Approximate Guarding of Monotone and Rectilinear Polygons

Bengt J. Nilsson

Technology and Society, Malmö University College
SE-205 06 Malmö, Sweden.
email: Bengt.Nilsson@ts.mah.se

Abstract. We show a constant factor approximation algorithm for interior guarding of monotone polygons. Using this algorithm we obtain an approximation algorithm for interior guarding rectilinear polygons that has an approximation factor independent of the number of vertices of the polygon. If the size of the smallest interior guard cover is OPT for a rectilinear polygon, our algorithm produces a guard set of size $O(OPT^2)$.

1 Introduction

The art gallery problem is perhaps the best known problem in computational geometry. It asks for the minimum number of guards to guard a space having obstacles. Originally, the obstacles were considered to be walls mutually connected to form a closed Jordan curve, hence, a simple polygon. Tight bounds for the number of guards necessary and sufficient were found by Chvátal [7] and Fisk [14]. Subsequently, other obstacle spaces, both more general and more restricted than simple polygons have also been considered for guarding problems, most notably, polygons with holes and simple rectilinear polygons [16, 22].

Art gallery problems are motivated by applications such as line-of-sight transmission networks in polyhedral terrains, e.g., signal communications and broadcasting, cellular telephony, and other telecommunication technologies as well as placement of motion detectors and security cameras.

We distinguish between two types of guarding problems. Vertex guarding considers only guards positioned at vertices of the polygon, whereas interior guarding allows the guards to be placed anywhere in the interior of the polygon.

The computational complexity question of guarding simple polygons was settled by Aggarwal [1] and Lee and Lin [19] independently when they showed that the problem is NP-hard for both vertex guards and interior guards. Further results have shown that already for very restricted subclasses of polygons the problem is still NP-hard [3, 21]. Also, Chen *et al.* [5] claim that vertex guarding a monotone polygon is NP-hard, however the details of their proof are omitted and still to be verified.

The approximation complexity of guarding polygons has been studied by Eidenbenz and others. Eidenbenz [13] shows that polygons with holes cannot be efficiently guarded by fewer than $\Omega(\log n)$ times the optimal number of interior

or vertex guards, unless P=NP, where n is the number of vertices of the polygon. Brodén *et al.* and Eidenbenz [3, 12] independently prove that interior guarding simple polygons is APX-hard, thus showing that, unless P=NP, no PTAS is possible for this problem.

Any polygon (with or without holes) can be efficiently vertex guarded with logarithmic approximation factor in the number of vertices of the polygon. The algorithm is a simple reduction to SET COVER and goes as follows [15]: compute the arrangement produced by the visibility polygons of the vertices. Next, let each vertex v correspond to a set in the set cover instance consisting of elements corresponding to the faces of the arrangement that lie in the visibility polygon of v. The greedy algorithm for SET COVER will then produce a guard cover having logarithmic approximation factor.

The above result can be improved for simple polygons using randomization, giving an algorithm with expected running time $O(nOPT^2 \log^4 n)$ that produces a vertex guard cover with approximation factor $O(\log OPT)$ with high probability, where n is the size of the polygon and OPT is the smallest vertex guard cover for the polygon [11].

Recently, interesting constant factor approximation algorithms have been developed for the special case of one-dimensional terrains [2, 8]. A terrain is a two-dimensional region above a monotone chain.

We prove polynomial time approximation algorithms for interior guarding of monotone and rectilinear polygons. As mentioned, vertex guarding of monotone polygons is believed to be NP-hard, and furthermore, it is known that covering rectilinear polygons with the minimum number of convex pieces (rectangles) is NP-hard [9]. This suggests that interior guarding these two classes of polygons is difficult and provides the basis for our interest in approximation algorithms for these problems.

The next section contains some useful definitions and an Sections 3 and 4 we describe the algorithms for monotone and rectilinear polygons respectively.

