Budgeted Coverage of a Maximum Part of a Polygonal Area

Euripides Markou[1], Stathis Zachos[1,2] and Christodoulos Fragoudakis[1]

[1] Computer Science, ECE, National Technical University of Athens
[2] CIS Department, Brooklyn College, CUNY
email{emarkou, zachos, cfrag}@cs.ntua.gr

Abstract. Given is a polygonal area and the goal is to cover the polygon's interior or boundary by placing a number of stations in the interior or on the boundary. There are several parameters for this problem: what part of the polygon (interior, boundary) is to be covered, and where can stations be placed (vertices, edges, interior points); Furthermore there are minimization problems (minimum number of stations) or maximization problems (given the number of stations, maximize the covered portion of the polygon). Most of these problems are NP-hard and some are even APX-hard. We consider here a new more realistic version: the boundary is subdivided into segments with lengths and costs and the meaning of covering is relaxed and budgeted. More specifically we consider maximization problems in polygons with weighted edge segments and placement costs. We introduce a new concept: each candidate place for a station has a cost. We study the following two problems: Given a polygon with or without holes and a budget B, place stations so that the total cost of stations does not exceed B and a) the *length* of the boundary covered is maximized or b) the *total weight* of segments *watched* or *overseen* is maximized. We present constant ratio approximation algorithms for all the above problems.
Keywords: Wireless Communication, Point to Station Communication, Computational Geometry, Approximation Algorithms, Visibility.

1 Introduction

The problem of covering a polygonal area by placing a minimum number of stations is central in the development of wireless communication technology. An area is considered covered if every point of the area can communicate with at least one station (two points can communicate if they are mutually visible). The goal is to cover the whole area using a minimum number of stations. We can also find this problem under the name 'Art Gallery'. In the Art Gallery problem the goal is to place a minimum number of guards in a gallery so that every point in the interior of the gallery can be seen by at least one guard.

Many variations of the Art Gallery problem have been studied during the last two decades [2–4]. These variations can be classified with respect to where the guards are allowed to be placed (e.g. on vertices, edges, interior of the polygon) or whether only the boundary or all of the interior of the polygon needs to be guarded, etc. Most known variations of this problem are NP-hard. Related problems that have been studied are Minimum Vertex/Edge/Point Guard for polygons with or without holes

(APX-hard and O($\log n$)-approximable [1, 5, 6]) and Minimum Fixed Height Vertex/Point Guard On Terrain ($\Theta(\log n)$-approximable [5–7]). In [9] the case of guarding the walls (and not necessarily every interior point) is studied. In [10] the following problem has been introduced: suppose we have a number of valuable treasures in a polygon P; what is the minimum number of mobile (edge) guards required to patrol P in such a way that each treasure is always visible from at least one guard? In [10] they show NP-hardness and give heuristics for this problem. In [11] weights are assigned to the treasures in the gallery. They study the case of placing one guard in the gallery in such a way that the sum of weights of the visible treasures is maximized. Recent (non-)approximability results for art gallery problems can be found in [1–5, 7]. For a nice survey of approximation classes and important results the reader is referred to [8].

Here we consider two new problems and we introduce a new concept which makes the general problem more realistic: every candidate position of a station (guard) in the polygon has been assigned a cost. The first problem we investigate here, is the placement of stations on the boundary of the polygon (vertices or edges) so that a maximum length of the boundary is covered. We can place as many stations as long as the total cost does not exceed a given budget. The second problem we study is the following: the given polygon has its boundary subdivided into weighted disjoint line segments. The goal is to maximize the total value of the overseen segments by placing stations on vertices or edges so that the total cost of stations is within a given budget.

A segment is *overseen* or *visible* or *covered* iff every point of the segment is visible by at least one station. Two points are mutually visible iff the straight line segment connecting them lies (everywhere) inside the polygon.

We also investigate what happens when the goal is simply to *watch* a maximum total value of segments. We consider a segment watched if at least a part of it is overseen by a station.

Besides the applications in an art gallery where the interpretation of line segments are valued paintings, there are also important applications in wireless communication networks: An interpretation of weighted line segments are inhabited areas. The polygon models the geographical space. The weight interpretation is the population of an area. Imagine a number of towns lying on the boundary of a polygonal geographical area. The goal is to place a number of stations such that the total number of people that can communicate is maximized. The total cost of the stations must be within a budget B. Moreover, it could be the case that the towns are on the shore of a lake, so we can only place stations on the boundary. Similar situations may arise in various other types of landscape. A possible interpretation of an edge station (guard) is a mobile station (guard) which is moving between the edge's endpoints.

