OPTIMUM PLACEMENT OF GUARDS

Extended Abstract

Simeon Ntafos [1] and **Markos Tsoukalas**

Computer Science Program
The University of Texas at Dallas
Richardson, TX 75083-0688

Abstract: We consider the problem of placing a guard(s) in a polygon so that the area (or the portion of the boundary) visible to the guard(s) is maximized. We show that finding optimum placements for k guards is NP-hard if k is a variable. We reduce the problem of optimally placing one guard to solving a high order equation and give a polynomial time approximation scheme for placing one guard in a simple polygon.

I. Introduction.

The Art Gallery problem has received considerable attention in the Computational Geometry literature. The problem is to find the minimum number of guards and a placement for them so that every point in the polygon is visible to at least one guard. It is known that finding a minimum set of guards is NP-Hard [GJ79,Ag84,LL86]. This and many other results on the Art Gallery and related problems can be found in [OR87].

In this paper we consider the related problem of finding an optimum placement for a specified number of guards. We consider two natural optimization criteria: (a) placing the guards so that the area inside the polygon that is visible to at least one guard is maximized and (b) placing the guards so that the portion of the boundary of the polygon that is visible to at least one guard is maximized. Since the area visible to a guard is a star-shaped polygon, the guard placement problems are equivalent to maximizing the area/perimeter covered by an equal number of star-shaped polygons. We will refer to guards seeing or covering portions of the polygon interchangeably.

In the next section we show that the optimum placement problems for k guards remain NP-Hard if k is a variable. In section III we consider the problem of optimally placing a single guard inside a polygon. We distinguish vertex, perimeter and point guards to refer to guards that are restricted to lie at a vertex, edge or any point in the polygon respectively. We give a polynomial time algorithm for optimally placing one vertex guard. We reduce the problem of optimally placing one edge or point guard to solving a high order equation and present polynomial approximation schemes for these problems.

II. Placing k Guards.

Let P be a simple polygon with n vertices (edges), let k be an integer, $1 \leq k \leq n$ and let M be a positive real number. The k-Guard Placement problem is stated as follows:

k-Guard Placement: Is there a placement of k guards in P such that the total area inside P that is visible to at least one guard is at least M?

[1] Supported in part by NSF Grant IRI-9000470 and a grant from Texas Instruments, Inc.

By making M equal to the area of the polygon, the k-guard placement problem becomes a simple variation of the Art Gallery problem. If there were a polynomial time algorithm for the k-guard placement problem, we could use it repeatedly to find the smallest k such that all of the polygon is visible to a guard. This would then provide a polynomial time solution to the Art Gallery problem which is known to be NP-Hard. Thus, the k-guard placement problem is NP-Hard. Even if k is known to be less than the number of guards needed to cover the whole polygon (but remains a variable), the k-guard placement problem remains NP-Hard. The reduction again uses the Art Gallery problem but modifies P by adding a very small area that contains an appropriate number of bends so that a certain number of guards is always required to see it. Similar modifications can be used to show that the k-guard placement problem remains NP-Hard if M is known to be less than the area of P. Similar results can be obtained if we require that the k guards see at least M units of length along the boundary of P. These intractability results hold for vertex, perimeter and point guards.

Theorem 1: Placing k (variable) guards in P so as to maximize the area or portion of the boundary visible to them is NP-hard.

III. Placing a Single Guard.

Although the k-guard placement problem is NP-hard for variable k, it may be possible to develop efficient algorithms for fixed values of k. We consider the problem of optimally placing a single guard in a polygon P. We distinguish three progressively more powerful guard types: vertex, perimeter and point guards. Vertex guards are restricted to vertices of P, perimeter guards must be placed at points on the boundary of P and point guards may be placed anywhere on the boundary or the interior of P. Clearly, the best point guard covers at least as much area (boundary) as the best perimeter guard which in turn sees at least as much as the best vertex guard.

Finding the optimum placement for one vertex guard is simple. Since the number of possibilities is limited, we can try all of them and select the best one. To determine the coverage from each vertex we compute the visibility polygon from that vertex and compute its area and the length of the boundary of P that also appears in the boundary of the visibility polygon. All this can be done in $O(n)$ per vertex using existing algorithms for computing the visibility polygon from a point [EA81] and using the recent linear triangulation algorithm in [Ch90] to compute areas from a triangulation.

Theorem 2: The optimum placement for one vertex guard in a simple polygon P can be found in $O(n^2)$.

Consider now perimeter guards. Here the number of possible locations for the guard is not bounded. Our approach is based on the observation that the area (perimeter) visible to a point along an edge does not usually change significantly if the point is moved slightly along the edge. The visibility polygon changes abruptly only at certain points along an edge and those points correspond to "turning-the-corner" situations. Consider the subdivision of edges of P induced by taking the straight line segments connecting each reflex vertex in P to all other vertices visible to it and extending them until they intersect the boundary of P. The resulting edge segments have the property that the visibility polygons for all points along each one of them are closely related. In general, a visibility polygon is defined by vertices of P and by Steiner points where the shadow of a reflex vertex of P contacts the boundary. The property that all the visibility polygons for points along an induced segment share is that they all contain the same subset of vertices of P. Furthermore, as a guard moves along the segment, the Steiner points in the corresponding visibility polygons sweep portions of edges of P without ever going through a vertex of P.

Lemma 1: The number of induced segments along edges of P is $O(n^2)$.

