How to Place Efficiently Guards and Paintings in an Art Gallery

Christodoulos Fragoudakis*[†], Euripides Markou*[‡] and Stathis Zachos[§]

April 25, 2005

Abstract

In the art gallery problem the goal is to place guards (as few as possible) in a polygon so that a maximal area of the polygon is covered. We address here a closely related problem: how to place paintings and guards in an art gallery so that the total value of guarded paintings is a maximum. More formally, a simple polygon is given along with a set of paintings. Each painting, has a length and a value. We study how to place at the same time: i) a given number of guards on the boundary of the polygon and ii) paintings on the boundary of the polygon so that the total value of guarded paintings is maximum. We investigate this problem for a number of cases depending on: i) where the guards can be placed (vertices, edges), ii) whether the polygon has holes or not and iii) whether the goal is to oversee the placed paintings (every point of a painting is seen by at least one guard), or to watch the placed paintings (at least one point of a painting is seen by at least one guard). We prove that the problem is NP-hard in all the above cases and we present polynomial time approximation algorithms for all cases, achieving constant ratios.

Keywords: Visibility, Computational Geometry, Approximation Algorithms.

Designated Track: Research Paper.

1 Introduction

In the Art Gallery problem, a polygon is given and the goal is to place as few as possible guards in the polygon, so that a maximal area of the polygon is covered. This is a well known problem having the variation where a number of guards is given and the goal is to cover as many points in the polygon as possible. More variations arise when the polygon has holes and the points that must be covered lie in general on the boundary of the

*Christodoulos Fragoudakis and Euripides Markou wish to acknowledge partial support by PYTHAGORAS, project 70/3/7392 under the EPEAEK program of the Greek Ministry of Educational and Religious Affairs

†Computer Science, ECE, National Technical University of Athens, `cfrag@cs.ntua.gr`

‡Department of Informatics and Telecommunications, National University of Athens, `emarkou@cs.ntua.gr`

§CIS Department, Brooklyn College, CUNY, USA, and Computer Science, ECE, National Technical University of Athens, `zachos@cs.ntua.gr`

polygon and of its holes. On the other hand guards may be realized as vertices (vertex guards) or whole edges (edge guards) of the polygon ([2, 4, 5]).

We address here a closely related problem: how to place exhibits like paintings and guards in an art gallery so that the total value of guarded paintings is a maximum. More formally, a polygon is given along with a set of paintings. Each painting has a length and a value. We study how to place simultaneously a given number of guards and the paintings on the boundary of the polygon so that the total value of guarded paintings is maximum.

Some related problems that have been studied: MINIMUM VERTEX/EDGE/POINT GUARD for polygons with (without) holes (known to be APX-hard and $O(\log n)$ approximable [1, 6, 7]), MINIMUM FIXED HEIGHT VERTEX/POINT GUARD ON TERRAIN (best approximation possible $\theta(\log n)$ [6, 7, 8]), MAXIMUM WEIGHTED CLIQUE ON VISIBILITY GRAPH (known to be in P [11, 12, 13]), MINIMUM CLIQUE PARTITION ON VISIBILITY GRAPH for polygons without holes (known to be APX-hard and $O(\log n)$ approximable [6]). We prove that our problem is NP-hard and give polynomial time algorithms achieving a constant approximation ratio, based on a well known greedy algorithm which approximates the MAXIMUM COVERAGE problem ([9, 10]).

The remainder of this paper is organized as follows: In section 2 we define the *Finest Visibility Segmentation* (FVS) of a polygon and settle that this construction is the finest relevant segmentation with respect to visibility: a FVS segment cannot be only partly visible from a vertex or an edge. In section 3 we define the MAXIMUM VALUE VERTEX GUARD WITH PAINTING PLACEMENT problem and prove that the problem is NP-hard. We present a polynomial time algorithm that achieves a constant approximation ratio and extend the result for edge guards and polygons with holes. Finally, in section 4 we present the conclusion.

2 Finest Visibility Segmentation

We start with some preliminary definitions (see figure 1). Let P be a polygon, $a, b \in P$ two points inside P and $L, M \subseteq P$ two sets of points inside P. We say that point a *sees* point b, i.e. a and b are mutually visible, if the straight line segment connecting a and b lies everywhere inside P. Notice that if point a sees point b then also point b sees point a. We say that the point set L is *visible* from the point set M or that M *oversees* L if for all points that belong to L, there exist a point that belongs to M, such that the points are mutually visible. Notice that if M oversees L, it is not necessary for L to oversee M. Finally, we say that M *watches* L if there exist a point that belongs to L and a point that belongs to M such that the points are mutually visible. Notice that if M watches L then also L watches M.

