of Q is called an element of a *cluster* (visually more appealing), Q is called a cluster, and $\mathscr{Q}$ is said to be a *cluster covering*.

Piercing. Pierce c-oriented polygons whose corresponding induced families of strips intersect pairwise. That is, pierce families of pairwise intersecting c-oriented polygons (clusters, see Theorem 1).

3.1. Getting a partition into pairwise intersecting convex polygons

We choose an arbitrary direction, say h_1, and consider the induced family $\mathscr{S}_1$ of n intervals (perpendicular projection):

$$\mathscr{S}_1 = \{strip_1(S) \mid S \in \mathscr{S}\} = \{[\alpha_{i,1}, \beta_{i,1}] \mid i \in \{1,\dots,n\}\}.$$

Then, we compute an optimal covering point set of $\mathscr{S}_1$ using the algorithm in [23]. It is clear that $\Delta(\mathscr{S}_1) = \phi(\mathscr{S}_1) \leqslant \phi(\mathscr{S})$. Let $x_1 \leqslant \cdots \leqslant x_{l_1}$ be the l_1 ordered real points piercing $\mathscr{S}_1$ with $l_1 \leqslant \phi(\mathscr{S})$. We consider the median value x_{m} of $X = \{x_1, \dots, x_{l_1}\}$, i.e., the $[l_1/2]$th smallest value of X. We get an induced partition of $\mathscr{S}$ into three subsets $\mathscr{S}_l = \{P_i \mid \beta_{i,1} < x_{\mathrm{m}}\}$, $\mathscr{S}' = \{P_i \mid \alpha_{i,1} \leqslant x_{\mathrm{m}} \leqslant \beta_{i,1}\}$ and $\mathscr{S}_r = \{P_i \mid \alpha_{i,1} > x_{\mathrm{m}}\}$. Observe also that for any pair of polygons $P_l \in \mathscr{S}_l$ and $P_r \in \mathscr{S}_r$ then $P_l \cap P_r = \emptyset$ since $strip_1(P_l) \cap strip_1(P_r) = \emptyset$. We first recurse on $\mathscr{S}'$ until at some stage $\phi(\mathscr{S}'_1) \leqslant 1$, i.e., $\mathscr{S}_l = \mathscr{S}_r = \emptyset$. Then, we perform a recursion on the $c-1$ remaining distinct orientations.

Lemma 1. *Partition $\mathscr{Q}$ has at most $\phi(\mathscr{S}) \log_2^{c-1}(\phi(\mathscr{S})+1)$ cluster coverings.*

Proof. Let $f(\phi, c)$ denote the minimum number of covering points required for covering a set $\mathscr{S}$ of c-oriented polygons so that $\phi(\mathscr{S}) = \phi$. More precisely,

$$f(\phi, c) = \max_{\mathscr{S} \in \mathscr{P}_c} \{\tau(\mathscr{S}) \mid \phi(\mathscr{S}) = \phi \text{ and } \mathscr{S} \text{ is } c\text{-oriented}\},$$

where $\mathscr{P}_c$ is the class of c-oriented polygons of $\mathbb{E}^2$ $(c \geqslant 3)$.

From our algorithm, we get $f(\phi, c) \leqslant \min_{0 \leqslant k \leqslant \phi-2}\{f(k,c) + f(\phi-k-1,c) + f(\phi, c-1)\}$, for $\phi \geqslant 2$ and $c \geqslant 2$. If $c=1$, we have $f(\phi,1) = \phi$ [23]. Note that $f(1,c) = \mathfrak{G}(2,c)$ for $c \geqslant 1$. Let $h(\phi,c) = \phi \log_2^{c-1}(\phi+1)$. Then, we claim that $f(\phi,c) \leqslant \mathfrak{G}(2,c) h(\phi,c)$. We do an induction on the lexicographically ordered vector (c,ϕ). For $c=1$, we have $f(\phi,1) = \phi = \mathfrak{G}(2,1)\phi \log^0(\phi+1)$. If $\phi = 1$ then this means that we cannot find two disjoint c-oriented polygons and therefore $f(1,c) = \mathfrak{G}(2,c) \leqslant \mathfrak{G}(2,c)(\phi \log_2^{c-1} 2)$. Otherwise ($c>1$ and $\phi>1$), we have

$$f(\phi,c) \leqslant f\left(\frac{\phi-1}{2}, c\right) + f\left(\frac{\phi-1}{2}, c\right) + f(\phi, c-1).$$