2   Definitions

A polygon $\mathbf{P}$ is *l-monotone* if there is a line of monotonicity l such that any line orthogonal to l has a simply connected intersection with $\mathbf{P}$. When we talk about monotone polygons, we will henceforth assume that they are x-monotone, i.e., the x-axis is the line of monotonicity for the polygons we consider.

The boundary of a monotone polygon $\mathbf{P}$ can be subdivided into two chains, the *upper chain* U and the *lower chain* D. Let s and t be the leftmost and rightmost vertices of $\mathbf{P}$ respectively. The chain U consists of the boundary path followed from s to t in clockwise direction, whereas D is the boundary path followed from s to t in counterclockwise direction.

A polygon $\mathbf{P}$ is *rectilinear* if the boundary of $\mathbf{P}$ consists of axis parallel line segments. Hence, at all vertices, the interior angle between segments are either 90 or 270 degrees; see Figure 1(a).
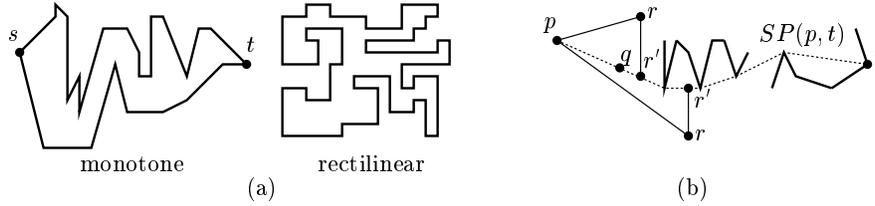
Fig. 1. Illustrating definitions and Lemma 1.

Let $\mathbf{VP}(p)$ denote the visibility polygon of $\mathbf{P}$ from the point p, i.e, the set of points in $\mathbf{P}$ that can be connected with a line segment to p without intersecting the outside of $\mathbf{P}$.

Consider a partial set of guard points $g_1, \ldots, g_m$ in $\mathbf{P}$ and the union of their visibility polygons $\bigcup_{i=1}^{m} \mathbf{VP}(g_i)$, the set $\mathbf{P} \setminus \bigcup_{i=1}^{m} \mathbf{VP}(g_i)$ is the region of $\mathbf{P}$ not seen by the points $g_1, \ldots, g_m$. This region consists of a set of simply connected polygonal regions called *pockets* bounded by either the polygon boundary or the edges of the visibility polygons.

The following definitions are useful for monotone polygons. Let q be a point in $\mathbf{VP}(p)$ that lies to the right of p. We denote by $\mathbf{VP}_R(p, q)$ the part of $\mathbf{VP}(p)$ that lies to the right of q. Also, $\mathbf{VP}_R(p) = \mathbf{VP}_R(p, p)$.

A pocket in a monotone polygon $\mathbf{P}$ is an *upper pocket* if it is adjacent to the upper boundary U of $\mathbf{P}$, otherwise it is a *lower pocket*. Note that an upper pocket can be adjacent to D whereas a lower pocket is never adjacent to U.

Let $SP(p, q)$ denote the shortest (Euclidean) path between points p and q inside $\mathbf{P}$.

Lemma 1. *If q is a point on $SP(p, t)$ inside a monotone polygon $\mathbf{P}$, then* $\mathbf{VP}_R(p, q) \subseteq \mathbf{VP}_R(q)$.

Proof. Let r be a point to the right of q in $\mathbf{P}$ that is visible from p. To prove that r is seen from q consider the vertical line through r and its intersection point r' with $SP(p, t)$. The three points p, r, and r' define a polygon in $\mathbf{P}$ having three convex vertices and possibly some reflex vertices on the path $SP(p, r')$. Since r sees both p and r', r sees all of the path $SP(p, r')$ and hence also the point q; see Figure 1(b). □

3 Interior Guarding Monotone Polygons

Our algorithm for guarding a monotone polygon $\mathbf{P}$ will incrementally guard $\mathbf{P}$ starting from the left and moving right. Hence, we are interested in the structure of the pockets that occur when guarding is done in this way. We first define *kernel expansions* of the pockets given a partial guard cover $\mathcal{G}_p$, and then taking maximal nonempty intersection of these we produce the main region that we will be interested in. This region is called a *spear* and with this we can define a well behaved guard cover $\mathcal{G}^*$ that has small size. We finally prove that our incremental algorithm produces a guard cover at most a constant times larger than $\mathcal{G}^*$.