We are going to describe a method that descritizes the boundary as well as the interior of any polygon in terms of visibility. Assume any polygon P and the corresponding visibility graph $V_G(P)$: the visibility graph's vertex set is the vertex set of the polygon and two vertices share an edge in the visibility graph if and only if they are mutually visible in the polygon. By extending the edges of $V_G(P)$ inside P up to the boundary of P, see figure 2(a), we obtain a set of points FVS of the boundary of P, see figure 2(b), that includes of course all vertices of P. An extended edge of $V_G(P)$ generates at most two FVS points and there are $O(n^2)$ edges in $V_G(P)$, so there are $O(n^2)$ points in any

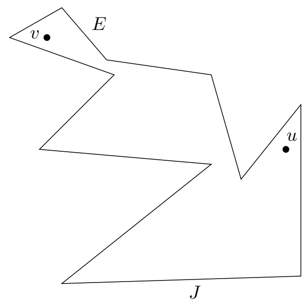
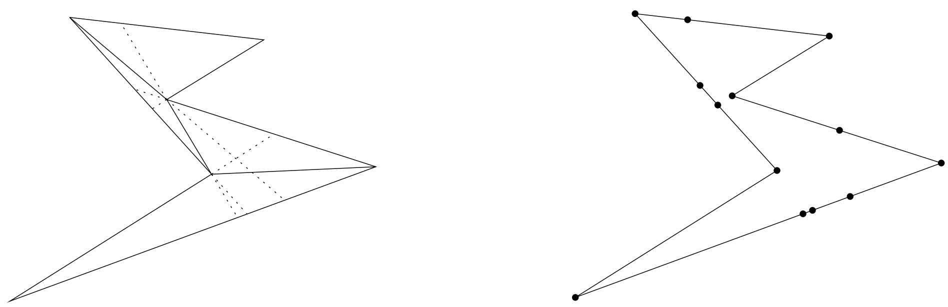


Figure 1: Guards v and u are not mutually visible, guard v oversees edge E and guard u watches edge J.

polygon's FVS set. We call this construction the *Finest Visibility Segmentation* of the polygon P. Any *open* segment (a, b), i.e. a and b are excluded, defined by consecutive FVS points, is called an FVS segment of P. The following two lemmas settle that a FVS segment cannot be **only partly** visible from a vertex or an edge.



(a) (b)

Figure 2: Discretizing the boundary of a polygon

Lemma 1 *For any vertex v of the polygon P, an* open *segment (a, b) defined by consecutive FVS points a, b, is visible by v if and only if it is watched by v.*

Proof. Of course if (a, b) is visible from v, then it is watched by v. Suppose now that (a, b) is watched by v but not overseen by v. Without loss of generality assume that v sees only (c, d) and cannot see any point between a and c, as well as, cannot see any point between d and b, see figure 3(a). So there must be an edge with endpoint vertex u that blocks v's visibility left of c and another vertex w that blocks v's visibility right of d that is $vu, vw \in V_G(P)$. The extensions of vu and vw meet the boundary at c, d respectively, hence $c, d \in FVS$. So a, b cannot be consecutive FVS points. $\square$

Lemma 2 *For any edge e of the polygon P, an* open *segment (a, b) defined by consecutive FVS points a, b, is visible by e if and only if it is watched by e.*