The two problems above have been studied in [12] for the case in which a cost 1 has been assigned to each candidate position for a station. However, the polynomial time approximation algorithms that have been presented there do not apply to the problems we study here.

We present here polynomial time approximation algorithms achieving constant ratios for the two problems based on a well known greedy algorithm which approximates the Budgeted Maximum Coverage problem ([14]).

2 The Budgeted Maximum Length Vertex/Edge Guard problem for polygons with (or without) holes

Suppose a polygon P with (or without) holes is given with costs on vertices, along with a number $B > 0$. We are asked to cover a maximum portion of the polygon's boundary (including boundaries of the possible holes), using guards that have total cost no more than B. We are allowed to use either only vertex guards or only edge guards (occupying whole edges).

Definition 1 *Given is a polygon P with or without holes with costs on vertices (or edges) and a number $B > 0$. Let $L(b)$ be the euclidean length of the line segment b. The goal of the* Budgeted Maximum Length Vertex/Edge Guard *problem is to place vertex (or edge) guards so that the euclidean length of that part of P's boundary that is* overseen *by the guards is maximum and the total cost of vertices (or edges) with guards is at most B.*

In [12] we introduced the Maximum Length Vertex/Edge Guard problem, for polygons with or without holes and proved that is NP-hard. Maximum Length Vertex/Edge Guard is a special case of Budgeted Maximum Length Vertex/Edge Guard where each vertex (edge) of the polygon has cost 1.

Fact 1 Budgeted Maximum Length Vertex/Edge Guard *for polygons with (or without) holes is NP-hard.*

First we will discuss the case of polygons with or without holes where guards can be placed only on vertices of the polygon.

In [12] we described an approximation algorithm for Maximum Length Vertex Guard for polygons with or without holes. However, it is easy to verify that this algorithm, does not approximate the budgeted case. We present here a new polynomial time approximation algorithm which achieves a constant ratio for the budgeted case. First we construct for every $v \in V(P)$ a set $E'(v)$ which is the set of line segments on the boundary visible from vertex v. We use the FVS construction which we introduced in [12] to discretize the boundary of the polygon into segments that are visible iff watched by a vertex guard. We recall from [12]: 'We use the visibility graph $V_G(P)$. By extending edges of $V_G(P)$ inside P up to the boundary of P we obtain a set of points FVS of the boundary of P (that includes of course all vertices) (see figure 1). There are $O(n^2)$ points in FVS (= finest visibility segmentation) and these points are endpoints of line segments with the following property: for any vertex $v \in V(P)$, a segment (a, b) defined by consecutive FVS points a, b is visible by v iff it is watched by v'. Remember that a segment is watched by a point if at least a part of it is visible by the point. In every step of the greedy algorithm which maximizes the overall overseen boundary, a guard is placed on a vertex so that a maximum increase in $\frac{length}{cost}$ ratio is achieved.

Let OPT denote the collection of the sets of the overseen FVS segments of an optimal solution. Let SOL denote the collection returned by the algorithm. Let r be the number of iterations executed by the algorithm until the first set S_{l+1} belonging

Algorithm 1 BudgetedMaxLegthVertexGuards (* greedy *)

$SOL \leftarrow \emptyset$
$G \leftarrow \emptyset$
$C_{TOT} \leftarrow 0$
$M \leftarrow V$
repeat
  select $v_i \in V$ that maximizes $\frac{L(G \cup E'(v_i))}{c_i}$
  if $C_{TOT} + c_i \leq B$ **then**
    $G \leftarrow G \cup E'(v_i)$
    $C_{TOT} \leftarrow C_{TOT} + c_i$
  end if
  $M \leftarrow M - \{v_i\}$
until $M = \emptyset$
select $v_t \in V$ that maximizes $L(E'(v_t))$
if $L(G) \geq L(E'(v_t))$ **then**
  $SOL \leftarrow G$
else
  $SOL \leftarrow E'(v_t)$
end if
return $L(SOL)$

the OPT collection is found but is not added to SOL because its addition would violate budget B. We denote:

$$L(S_k) = L(\cup_{i=1}^{k} S_i) - L(\cup_{i=1}^{k-1} S_i)$$

Thus the set S_k does not contain segments which have already been covered. Actually the set S_k is the k-th set $E'(v_i)$ selected by the algorithm. We also sometimes call cost of a set of segments, the cost of the vertex guard which oversees them.