Assume that we have a partial guard cover $\mathcal{G}_p$ in $\mathbf{P}$ and that everything to the left of the rightmost guard is seen. Consider the upper pockets resulting from this guard cover and enumerate them $\mathbf{p}_1^U, \ldots$ from left to right. The lower pockets are enumerated from left to right $\mathbf{p}_1^D, \ldots$ in the same way; see Figure 2.
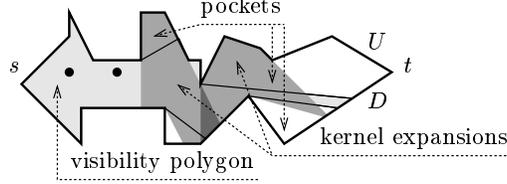
Fig. 2. Illustrating pockets and kernel expansions.

Let $\mathbf{p}^U$ be an upper pocket. The *kernel expansion* $\mathbf{ke}(\mathbf{p}^U)$ consists of all the points in $\mathbf{P}$ that see everything in $\mathbf{p}^U$ to the left of themselves. For the lower pockets we define the kernel expansion symmetrically. The definition of kernel expansion is valid also when no guards have as yet been placed in the polygon. In this case, we take all of the polygon $\mathbf{P}$ to be an upper pocket.

Let k be the largest index so that $\bigcap_{i=1}^{k} \mathbf{ke}(\mathbf{p}_i^U)$ is nonempty. This nonempty intersection of kernel expansions is called the *upper spear*, also denoted $\mathbf{sp}^U$. We can in the same way define the *lower spear* $\mathbf{sp}^D$ as the maximal nonempty intersection of the kernel expansions for the lower pockets.

Given the partial guard cover $\mathcal{G}_p$ the upper spear $\mathbf{sp}^U$ can be computed in linear time as follows. Let r_1^U be the leftmost point of $\mathbf{p}_1^U$. An edge e of the pocket that is also part of U is defined to have the same direction as when it is traversed during a traversal of U from s to t. Following the boundary of the upper pockets starting at r_1^U, for each edge e of the pocket that is also an edge of U, we issue a half line from e having the same direction as e. When the traversal of a pocket $\mathbf{p}_i^U$ reaches the last edge we establish the last point q_i^U of the edge not seen by $\mathcal{G}_p$ and we extend the directed half line issuing from q_i^U toward the vertex v on U between q_i^U and r_{i+1}^U so that this half line has minimum interior angle and does not intersect the exterior of $\mathbf{P}$ above U. Using these half lines in the order they were computed we incrementally find their right half plane intersection in the same way as is done to compute the kernel of a polygon [17, 18]. This gives us the upper boundary of the spear.

To compute the lower boundary of the spear we follow the lower boundary D of $\mathbf{P}$ from the point having the same x-coordinate as r_1^U toward t and maintain the half lines issuing from r_1^U having maximal interior angle to U and such that they do not intersect the exterior of $\mathbf{P}$ below D; see Figure 3. The intersection point between the upper and lower boundary of the spear is called the upper *spear tip* and we denote it u^U.

In a similar manner we can compute the lower spear $\mathbf{sp}^D$ and its lower spear tip u^D.