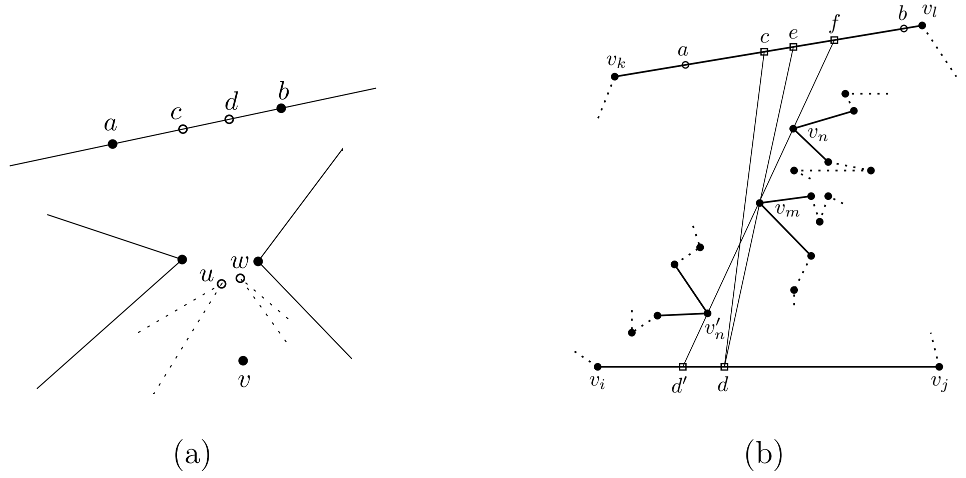


Figure 3: Any vertex (edge) oversees a *FVS* segment if and only if watches the segment

Proof. Of course if (a, b) is visible from e, then it is watched by e. Suppose now that (a, b) is watched by $e = (v_i, v_j)$ but not overseen by e. The fact that e watches (a, b) implies that there exist a point $c \in (a, b)$ and a point $d \in e$ such that the line segment cd is everywhere inside the polygon, see figure 3(b). Start now an angular sweep of the line that passes through c and d, around d. If the sweep reaches b we have overseen all the cb segment, otherwise we stop at the vertex that belongs to the first edge that blocks d's visibility to (a, b) (vertex v_m in figure 3(b)). So point d oversees the ce segment and in order to see further towards b we have to consider points right of d. We start a new angular sweep of the line that passes through e and d, around v_m, that passes through different positions d' left of d. Consider the following cases:

- The sweep reaches b so we have overseen all of the cd segment.
- The sweep stops at the vertex that belongs to the first edge that blocks d''s visibility to (a, b) (vertex v_n or v'_n in figure 3(b)). This means that there is a line segment that starts from d' passes through the v_m and v_n (or v'_n) vertices and is everywhere inside P. The latter means that $v_m v_n \in V_G(P)$ and $v_m v_n$'s extension intersects P's boundary to f, hence f is a *FVS* point left of b, so a and b cannot be consecutive *FVS* points.
- The sweep reaches v_i but not b. This means that there is a line segment that starts from v_i passes through v_m, intersects (a, b) at f and is everywhere inside the polygon P. But the latter means that $v_i v_m$ is an edge of the visibility graph $V_G(P)$ and its extension intersects the boundary of P to point f, hence f is a *FVS* point left of b. So a and b cannot be consecutive *FVS* points.

Using the exact reasoning we can prove that e oversees also the ac segment. □

The above lemmas settle the following:

Theorem 1 *The boundary of any polygon P can be effectively descritized in terms of visibility to $O(n^2)$ FVS segments. Any vertex (edge) of P sees a FVS segment if and only if watches the FVS segment.*

In order to find the set of all overseen FVS segments from a polygon vertex v, namely the $FVS(v)$ set, (using lemma 1) it suffices to pick an arbitrary point p in every FVS segment. For every selection of p we have to check if vp is everywhere inside the polygon P. If this is the case, we augment the $FVS(v)$ set with the relative FVS segment. Notice that the segment inclusion in the polygon can be effectively checked in polynomial time by simple orientation tests, hence the construction of the $FVS(v)$ set costs polynomial time.

For the case of the $FVS(e)$ set, that is the set of all overseen FVS segments from a polygon edge e, (using lemma 2) it suffices to pick an arbitrary point p in every FVS segment and check if there exists a point $p' \in e$ such that pp' is everywhere inside P. This can be done by an angular sweep manner around p, stopping at the polygon vertices. If there exists a vertex v_i (possibly an endpoint of e) such that the extension of pv_i intersects e to p', we augment the $FVS(e)$ set with the relative FVS segment. Notice that again the construction of the $FVS(e)$ set costs polynomial time.

3 The Maximum Value Vertex Guard with Painting Placement problem

Given is a polygon P, a set of ordered pairs (x, y) and an integer $k > 0$. The boundary of the polygon models the walls of an art gallery while an ordered pair (x, y) represents a painting with length x and value y. The goal of the Maximum Value Vertex Guard with Painting Placement problem is to place k vertex guards as well as place paintings on the boundary of P so that the total weight of the overseen paintings is maximum. Notice that in the given set of paintings $\{(x_1, y_1), \dots, (x_i, y_i)\}$, there is an unlimited number of paintings of length x_1 and value y_1. Another restriction is that if there is an area on the boundary overseen by at least two guards, then all paintings that have their parts in this area should be overseen by the same guard. We call this problem Maximum Value Vertex Guard with Painting Placement. In the following we prove that it is NP-hard to place guards on the vertices (at most k) and paintings from the given set with respect to the above restriction so that the total value of the overseen paintings is maximum.