Lemma 1. *After l iterations it holds:*

$$L(\cup_{i=1}^{l} S_i) - L(\cup_{i=1}^{l-1} S_i) \geq \frac{c_l}{B}(L(OPT) - L(\cup_{i=1}^{l-1} S_i))$$

Proof. For every set $S_k \in OPT - \cup_{i=1}^{l-1} S_i$ it holds:

$$\frac{L(S_k)}{c_k} \leq \frac{L(S_l)}{c_l}$$

The total cost of the sets in $OPT - \cup_{i=1}^{l-1} S_i$ is at most B. The total length of the segments of the sets in $OPT - \cup_{i=1}^{l-1} S_i$ is:

$$L(\cup_{i=m}^{k} S'_i)$$

Since

$$\frac{L(S'_m)}{c_m} \leq \frac{L(S_l)}{c_l}$$

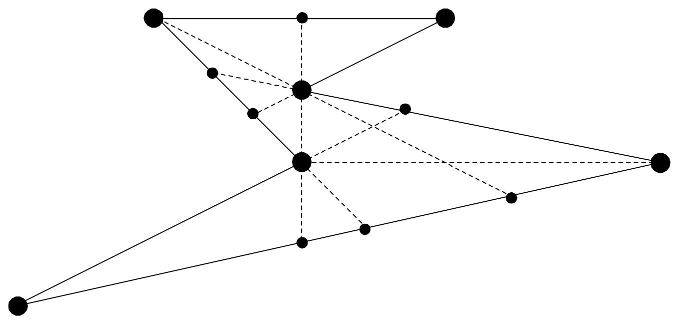

Fig. 1. Subdividing the boundary into line segments with endpoints in FVS

it holds

$$L(\cup_{i=m}^{k} S_i') \leq c_m \frac{L(S_l)}{c_l} + c_{m+1} \frac{L(S_l)}{c_l} + ... + c_{m+k} \frac{L(S_l)}{c_l} \leq B \frac{L(S_l)}{c_l}$$

Therefore

$$L(OPT) - L(\cup_{i=1}^{l-1} S_i) \leq B \frac{L(S_l)}{c_l}$$

It also holds

$$L(\cup_{i=1}^{l} S_i) - L(\cup_{i=1}^{l-1} S_i) = L(S_l)$$

Thus

$$L(\cup_{i=1}^{l} S_i) - L(\cup_{i=1}^{l-1} S_i) \geq \frac{c_l}{B}(L(OPT) - L(\cup_{i=1}^{l-1} S_i))$$

Lemma 2. *After l iterations it holds:*

$$L(\cup_{i=1}^{l} S_i) \geq (1 - \prod_{k=1}^{l}(1 - \frac{c_k}{B}))L(OPT)$$

Proof. We will prove this by induction on l. It holds

$$L(S_1) \geq \frac{c_1}{B} L(OPT)$$

Suppose it holds for $l - 1$:

$$L(\cup_{i=1}^{l-1} S_i) \geq (1 - \prod_{k=1}^{l-1}(1 - \frac{c_k}{B}))L(OPT)$$

So

$$L(\cup_{i=1}^{l} S_i) = L(\cup_{i=1}^{l-1} S_i) + (L(\cup_{i=1}^{l} S_i) - L(\cup_{i=1}^{l-1} S_i))$$

From the previous lemma:

$$L(\cup_{i=1}^{l} S_i) \geq L(\cup_{i=1}^{l-1} S_i) + \frac{c_l}{B}(L(OPT) - L(\cup_{i=1}^{l-1} S_i)) \rightarrow$$

$$L(\cup_{i=1}^{l} S_i) \geq L(\cup_{i=1}^{l-1} S_i)(1 - \frac{c_l}{B}) + \frac{c_l}{B} L(OPT)$$

From the inductive hypothesis:

$$L(\cup_{i=1}^{l} S_i) \geq (1 - \prod_{k=1}^{l-1}(1 - \frac{c_k}{B}))L(OPT)(1 - \frac{c_l}{B}) + \frac{c_l}{B} L(OPT) \rightarrow$$

$$L(\cup_{i=1}^{l} S_i) \geq (1 - \prod_{k=1}^{l}(1 - \frac{c_k}{B}))L(OPT)$$

Theorem 1 *Algorithm 1 runs in polynomial time and achieves a constant* $0,316$ *approximation ratio with respect to the optimum of the* Budgeted Maximum Length Vertex Guard *problem.*

Proof. From lemma 2 it holds:

$$L(\cup_{i=1}^{l+1} S_i) \geq (1 - \prod_{k=1}^{l+1}(1 - \frac{c_k}{B}))L(OPT)$$

$$\geq (1 - \prod_{k=1}^{l+1}(1 - \frac{c_k}{\sum_{i=1}^{l+1} c_i}))L(OPT), \text{ since } \sum_{i=1}^{l+1} c_i > B$$