To every spear $\mathbf{sp}$ we also associate a region called the *shadow* of the spear, denoted $\mathbf{shd}(\mathbf{sp})$. If the spear tip lies on the lower boundary D the shadow is
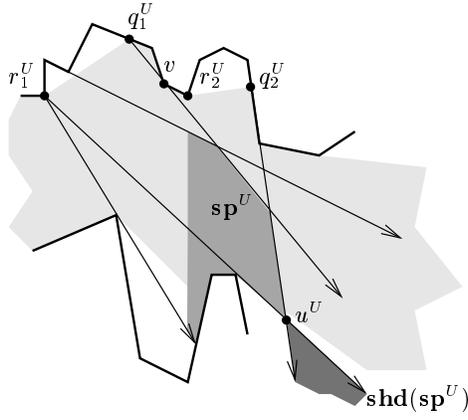
Fig. 3. Computing the upper spear and the shadow.

the empty set. If the spear tip lies in the interior of $\mathbf{P}$, the shadow of the spear is the region to the right of the spear tip between the two half lines bounding the spear that intersect at the spear tip; see Figure 3.

The upper and lower spears are dependent on the placement of the previously placed guards so we will henceforth refer to them as $\mathbf{sp}^U(\mathcal{G}_p)$ and $\mathbf{sp}^D(\mathcal{G}_p)$ given the partial guard set $\mathcal{G}_p$. For each spear, $\mathbf{sp}^U(\mathcal{G}_p)$ and $\mathbf{sp}^D(\mathcal{G}_p)$ we denote the upper spear tip $u^U(\mathcal{G}_p)$ and the lower spear tip $u^D(\mathcal{G}_p)$. If $\mathcal{G}_p = \emptyset$, the upper spear $\mathbf{sp}^U(\emptyset)$ and the upper spear tip $u^U(\emptyset)$ are well defined.

We prove the following two lemmas.

Lemma 2. *If $\mathcal{G}_p$ and $\mathcal{G}_p'$ are two partial guard covers of $\mathbf{P}$ such that $\mathbf{sp}^U(\mathcal{G}_p)$ and $\mathbf{sp}^U(\mathcal{G}_p')$ do not intersect, then $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_p)) \cap \mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_p')) = \emptyset$.*

Proof. We make a proof by contradiction and assume that the two shadows intersect. Assume that $\mathbf{sp}^U(\mathcal{G}_p)$ lies to the left of $\mathbf{sp}^U(\mathcal{G}_p')$ and let p be a point in the intersection of the two shadows. We can connect p to $u^U(\mathcal{G}_p)$ with a line segment and then follow the line segment from $u^U(\mathcal{G}_p)$ back to its starting point r_1^U at the leftmost point of the first upper pocket associated to $\mathbf{sp}^U(\mathcal{G}_p)$. From r_1^U we follow the upper boundary of the pocket to the rightmost point q_k^U of the last upper pocket associated to $\mathbf{sp}^U(\mathcal{G}_p')$, from this point on to $u^U(\mathcal{G}_p')$, and then back to p. This traversal bounds a polygon interior to $\mathbf{P}$ that contains completely the lower boundary segment of $\mathbf{sp}^U(\mathcal{G}_p')$. However, this is not possible because by construction this segment must intersect the lower boundary D of $\mathbf{P}$, giving us a contradiction; see Figure 4. $\square$
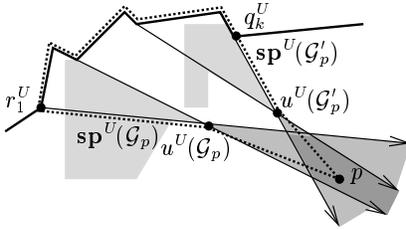
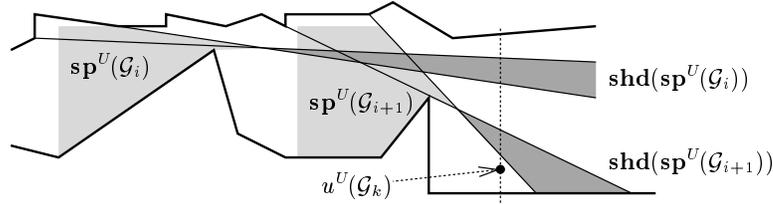**Fig. 4.** Illustrating the proof of Lemma 2.