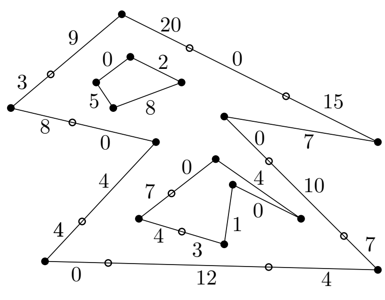


Figure 4: A possible placement of paintings on the walls of the art gallery (zero value means that there is no painting placed).

Proposition 1 *Consider a polygon P along with a set of paintings (each painting has a length and a value assigned) and integers $k, V > 0$. It is NP-hard to decide whether we can place at most k guards on vertices of P and paintings on the walls of the gallery (boundary of P) so that the total value of overseen paintings is at least V.*

Proof. The decision version of MINIMUM VERTEX GUARD for a polygon P reduces to the corresponding decision version of MAXIMUM VALUE VERTEX GUARD WITH PAINTING PLACEMENT. We construct an instance of the last problem as follows: We take the same polygon P. We construct the FVS, i.e. the finest line segment subdivision of the edges of P, and take a painting for each FVS segment with length the length of the segment and value also the length of the segment. Finally we take as V the total sum of values of the segments. Now the truth of the proposition is straightforward. $\square$

Algorithm 1 is an approximation algorithm for the MAXIMUM VALUE VERTEX GUARD WITH PAINTING PLACEMENT problem.

Algorithm 1 *MaxValueVertexGuardWithPaintingPlacement (greedy)*

compute the FVS points
foreach $v \in V(P)$
    compute $FVS(v)$
$SOL := \emptyset$
for $i := 1$ **to** k
    select $v \in V$ that maximizes $W(\texttt{multknap}(FVS(v) \setminus SOL \cap FVS(v), D))$
    update SOL
return $W(SOL)$

In algorithm 1 we use the MULTIPLE KNAPSACK problem. This problem is known to be NP-hard and there is a polynomial time 2-approximation algorithm for it ([3]). Actually in [14] they proved that there is a PTAS for the MULTIPLE KNAPSACK problem.

Algorithm 1 starts by calculating the FVS points and then for every $v \in V(P)$ the set $FVS(v)$. During each iteration of the algorithm, for any vertex v that hasn't been assigned a guard yet, the set $FVS(v) - SOL \cap FVS(v)$ (of the visible segments not previously overseen) is calculated. Then for every such set, the MULTIPLE KNAPSACK problem is solved, taking as knapsacks the segments in the set (the capacity of a knapsack is the length of the corresponding segment). The solution of the MULTIPLE KNAPSACK problem results to a placement of paintings of a total maximum value in the knapsacks. The vertex that maximizes the total value of the fitted paintings is then chosen, causing an overall increase of $1 - \epsilon$ of the maximum possible increase of the solution, due to the PTAS of the MULTIPLE KNAPSACK problem. Then the algorithm updates the set SOL by adding the new FVS segments along with the fitted paintings.

In order to prove that algorithm 1 approximates MAXIMUM VALUE VERTEX GUARD WITH PAINTING PLACEMENT by a constant approximation factor, let OPT denote the collection of the set of paintings in an optimal solution and SOL denote the collection returned by the algorithm. These collections have $W(OPT)$ and $W(SOL)$ values respectively. Suppose that the algorithm places a guard at vertex v_i at iteration i, and a set of new paintings P_i. Therefore the added total value of paintings at iteration i is $W(P_i)$.