Therefore, we get

$$h(\phi,c) \leqslant 2\frac{\phi}{2} \log_2^{c-1} \frac{\phi+1}{2} + \phi \log_2^{c-2}(\phi+1) \leqslant \phi \log_2^{c-2}(\phi+1)\left(\log_2 \frac{\phi+1}{2} + 1\right),$$

$$h(\phi,c) \leqslant \phi \log_2^{c-1}(\phi+1). \qquad \square$$

A polytope P is said *α-fat* if the ratio of the widths of a smallest hypersquare $H^+(P)$ containing P and a greatest hypersquares $H^-(P)$ included in P is bounded by α. In the following, α is considered to be a predefined constant. A set $\mathscr{S}$ is said to have the *bounded aspect ratio property* (or alternatively *β-sized*) if there exists a constant β so that $H^+(\mathscr{S})/H^-(\mathscr{S}) \leqslant \beta$, where $H^+(\mathscr{S}) = \max\{H^+(P) \mid P \in \mathscr{S}\}$ and $H^-(\mathscr{S}) = \min\{H^-(P) \mid P \in \mathscr{S}\}$.

Corollary 1. *For a set $\mathscr{S}$ of β-sized c-oriented polygons, partition $\mathscr{Q}$ has at most $\phi(\mathscr{S})\mathrm{O}(\beta)^{c-1}$ cluster coverings.*

Proof. The key idea is to analyze the left to right sequence of generated subsets $Q_1, \ldots, Q_{l_1}$ induced when separating the objects using some direction c_1. $\mathscr{S}_1$, the set of intervals obtained by considering for each polygon P_i its projection $strip_1(P_i)$, has the bounded aspect ratio property since objects of $\mathscr{S}$ are more or less the same size (i.e., O(1)-sized), denoted by $r^+ = \max_{\mathrm{O} \in \mathscr{S}}\{H^+(\mathrm{O})\}$ and $r^- = \min_{\mathrm{O} \in \mathscr{S}}\{H^-(\mathrm{O})\}$. Our goal is to prove that we can create at most $(2\lceil\beta\rceil + 1)\phi(\mathscr{S})$ sub-partitions after separating $\mathscr{S}$ with direction c_1. Note that if $\mathscr{S}$ is β-sized then so is $\mathscr{S}_1$. It suffices to notice that $Q_i \cap Q_j = \emptyset$ if $j \geqslant i + 2\lceil\beta\rceil + 1$, where $Q_i \in \mathscr{Q}_i$ and $Q_j \in \mathscr{Q}_j$. This comes from the fact that $x_{i+2} - x_i > \min_{\mathrm{O} \in \mathscr{S}}\{H^-(\mathrm{O})\}$. Thus, $x_{i+2\lceil\beta\rceil+1} - x_i > \beta r^- \geqslant r^+$. This enables us to prove that $l \leqslant \phi(\mathscr{S})\mathrm{O}(\beta)^{c-1}$ whenever $\mathscr{S}$ is a family of objects which have more or less the same size. □

Lemma 2. *The algorithm runs in* $\mathrm{O}(nc \log \phi(\mathscr{S}))$*-time using linear storage* $\mathrm{O}(nc)$.

Proof. Let $T(n,c)$ denote the running time of the algorithm. We have the recursive time-complexity system

$$T(n,c) = \begin{cases} \mathrm{O}(n \log \phi) & \text{if } c = 1, \\ \mathrm{O}(n \log \phi) + \sum_i T(n_i, c-1) & \text{otherwise} \end{cases}$$

with $\sum_i n_i = n$ (that is $\phi(\mathscr{S}_1)$ terms in the sum, with $\phi(\mathscr{S}_1) \leqslant \phi(\mathscr{S})$). Clearly, a simple induction on the lexicographic ordered vector (c,n) proves that $T(n,c) = \mathrm{O}(nc \log \phi(\mathscr{S}))$ [23]. Trivially, the algorithm uses linear storage $\mathrm{O}(N) = \mathrm{O}(nc)$. Note that this algorithm is robust to numerical tests since it performs only comparisons of algebraic degree 1. □

3.2. Piercing pairwise intersecting c-oriented polytopes

We consider below the problem in arbitrary dimension d.