At this point, it is important to note that a single guard placed in a spear will guard all pockets associated to the spear. However, these pockets can also be guarded by placing one guard above and below the shadow and possibly one or more guards inside the shadow. Now, we are interested in bounding from below the number of guards needed to the right of a spear for the case that no guard is in the spear.

Let $\mathcal{G}$ be any guard cover for $\mathbf{P}$ and let $\mathcal{G}_p \subset \mathcal{G}$ be a possibly empty partial guard cover. If $\mathcal{G}_p$ is nonempty assume that it has g as its rightmost guard and assume further that all of $\mathbf{P}$ to the left of g is guarded by $\mathcal{G}_p$. Let $\mathcal{G}_f = \mathcal{G} \setminus \mathcal{G}_p$ and define the following sets recursively.

$$\mathcal{G}_0 = \mathcal{G}_p$$
$$\mathcal{G}_i = \mathcal{G}_{i-1} \cup \{u^U(\mathcal{G}_{i-1})\} \qquad \text{for } i > 0$$

Lemma 3. *Let $\mathcal{G}_f$ and $\mathcal{G}_0, \ldots, \mathcal{G}_k$ be sets as defined above. If all guards of $\mathcal{G}_f$ lie to the right of $u^U(\mathcal{G}_k)$, then $\mathcal{G}_f$ contains at least $k + 2$ guards.*

Proof. By the construction of the sets $\mathcal{G}_i$, we know that their corresponding spears do not intersect, and hence, from Lemma 4 their shadows do not intersect either. Let g be the rightmost guard of $\mathcal{G}_p = \mathcal{G}_0$.

Fig. 5. Illustrating the proof of Lemma 3.

If some shadow $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_i))$ does not intersect the vertical line through $u^U(\mathcal{G}_k)$ then at least one guard is needed in the interval between g and $u^U(\mathcal{G}_k)$, contradicting that $\mathcal{G}$ is a guard cover for $\mathbf{P}$. Hence, all shadows intersect this vertical line. Now, assume that $\mathcal{G}_f$ contains at most $k+1$ guards. By the pigeon-hole principle there are two consecutive shadows $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_i))$ and $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_{i+1}))$ such that at least one of them and the region between them does not contain any guard, i.e., all guards of $\mathcal{G}_f$ lie either above the upper boundary of $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_i))$ and below the upper boundary of $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_{i+1}))$ or above the lower boundary of $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_i))$ and below the lower boundary of $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_{i+1}))$. In both cases this means that there are points of the last pocket associated to $\mathbf{sp}^U(\mathcal{G}_i)$ that are not seen by $\mathcal{G}$, giving us a contradiction; see Figure 5. □

We say that a guard *belongs* to a shadow $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_i))$, if it lies in the interior of $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_i))$ or it lies below $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_i))$ but above all other shadows lying below $\mathbf{shd}(\mathbf{sp}^U(\mathcal{G}_i))$.

We can, of course, prove similar results as those in Lemmas 2 and 3 for the shadows of lower spears.

A guard cover is called *serial*, if the following invariant condition holds when we incrementally place the guards of the cover in **P** one by one in order from left to right.

> If g_m is the mth guard placed in the order, then g_m either lies in the upper spear or the lower spear of the guards $g_1, \ldots, g_{m-1}$.

The next lemma shows that there is a serial guard cover of small size.