In the ordered sequence of vertices (as they have been selected by the algorithm), consider the first vertex v_l selected by the algorithm but not by the optimal solution for placing a guard. In other words, let v_l be the first vertex in the ordered sequence where there is a guard placed by the algorithm but there is no guard in the optimal solution. It holds:

$$W(P_i) = W(\cup_{m=1}^{i} P_m) - W(\cup_{m=1}^{i-1} P_m)$$

The PTAS for the MULTIPLE KNAPSACK problem implies:

$$W(P_i) \geq \alpha W(P_i'), \alpha > 0$$

where $W(P_i')$ is the new total value overseen by a guard placed on v_i in the optimal solution. We settle the following lemmas:

Lemma 3 *After l iterations of algorithm 1 the following holds:*

$$W(\cup_{i=1}^{l} P_i) - W(\cup_{i=1}^{l-1} P_i) \geq \frac{\alpha}{k}(W(OPT) - W(\cup_{i=1}^{l-1} P_i)), \;\; l = 1, 2, ..., k$$

Proof. Consider vertices where guards have been placed in the optimal solution but no guard has been placed there by the algorithm. By the pigeonhole principle, there is at least one such vertex v_m so that the following holds:

$$W(P_m') \geq \frac{W(OPT) - W(\cup_{i=1}^{l-1} P_i')}{k}$$

since $W(P_i) \geq \alpha W(P_i')$, it holds:

$$W(P_m') \geq \frac{W(OPT) - W(\cup_{i=1}^{l-1} P_i)}{\alpha k}$$

Notice that

$$W(P_l) \geq W(P_m) \geq \alpha W(P_m')$$

and

$$W(\cup_{i=1}^{l} P_i) - W(\cup_{i=1}^{l-1} P_i) = W(P_l) \geq \alpha W(P_l')$$

Therefore:

$$W(\cup_{i=1}^{l} P_i) - W(\cup_{i=1}^{l-1} P_i) \geq \frac{\alpha}{k}(W(OPT) - W(\cup_{i=1}^{l-1} P_i))$$

□

Lemma 4 *After l iterations of algorithm 1 it holds:*

$$W(\cup_{i=1}^{l} P_i) \geq (1 - (1 - \frac{\alpha}{k})^l) W(OPT), \;\; l = 1, ..., k$$

Proof. We are going to prove this by induction on l During the first step the algorithm chooses the set with value $W(P_1)$. It holds:

$$W(P_1) \geq \alpha W(P_1')$$

$W(P_1')$ is the biggest possible value that OPT achieves, so from the pigeonhole principle:

$$W(P_1') \geq \frac{W(OPT)}{k} \rightarrow W(P_1) \geq \frac{\alpha}{k}W(OPT)$$

Assume that the given holds for $i = l - 1$:

$$W(\cup_{i=1}^{l-1} P_i) \geq (1 - (1 - \frac{\alpha}{k})^{l-1})W(OPT)$$

So:

$$W(\cup_{i=1}^{l} P_i) = W(\cup_{i=1}^{l-1} P_i) + (W(\cup_{i=1}^{l} P_i) - W(\cup_{i=1}^{l-1} P_i))$$

Using lemma 3:

$$W(\cup_{i=1}^{l} P_i) \geq W(\cup_{i=1}^{l-1} P_i) + \frac{\alpha}{k}(W(OPT) - W(\cup_{i=1}^{l-1} P_i)) \rightarrow$$

$$W(\cup_{i=1}^{l} P_i) \geq W(\cup_{i=1}^{l-1} P_i)(1 - \frac{\alpha}{k}) + \frac{\alpha}{k}W(OPT)$$

From the inductive hypothesis:

$$W(\cup_{i=1}^{l} P_i) \geq (1 - (1 - \frac{\alpha}{k})^{l-1})W(OPT)(1 - \frac{\alpha}{k}) + \frac{\alpha}{k}W(OPT) \rightarrow$$

$$W(\cup_{i=1}^{l} P_i) \geq (1 - (1 - \frac{\alpha}{k})^{l})W(OPT)$$

□

Theorem 2 *Algorithm 1 runs in polynomial time and achieves an approximation of* $\frac{1}{1-\frac{1}{e^\alpha}}$ *with respect to the optimum of the* Maximum Value Vertex Guard with Painting Placement *problem, where* $\frac{1}{\alpha}$ *is the approximation ratio of* Multiple Knapsack.