The function

$$(1 - \prod_{i=1}^{n}(1 - \frac{a_i}{A}))$$

where $\sum_{i=1}^{n} a_i = A$ has a minimum when $a_1 = a_2 = ... = a_n = \frac{A}{n}$. Therefore

$$L(\cup_{i=1}^{l+1} S_i) \geq (1 - (1 - \frac{1}{l+1})^{l+1})L(OPT) \geq (1 - \frac{1}{e})L(OPT)$$

It holds $L(S_{l+1}) \leq L(S_t)$ since $L(S_t)$ is a maximum length that a vertex can cover. Thus

$$L(\cup_{i=1}^{l} S_i) + L(S_t) \geq L(\cup_{i=1}^{l} S_i) + L(S_{l+1}) \geq (1 - \frac{1}{e})L(OPT)$$

Thus one of the values $L(\cup_{i=1}^{l} S_i)$ or $L(S_t)$ is greater or equal to

$$\frac{1}{2}(1 - \frac{1}{e})L(OPT)$$

In case of edge guards, the only difference is that before running algorithm 1, we have to calculate $E'(e_i)$ for every edge. All we have to do is to examine the FVS segments which are watched by FVS points belonging to edge e_i. Then, we run algorithm 1 using $E'(e_i)$ instead of $E'(v_i)$ and of course the edge costs instead of the vertex costs. For both of the cases above, we proved in [13] that the special case of vertices or edges with cost 1, i.e. MAXIMUM LENGTH VERTEX/EDGE GUARD is APX-hard. Thus BUDGETED MAXIMUM LENGTH VERTEX/EDGE GUARD in polygons with (or without) holes is APX-complete.

3 The BUDGETED MAXIMUM VALUE VERTEX/EDGE GUARD problem

Suppose a polygon P with (or without) holes is given with weighted disjoint line segments on its boundary (including boundaries of the possible holes). Our line segments are open intervals (a, b). Every vertex of the polygon has a cost value assigned (see figure 2). Finally a budget $B > 0$ is given. The goal is to place guards on vertices (or edges) so that a maximum weight is covered and the total cost of vertices (or edges) with a guard is at most B.

Definition 2 *Given is a polygon P with or without holes and an integer $B > 0$. Assume the boundary of P is subdivided into disjoint line segments with non negative weights (see figure 2). Every vertex (or edge) has a cost value. The goal of the* BUDGETED MAXIMUM VALUE VERTEX/EDGE GUARD *problem is to place vertex (or edge) guards so that the total weight of the set of line segments watched (or overseen) is maximum and the total cost of vertices (or edges) that have guards is at most B.*

In [12] we introduced MAXIMUM VALUE VERTEX/EDGE GUARD for polygons with or without holes and proved that it is NP-hard. MAXIMUM VALUE VERTEX/EDGE GUARD is a special case of BUDGETED MAXIMUM VALUE VERTEX/EDGE GUARD where all vertices (or edges) have cost 1.

Fact 2 BUDGETED MAXIMUM VALUE VERTEX/EDGE GUARD*, for polygons with or without holes is NP-hard.*

First, we study the case of watching the segments, placing vertex guards. Algorithm 2 uses the precalculated set $E'(v_i)$, $\forall i$, which is the set of segments that are watched by v_i. This can be done easily by noticing that a segment (a, b) is watched by a vertex v if at least one of the FVS segments that have a part in (a, b) is watched by vertex v.

Theorem 2 *Algorithm 2 runs in polynomial time and achieves a 0.316 constant approximation ratio with respect to the optimum of the* BUDGETED MAXIMUM VALUE VERTEX GUARD *problem.*

The proof of theorem 2 is similar to that of theorem 1.

In case of edge guards, the only difference in algorithm 2 is that we need to calculate $E'(e_i)$, $\forall i$, which is the set of segments that are watched by e_i. We notice that a segment

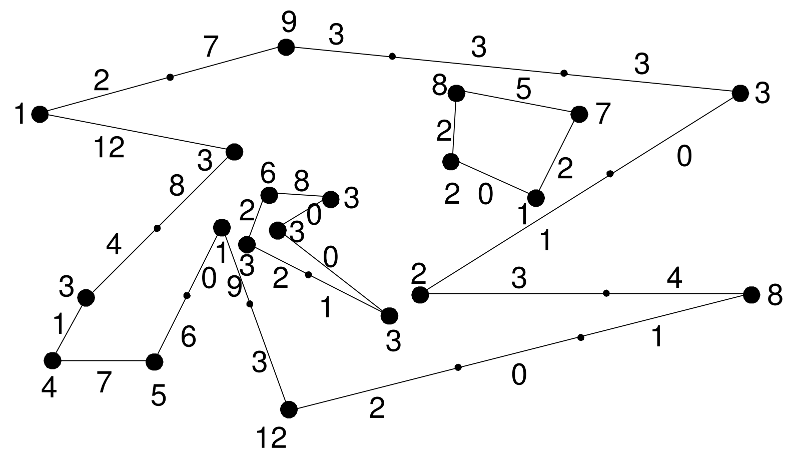

Fig. 2. A weighted polygon with weights on edge segments and costs on vertices

(a, b) is watched by an edge e if at least one of the FVS segments that have a part in (a, b) is watched by any of the endpoints of FVS segments contained in edge e.