Consider an induced edge segment (a,b) and let x be a point in it. There are areas of P that are visible to x regardless of where x lies along (a,b). Also, there are triangular areas with a reflex vertex of P at one apex that are completely visible from one of a, b and completely invisible to the other. As x moves from a to b along the segment (a,b), the triangular areas that are visible to a only become progressively more and more blocked from view while the triangular areas that are visible to b only become more and more visible. We refer to these triangular areas as **variable triangles** and denote them positive or negative depending on whether the triangle becomes more or less visible as x moves from a to b respectively (see Figure 1). Since a reflex vertex of P is always at one apex of the triangular areas (the pivot), we have:

Lemma 2: The number of variable triangles for an induced segment is $O(n)$.

To find the point x along an induced segment that covers the most area in P or sees the largest portion of the boundary of P, we need consider only the variable triangles. If our goal is to maximize the length of the boundary of P that is visible to x, we need to maximize a summation of the form:

$$\sum_{i=1}^{m} (A_i\, y) / (B_i\, y + C_i)$$

where m is the number of triangular areas involved, y is the distance of x from a and A_i, B_i, C_i are quantities that depend on the length of the induced segment (a, b), the position of the pivot reflex vertex and the orientation and length of the side of each triangular area that is not adjacent to the pivot. Minimizing such a summation reduces to solving a high order equation that does not appear to have a closed form solution. Thus we need to resort to numerical methods which will find an approximation (to any desired accuracy) for the best location of the guard along the segment (a, b). To find the best placement overall, we repeat this for each induced segment and compare the resulting candidates. If the goal is to optimize the area visible to the guard, we use exactly the same approach but the summation is slightly more complex as it involves square roots.

Theorem 3: The optimum placement of a single perimeter guard can be found in $O(n^2n+L)$, where n is the size of P and L is the time it takes to solve an equation of degree $O(n)$ numerically.

While numerical solutions will probably work very well in practice, it is more interesting to develop approximation algorithms whose complexity and quality of solutions can be related to problem parameters. To this end we can partition each segment by placing m equally spaced points along it, find the visibility polygon for each such point and select one that maximizes the area/perimeter seen. However, this does not always result in good approximate solutions as the example in Figure 2 shows. We obtain a polynomial approximation scheme that obtains solutions within a factor of $(1/2m)$ of the optimum by subdividing each of the bases of the triangular areas into m parts. This induces a partition of size $O(m^2)$ on the segment (a, b). The computation of the areas or perimeters can be done in $O(m^2n)$ per induced segment (we compute one visibility polygon, find its area/perimeter and then adjust it in constant time per point to obtain the remaining areas/perimeters).

Theorem 4: There is an approximation scheme for placing a single perimeter guard that runs in $O(m^2n^3)$ and obtains solutions within $1/2m$ of the optimum.

Consider now the placement of a point guard inside P. The line segments that we used to partition the edges into segments also induce a partition of the interior of P into regions. There are $O(n^2)$ lines so there can be at most $O(n^4)$ induced regions. All points within the same region

see the same set of vertices of P. Again we have $O(n)$ variable triangles which are defined by a reflex vertex (pivot) and supporting lines to the region. Finding the optimum guard placement for each region involves optimizing a summation over an area. We can obtain a polynomial time approximation scheme by subdividing the bases of the variable triangles into m parts and considering all intersection points in the induced subdivision of each region. The complexity however is rather high.

References.

[Ag84] Aggarwal, A., "The Art Gallery Theorem and Algorithms" Ph.D. Thesis, John Hopkins University, 1984.

[Ch90] Chazelle, B., "Triangulating a Simple Polygon in Linear Time", Proc. of 31st FOCS, pp. 220-230, 1990.

[EA81] El Gindy, H. and D. Avis, "A Linear Algorithm for Computing the Visibility Polygon from a Point", Journal of Algorithms, Vol. 2, No. 2, pp. 186-197, 1981.

[LL86] Lee, D.T. and Lin, A.K., "Computational Complexity of Art Gallery Problems", IEEE Transactions on Information Theory, Vol. 32, pp. 276-282, 1986.

[GJ79] Garey, M. and D. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, Freeman, 1979.

[OR87] O'Rourke, J., Art Gallery Theorems and Algorithms, Oxford, 1987.

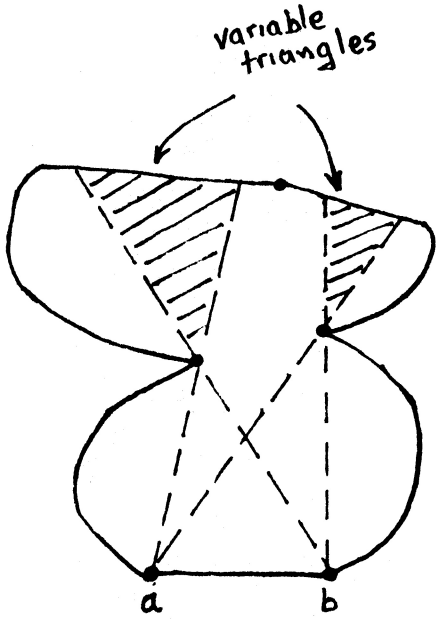

Figure 1

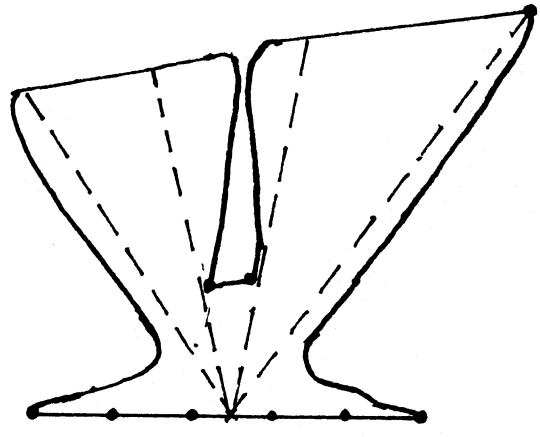

Figure 2