Proof. Using lemma 4, we set $l = k$ and get:

$$W(\cup_{i=1}^{k} P_i) \geq (1 - (1 - \frac{\alpha}{k})^{k})W(OPT)$$

It holds:

$$\lim_{k \to \infty}(1 - (1 - \frac{\alpha}{k})^{k}) = 1 - \frac{1}{e^\alpha}$$

As $(1 - (1 - \frac{\alpha}{k})^k)$ continuously gets smaller, we have:

$$1 - (1 - \frac{\alpha}{k})^k \geq 1 - \frac{1}{e^\alpha}$$

So:

$$W(SOL) > (1 - \frac{1}{e^\alpha})W(OPT)$$

That is the algorithm approximates the Maximum Value Vertex Guard with Painting Placement problem with a $\frac{1}{1-\frac{1}{e^\alpha}}$ ratio. Due to the existence of the PTAS for the Multiple Knapsack problem, $\alpha \rightarrow 1$ so $\frac{1}{1-\frac{1}{e^\alpha}} \rightarrow 1.58$. □

Similar to proposition 1, for the case of edge guards, it holds:

Proposition 2 *The* Maximum Value Edge Guard with Painting Placement *problem is NP-hard.*

Algorithm 2 approximates Maximum Value Edge Guard with Painting Placement. The only difference from algorithm 1 is that we need to calculate the $FVS(e)$ set using the techniques described in section 2.

Algorithm 2 *MaxValueEdgeGuardWithPaintingPlacement (greedy)*

compute the FVS points
foreach $e \in E(P)$
    compute $FVS(e)$
$SOL := \emptyset$
for $i := 1$ **to** k
      select $e \in E$ that maximizes $W(\texttt{multknap}(FVS(e) \setminus SOL \cap FVS(e), D))$
      update D
return $W(SOL)$

Similar to theorem 2 it holds:

Theorem 3 *Algorithm 2 runs in polynomial time and achieves an approximation of* 1.58 *for the* Maximum Value Edge Guard with Painting Placement *problem.*

Similar results apply also for the case of polygons with holes. Algorithms 1 and 2 can be applied to polygons with holes, achieving the same approximation.

4 Conclusion

We investigated the Maximum Value Vertex Guard with Painting Placement problem: we proved NP-hardness and presented a polynomial time algorithm that achieves a constant approximation ratio. The algorithm applies for a number of cases (edge guards, polygons with holes) and achieves the same approximation. While investigating the above problem we used a way to discretize the boundary of the polygon by subdividing it into $O(n^2)$ pieces of the *Finest Visibility Segmentation* which is the finest releavant segmentation with respect to visibility: a FVS segment cannot be only partly visible from a vertex or an edge.

References

[1] Lee, D., Lin, A., Computational complexity of art gallery problems, IEEE Trans. Inform. Theory 32, 276-282, 1986.

[2] O'Rourke, J., Art Gallery Theorems and Algorithms, Oxford University Press, New York, 1987.

[3] Shmoys, D., Tardos, E., An approximation algorithm for the generalized assignment problem, Mathematical Programming A, 62:461-74, 1993.

[4] Shermer, T., Recent results in Art Galleries, Proc. of the IEEE, 1992.

[5] Urrutia, J., Art gallery and Illumination Problems, Handbook on Computational Geometry, 1998.

[6] Eidenbenz, S., (In-)Approximability of Visibility Problems on Polygons and Terrains, PhD Thesis, ETH Zurich, 2000.

[7] Eidenbenz, S., Inapproximability Results for Guarding Polygons without Holes, Lecture notes in Computer Science, Vol. 1533 (ISAAC'98), p. 427-436, 1998.

[8] Ghosh, S., Approximation algorithms for Art Gallery Problems, Proc. of the Canadian Information Processing Society Congress, pp. 429-434, 1987.

[9] Hochbaum, D., Approximation Algorithms for NP-Hard Problems, PWS Publishing Company, 1996.

[10] Hochbaum, D., Approximating Covering and Packing Problems in: Approximation Algorithms for NP-Hard Problems (ed. Dorit Hochbaum), pp. 94-143, PWS Publishing Company, 1996.

[11] Avis, D., Rappaport, D., Computing the largest empty convex subset of a set of points, Proc. 1st Ann. ACM Symposium Computational Geometry, pp. 161-167, 1985.

[12] Dobkin, D., Edelsbrunner, H., Overmars, H., Searching for Empty Convex Polygons, Algorithmica 5, pp. 561-571, 1990.

[13] Edelsbrunner, H., Guibas, L., Topologically sweeping an arrangement, J. Comput. System Sci. 38, pp. 165-194, 1989.

[14] Chekuri, C., Khanna, S., A PTAS for the Multiple Knapsack problem, Proc. 11th ACM Symp on Discrete Algorithms, 213-222, 2000.