3.2.1. The case of fat (c-oriented) polytopes

Let $\mathscr{S} = \{P_1, \ldots, P_n\}$ be a set of n pairwise intersecting (c-oriented) polytopes. Since $\mathscr{S}$ is α-fat, we are able to pierce $\mathscr{S}$ with $\mathrm{O}(\alpha)^d$ points. Hence, we get the following lemma:

Lemma 3. *Let $\mathscr{S}$ be a set of pairwise intersecting c-oriented α-fat polytopes then $\tau(\mathscr{S}) = \mathrm{O}(\alpha)^d \phi(\mathscr{S}) = \mathfrak{G}(2,c)$. Moreover, there exists a simple algorithm with running time $t(n,c) = \mathrm{O}_d(nc)$ for piercing $\mathscr{S}$ with $\mathrm{O}(\alpha)^d$ points.*

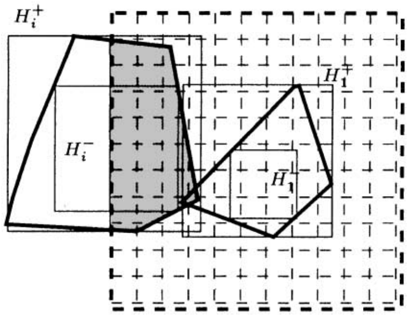

Fig. 5. Sampling $O(\alpha)^d$ covering points in $\mathbb{E}^d$.

Proof. Let $\mathscr{S}=\{P_1,\ldots,P_n\}$ be a set of n pairwise intersecting c-oriented α-fat polytopes. For each polytope P_i, let $H_i^+\,(H_i^-)$ denote a smallest enclosing axis-parallel hypercube (resp. a biggest included axis-parallel hypercube) of P_i. Let *volume*(P) denote the volume of object P. W.l.o.g. say P_1 has a smallest enclosing hypercube H_1^+ (an hypercube with the smallest edge length ρ_1 among $H_1^+,\ldots,H_n^+$). Since the polytopes of $\mathscr{S}$ intersect pairwise, we have $\bigcap_{i=1,\ldots,n} H_i^+\neq\emptyset$. Let $H=\bigcap_{i=1,\ldots,n} H_i^+$ be the intersection hypercube. Consider the 'parellel' Minkowski hypercube $2H_1^+$. (That is a homothet with homothetic ratio 2.) For each polytope P_i, we have $volume(2H_1^+)/volume(2H_1^+\cap P_i)\leqslant\gamma$, where $\gamma\leqslant O(\alpha^d)$. Indeed, let p be some point in $P_i\cap H_1^+$, and draw an axis-parallel hypercube H' centered at p with edge length $\rho_1/2$. Then it is impossible that H' contains entirely P_i, since the smallest axis-parallel hypercube containing an object in $\mathscr{S}$ has length ρ_1. Therefore since P_i is convex and α-fat we have $volume(H')/volume(H'\cap P_i)\leqslant O(\alpha^d)$. But H' is fully contained in $2H_1^+$ so that $volume(2H_1^+)/volume(H')=4^d$. Hence we deduce that

$$\frac{volume(2H_1^+)}{volume(P_i\cap 2H_1^+)}\leqslant\frac{volume(2H_1^+)}{volume(P_i\cap H')}=4^d\frac{volume(H')}{volume(H'\cap P_i)}\leqslant 4^d O(\alpha^d)=O(\alpha)^d.$$

This allows us to draw regularly $O(\alpha)^d$ points inside $2H_1^+$ using a Fredman's sampling process (we borrow ideas from [21], see Fig. 5). (The set produced by Fredman's sampling pierces any convex object that covers some fraction of a given convex region.) The running time is clearly $O(nc+\alpha^d)=O(nc)$ for α-fat objects with $\alpha=O(1)$. $\square$

Lemma 4. *Let $\mathscr{S}$ be a collection of n α-fat c-oriented polygons of $\mathbb{E}^2$. Then, we get*

$$\tau(\mathscr{S})\leqslant O(\alpha)^2\phi(\mathscr{S})\log^{c-1}(\phi(\mathscr{S})+1).$$

Proof. We apply the partition algorithm of Section 3.1 and end up with a partition $\mathscr{Q}$ of at most $\phi(\mathscr{S})\log_2^{c-1}(\phi(\mathscr{S})+1)$ cluster coverings. Since the transversal number of a cluster (i.e. a clique) is bounded by $O(\alpha)^d$ for d-dimensional fat polytopes (see Lemma 3), we deduce that the transversal number of set $\mathscr{S}$ is bounded by $O(\alpha)^2\phi(\mathscr{S})\log^{c-1}(\phi(\mathscr{S})+1)$. □

In [10], we showed that $\tau(\mathscr{S})=\phi(\mathscr{S})O(\alpha)^d$ and give various output-sensitive precision-sensitive algorithms for arbitrary fat, not necessarily c-oriented, polytopes and balls of $\mathbb{E}^3$.