Lemma 4. *If $\mathcal{G}$ is a guard cover for the monotone polygon **P**, then there is a serial guard cover $\mathcal{G}^*$ for **P** such that $|\mathcal{G}^*| \leq 3|\mathcal{G}|$.*

Proof. Given a guard cover $\mathcal{G}$ we transform it to be serial as follows. Order the guards of $\mathcal{G} = \{g_1, \ldots, g_m\}$ from left to right. The transformation incrementally adds guards moving from left to right into two sets $\mathcal{G}^U$ and $\mathcal{G}^D$ ensuring that the next guard place lies in a spear. To make the constructed guard set serial we employ a plane sweep approach moving from left to right. As soon as the sweep line reaches a guard, the guard is attached to the sweep line and moves along it following the shortest path to t. By Lemma 1 this does not decrease visibility to the right. Now, as the sweep proceeds one of two things happen. Either a guard becomes the last guard to leave a spear (with respect to the previously released guards) and it is then released from the sweep line or the sweep reaches the spear tip without having released a guard. In this case, the spear has been completely empty of guards. In this latter case, we add a guard at the spear tip, placing it in $\mathcal{G}^U$ if the spear is an upper spear and in $\mathcal{G}^D$ if the spear is a lower spear. When the sweep line reaches t, those guards still attached to it are removed (except for possibly one) giving us the serial guard cover $\mathcal{G}^*$.

To count the number of extra guards placed by this process we can associate each new guard placed at a spear tip with one belonging to the shadow of the associated spear. From Lemma 3 we have that a guard belonging to an upper (lower) shadow can at worst belong also to a lower (upper) shadow. Hence, $|\mathcal{G}^U| \leq |\mathcal{G}|$ and $|\mathcal{G}^D| \leq |\mathcal{G}|$ giving us that $|\mathcal{G}^*| \leq |\mathcal{G}| + |\mathcal{G}^U| + |\mathcal{G}^D| \leq 3|\mathcal{G}|$. □

We can now give the details of the incremental algorithm, displayed in Figure 6, and prove its correctness, approximation factor and time complexity.

To prove the complexity of the algorithm we note that the loops of Steps 2 and 3 are performed $O(n)$ times. Computing the spear and its spear tip can be done in linear time as we showed before. Hence, it remains to show how to do Step 2.4 efficiently. Let $\mathbf{VP}(\mathcal{G})$ be the part of the polygon seen so far. We begin by placing g' at the top of the line segment l and compute the upper spear with g' in the guard set. Then, we slide g' along l continuously updating the point $u^U(\mathcal{G} \cup \{g'\})$ as we go along. The structural changes of $\mathbf{sp}^U(\mathcal{G} \cup \{g'\})$ occur at certain key points on l. These are

1. when the convex vertex of $\mathbf{VP}(\mathcal{G}) \cup \mathbf{VP}(g')$ on an edge adjacent to an upper pocket becomes incident to a vertex of the polygon boundary U.
2. when an edge of the boundary of $\mathbf{sp}^U(\mathcal{G} \cup \{g'\})$ becomes incident to two vertices of the upper boundary U.

```
┌─────────────────────────────────────────────────────────────────────┐
│  Algorithm    Guard-Monotone-Polygons                               │
│  Input:    A monotone polygon P                                     │
│                                                                     │
│  Output: A guard cover for P                                        │
│                                                                     │
│  1  Let G := ∅                                                      │
│  2  while  not all upper pockets are guarded  do                   │
│  2.1        Compute sp^U(G) and u^U(G)                             │
│  2.2        Place a guard g at u^U(G); G := G ∪ {g}               │
│  2.3        Compute ⋃_{g∈G} VP(g), let p^U be the first upper      │
│             pocket in P, and let l                                  │
│             be the vertical line segment through the leftmost      │
│             boundary point of p^U                                  │
│  2.4        Place a guard g' on l so that u^U(G ∪ {g'}) lies as    │
│             far to the right as                                     │
│             possible; G := G ∪ {g'}                               │
│       endwhile                                                      │
│  3  Repeat Step 2 for the lower pockets to guard these             │
│       return  G                                                     │
│  End   Guard-Monotone-Polygons                                     │
└─────────────────────────────────────────────────────────────────────┘
```
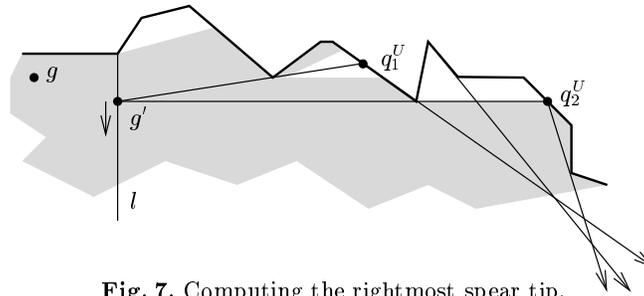
Fig. 6. The algorithm for monotone polygons.