Now we consider the case in which the goal is to oversee a maximum weight with the condition that an overseen segment must be overseen as a whole by **one** guard. We can use algorithm 2 with modified sets E': a) for the case of vertex guards we consider a segment (a, b) overseen by a vertex v, if all FVS segments of (a, b) are watched by v, b) for the case of edge-guards we consider a segment (a, b) overseen by an edge e, if all FVS segments of (a, b) are watched by some of the endpoints of FVS segments of edge e.

Thus, algorithm 2 (with proper modifications) approximates all the above cases achieving a constant ratio.

4 Conclusion

We investigated the following problems for polygons with or without holes: 1) Budgeted Maximum Length Vertex/Edge Guard, 2) watching Budgeted Maximum Value Vertex/Edge Guard, 3) overseeing Budgeted Maximum Value Vertex/Edge Guard. All problems are NP-hard and we found polynomial time approximation algorithms with constant ratio for all of them using a greedy technique based on the approximation of the Budgeted Maximum Coverage problem.

While investigating the above problems we used a) **weights** on pieces of the polygon's boundary, and b) the useful and promising concept of **watching** a set of points or line segments as opposed to completely **overseeing** it. In addition we introduced **costs** on candidate guard places (vertices and edges).

Algorithm 2 BudgetedMaxValueVertexGuards (* greedy *)

$SOL \leftarrow \emptyset$
$G \leftarrow \emptyset$
$C_{TOT} \leftarrow 0$
$M \leftarrow V$
repeat
  select $v_i \in V$ that maximizes $\frac{W(G \cup E'(v_i))}{c_i}$
  if $C_{TOT} + c_i \leq B$ **then**
    $G \leftarrow G \cup E'(v_i)$
    $C_{TOT} \leftarrow C_{TOT} + c_i$
  end if
  $M \leftarrow M - v_i$
until $M = \emptyset$
select $v_t \in V$ that maximizes $W(E'(v_t))$
if $W(G) \geq W(E'(v_t))$ **then**
  $SOL \leftarrow G$
else
  $SOL \leftarrow E'(v_t)$
end if
return $W(SOL)$

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. Shermer, T., Recent results in Art Galleries, Proc. of the IEEE, 1992.
4. Urrutia, J., Art gallery and Illumination Problems, Handbook on Computational Geometry, 1998.
5. Eidenbenz, S., (In-)Approximability of Visibility Problems on Polygons and Terrains, PhD Thesis, ETH Zurich, 2000.
6. Eidenbenz, S., Inapproximability Results for Guarding Polygons without Holes, Lecture notes in Computer Science, Vol. 1533 (ISAAC'98), p. 427-436, 1998.
7. Ghosh, S., Approximation algorithms for Art Gallery Problems, Proc. of the Canadian Information Processing Society Congress, pp. 429-434, 1987.
8. Hochbaum, D., Approximation Algorithms for NP-Hard Problems, PWS Publishing Company, 1996.
9. Laurentini A., Guarding the walls of an art gallery, The Visual Computer Journal, (1999) 15:265-278.
10. Deneen, L., Joshi, S., Treasures in an art gallery, Proc. 4th Canadian Conf. Computational Geometry, pp. 17-22, 1992.
11. Carlsson, S., Jonsson, H., Guarding a Treasury, Proc. 5th Canadian Conf. Computational Geometry, pp. 85-90, 1993.
12. Markou, E., Fragoudakis, C., Zachos, S., Approximating Visibility Problems within a constant, 3rd Workshop on Approximation and Randomization Algorithms in Communication Networks, Rome pp. 91-103, 2002.

13. Markou, E., Zachos, S., Fragoudakis, C., Maximizing the Guarded Boundary of an Art Gallery is APX-complete, 5th Italian Conference on Algorithms and Complexity, Rome, 2003 (accepted).
14. Khuller, S., Moss, A., Naor, J., The Budgeted Maximum Coverage Problem, Information Processing Letters 70(1): 39-45 (1999).