3.2.2. The case of homothets

Let $\mathscr{S}=\{P_1,\dots,P_n\}$ be a set of n pairwise intersecting homothets of a c-oriented polytope. Grünbaum showed that $\mathfrak{G}(2,c)\leqslant d^d=O_d(1)$ [14].

Theorem 2. *Let $\mathscr{S}$ be a collection of c-oriented homothets in the Euclidean plane $\mathbb{E}^2$. Then,*

$$\tau(\mathscr{S})\leqslant 4\phi(\mathscr{S})\log^{c-1}(\phi(\mathscr{S})+1).$$

In the case of translates, it has been shown that $\mathfrak{G}(2,c)\leqslant(3d^{3/2})^d$ [14]. For centrally symmetric translates, the bound can be lowered to $(\sqrt{2}d)^d$. Interestingly, there is a linear-time algorithm (that is, $t(n,c)=\Theta_d(nc)$) for piercing pairwise intersecting c-oriented homothets of a polytope with at most d^d points.

3.2.3. The general case

When $\mathscr{S}$ is neither α-fat nor a collection of homothets of a given c-oriented polytope then the problem becomes far more difficult to tackle. The main open question to answer is whether $\mathfrak{G}(2,c)$ is bounded or not (assuming c as a constant parameter) for arbitrary infinitely many c-oriented polytopes. We noticed that $\mathfrak{G}(2,c)\geqslant\lceil(c-d+1)/2\rceil$ (see Fig. 1).

4. Computing an independent set

Given a graph $G=(V,E)$, an independent set I of G is a set of nodes $I\subseteq V$ such that there is no edge of E between two nodes of I. The intersection graph of a geometric set of objects is defined as follows: to each object we associate a corresponding node and we set an edge between two nodes if and only if the corresponding objects intersect. Therefore, a maximal independent set of the intersection graph corresponds to a maximal set of pairwise non-intersecting objects. Although finding a maximal independent set on general abstract graph G has been shown to be $\Omega(n^{1/4})$-hard to approximate [4] in polynomial time, we describe below a $1/[\log_2^{c-1}(\phi+1)]$ approximation heuristic based on the same partitioning scheme.

Let $\Delta'(\mathscr{S})$ be the size of an independent set found by some heuristic. Then, we have

$$\Delta'(\mathscr{S})\leqslant\Delta(\mathscr{S})\leqslant\phi(\mathscr{S})\leqslant\tau(\mathscr{S}).$$

On the other hand, we previously showed that $\Delta(\mathscr{S}) \leqslant \tau(\mathscr{S}) \leqslant \mathfrak{G}(2,c)\phi(\mathscr{S}) \log_2^{c-1}(\phi(\mathscr{S})+1)$. Taking a closer look at the complexity analysis, we are able to derive *far better* bounds;

Indeed, we may refine the analysis of Section 3.1 as

$$\Delta(\mathscr{S}) \leqslant \max_i \{b_i\} \mathfrak{G}(2,c) \log_2^{c-1}(\phi(\mathscr{S})+1),$$

where the b_i's are the maximal number of pairwise disjoint intervals of subsets of $\mathscr{S}$ induced by the partitioning scheme. We bounded the piercing number of each cluster (a clique) by a Gallai number $\mathfrak{G}(2,c)$. Considering the independent set problem, we can pick arbitrary any object inside a given cluster. Therefore, a simple heuristic consists in giving an independent set of objects of $\mathscr{S}$ from the subsets of $\mathscr{S}$ whose independent set of projected intervals is of maximal size. We get

$$\Delta(\mathscr{S}) \geqslant \Delta'(\mathscr{S}) = \max\{b_i\} \geqslant \frac{\phi(\mathscr{S})}{\mathfrak{G}(2,c) \log_2^{c-1}(\phi(\mathscr{S})+1)}.$$

A better heuristic, in practice, can be modeled as a binary tree where the root is set $\mathscr{S}$ and children correspond to recursive calls on subsets generated by partitioning the objects of $\mathscr{S}$ according to some orientation. Each internal node $\mathscr{S}$ has at most two children labeled with sets $\mathscr{S}_l$ and $\mathscr{S}_r$. We compute recursively in a bottom-to-top fashion an independent set of $\mathscr{S}$ by choosing either, at some node $\mathscr{S}$, the maximal independent set of $\mathscr{S}'$ or the union of the independent sets computed so far of sets $\mathscr{S}_l$ and $\mathscr{S}_r$. Let $\Delta'(\mathscr{S})$ denote the size of such an independent set found by this heuristic. Then, we can refine the analysis in Section 3.1 as follows:

$$\phi(\mathscr{S}) \leqslant \tau(\mathscr{S}) \leqslant \mathfrak{G}(2,c)\Delta'(\mathscr{S}) \log_2^{c-1}(\phi(\mathscr{S})+1).$$

In term of clusters, this means that $\phi(\mathscr{S}) \leqslant \Delta'(\mathscr{S}) \log_2^{c-1}(\phi(\mathscr{S})+1)$, from which we get

$$\Delta(\mathscr{S}) \geqslant \Delta'(\mathscr{S}) \geqslant \frac{\phi(\mathscr{S})}{\mathfrak{G}(2,c) \log_2^{c-1}(\phi(\mathscr{S})+1)}.$$

A program written in Java [2] demonstrates such a heuristic on planar boxes.

Finding maximal independent sets naturally arise in practice as for example in map labeling where one wants to maximize the number of labels so that they are pairwise non-intersecting (see [8, 24, 27, 1]). Agarwal et al. [1] gave a geometric polynomial-time approximation scheme for computing a maximal independent set of boxes on the plane in $O(n \log n + n^{2k-1})$ time having a performance ratio of $1 + 1/k$ for any integer $k \geqslant 1$ using the shifting lemma combined with dynamic programming.

[2] See the applet located at `http://www.csl.sony.co.jp/person/nielsen/PCD/pierce.html`.

Table 1

Class	$\tau \leqslant f(\phi)$?	Performance ratio $\tau(\mathscr{S})/\phi(\mathscr{S})$	References
Boxes	$\tau \leqslant \phi(\mathrm{o}(1) + \log\,\phi)^{d-1}$	$(\mathrm{o}(1) + \log\,\phi)^{d-1}$	[18, 23]
Fat objects	$\tau = \mathrm{O}_{d,\alpha}(\phi)$	$\mathrm{O}_{d,\alpha}(1)$	[10]
c-oriented polygons	$\tau \leqslant \mathfrak{G}(2,c)\phi \log^{c-1}(\phi+1)$	$\mathfrak{G}(2,c) \log^{c-1}(\phi+1)$	This paper
Bounded VC-dimension	None	$\mathrm{O}(\log\tau)$, $\mathrm{O}(1)$	[5]
Abstract sets	None	$\log n - \log\log n + 0.78$	[6, 26]

Table 2

Type	$\mathfrak{G}(2,c)$, $c \geqslant d+1$
Homothets	d^d
Translates	$3d^{3/2}$
Centrally symmetric translates	$(\sqrt{2}d)^d$
Fat objects	$\mathrm{O}(\alpha)^d$

5. Concluding remarks

Our precision-sensitive heuristic can easily be parallelized on the PRAM model of computation following the work of Akl and Lyons [2]. We get an $\mathrm{O}(N^{1-\varepsilon}\log\Delta(\mathscr{S}) + T(N,N^{\varepsilon}))$-time algorithm using $\mathrm{O}(N^{\varepsilon})$ processors, where $T(N,P)$ is the time required to pierce n c-oriented pairwise intersecting polygons on P processors, for any constant $\varepsilon > 0$. The main results obtained on the geometric SCP is given in Table 1.

Finally, we list some known Gallai numbers (see Table 2).

References

[1] P.K. Agarwal, Mark van Kreyeld, S. Suri, Label placement by maximum independent set in rectangles, Proc. 9th Canad. Conf. on Computational Geometry, 1997, pp. 233–238.
[2] S.G. Akl, K.A. Lyons, Parallel Computational Geometry, Prentice-Hall, Englewood Cliffs, NJ, 1993.
[3] S. Arora, D. Karger, M. Karpinski, Polynomial-time approximation schemes for dense instances of NP-hard problems. Symp. on Theory of Computing, 1995, pp. 284–293.
[4] M. Bellare, M. Sudan, Improved non-approximability results. Proc. 26th Annu. ACM Symp. on Theory of Computing, 1994, pp. 184–193.
[5] H. Brönnimann, M.T. Goodrich, Almost optimal set covers in finite VC-dimension, Proc. 10th Annu. ACM Symp. on Computational Geometry, 1994, pp. 293–302.
[6] V. Chvátal, A greedy heuristic for the set-covering problem, Math. Oper. Res. 4 (1979) 233–235.
[7] G.L. Ding, P. Seymour, P. Winkler, Bounding the vertex cover number of a hypergraph, Combinatorica 14 (1) (1994) 23–34.
[8] S. Doddi, M. Marathe, A. Mirzaian, B. Moret, B. Zhu, Map labeling and its generalization, Proc. 8th ACM-SIAM Symp. on Discrete Algorithms, 1997, pp. 148–157.
[9] R.-chii Duh, M. Fürer, Approximation of k-set cover by semi-local optimization, Proc. 29th Annu. ACM Symp. on Theory of Computing, 1997, pp. 256–264.
[10] A. Efrat, Matthew J. Katz, F. Nielsen, M. Sharir, Dynamic data structures for fat objects and their applications, Proc. Work. Alg. Data Struct. 97, Halifax, Nova Scotia, Canada, Lecture Notes in Computer Science, Springer, Berlin, 1997, pp. 297–306.

[11] U. Feige, A threshold of $\ln n$ for approximating set cover (preliminary version), Proc. 28th Annu. ACM Symp. on Theory of Computing, Philadelphia, Pennsylvania, 22–24 May, 1996, pp. 314–318.

[12] D.G. Fon der Flass, A.V. Kostochka, Covering boxes by points, Discrete Math. 120 (1993) 269–275.

[13] R.J. Fowler, M.S. Paterson, S.L. Tanimoto, Optimal packing and covering in the plane are NP-complete, Inform. Process. Lett. 12 (3) (1981) 133–137.

[14] B. Grünbaum, On intersections of similar sets, Portugal Math. 18 (1959) 155–164.

[15] D.S. Hochbaum, Approximation algorithms of the set covering and vertex cover problems, SIAM J. Comput. 11 (3) (1982) 555–556.

[16] D.S. Hochbaum, W. Maass, Approximations schemes for covering and packing problems in image processing and VLSI, J. ACM 31 (1984) 130–136.

[17] D.S. Hochbaum, W. Maass, Fast approximation algorithms for a nonconvex covering problem, J. Algorithms 8 (1987) 305–323.

[18] G.Y. Károlyi, On point covers of parallel rectangles, Period. Math. Hungar. 23 (2) (1991) 105–107.

[19] G. Károlyi, G. Tardos, On point covers of multiples intervals and axis-parallel rectangles, Technical Report, DIMACS TR 95-45, 1995.

[20] M. Karpinski, A. Zelikovsky, Approximating dense cases of covering problems, Technical Report, DIMACS TR 96-59, 1996.

[21] M.J. Katz, 3-D vertical ray shooting and 2-D point enclosure, range searching, and arc shooting amidst convex fat objects, Comput. Geom. Theory Appl. 8 (1997) 299–316.

[22] M.J. Katz, F. Nielsen, On piercing sets of objects, Proc. 12th Annu. ACM Symp. on Computational Geometry, 1996, pp. 113–121.

[23] F. Nielsen, Fast stabbing of boxes in high dimensions. Proc. 8th Canad. Conf. on Computational Geometry, 1996, pp. 87–92.

[24] C. Poon, B. Zhu, F. Chin, A polynomial time solution for labeling a rectilinear map. Proc. 13th Annu. ACM Symp. on Computational Geometry, 1997, pp. 451–453.

[25] F.P. Preparata, M.I. Shamos, Computational Geometry: An Introduction., Springer, New York, 1985.

[26] P. Slavík, A tight analysis of the greedy algorithm for set cover, Proc. 28th ACM Symp. on Theory of Computing, 1996.

[27] F. Wagner, A. Wolff, A practical map labeling algorithm, Comput. Geom. Theory Appl. 7 (1997) 387–404.