3. when three consecutive half lines issuing from pockets intersect at the same point.

The key points occur at an at most cubic number of discrete points on l. (The maximum number of possible common intersection points between three lines among n lines.) Moving g' in between the key points will make $u^U(\mathcal{G} \cup \{g'\})$ move monotonically to the right or to the left. Hence, by computing the key points, which can be done incrementally in at most linear time, we can find the point on l where $u^U(\mathcal{G} \cup \{g'\})$ lies as far to the right as possible; see Figure 7.

Fig. 7. Computing the rightmost spear tip.

We have the following theorem.

Theorem 1. *The algorithm Guard-Monotone-Polygons computes a guard cover for a monotone polygon* **P** *of size at most* $12\,OPT$ *in polynomial time, where* OPT *is the size of the smallest guard cover for* **P**.

Proof. To prove correctness, the algorithm incrementally guards everything to the left of the rightmost guard, hence, it will completely guard the polygon.

To prove the approximation factor, consider any serial guard cover $\mathcal{G}^* = \{g_1^*, \ldots, g_m^*\}$ ordered from left to right in the polygon. We prove by induction that after the ith iteration of the loop at Step 2, the last guard placed thus far lies further to the right than g_i^*. For the base case, after the first iteration of the loopi, note that the algorithm places a guard at the first spear tip, the rightmost point of the first spear, and hence, to the right of g_1^*. Assume now that after the ith iteration of the loop, the guard g lies to the right of g_i^*. Since we place g' so that the rightmost point $u^U(\mathcal{G} \cup \{g'\})$ of $\mathbf{sp}^U(\mathcal{G} \cup \{g'\})$ lies as far to the right as possible it has to lie at least as far to the right as g_{i+1}^*. Associating g and g' during the ith iteration to g_i^* we see that the loop will place at most $2|\mathcal{G}^*|$ guards in the polygon. Thus, a total of at most $4|\mathcal{G}^*|$ guards are placed. By Lemma 4 we can choose $\mathcal{G}^*$ as the smallest serial guard cover which in turn is bounded by $3\,OPT$.

The complexity of the algorithm follows from the previous discussion. □

4 Interior Guarding Rectilinear Polygons

The algorithm for computing a guard cover in a simple rectilinear polygon consists of two main steps. First, we find a subdivision of the polygon into monotone pieces, second, we use the previously given algorithm to compute a guard cover in each monotone piece.

Consider a simple rectilinear polygon $\mathbf{P}$. To every reflex vertex v we can associate two *extensions*, i.e., the two maximal line segments in $\mathbf{P}$ through v and collinear to the two edges adjacent to v. We associate a direction to an extension e collinear to an edge e_v by giving e the same direction as e_v gets when $\mathbf{P}$ is traversed in counterclockwise order. This allows us to refer to the regions to the left and right of an extension, meaning to the left or right of e if e is directed upward.

Given two extensions e and e', we say that e *dominates* e', if all points in $\mathbf{P}$ to the left of e are also to the left of e'. Using the algorithm of Chin and Ntafos [6] in conjunction with Chazelle's triangulation algorithm [4], we can in linear time compute the most dominant extensions that we call the *essential extensions*. Assume that this computation gives us k essential extensions. An essential extension e_i is collinear to an edge with one reflex and one convex vertex. Let v_i denote the convex vertex. This gives us k convex vertices and we choose one of them, say v_k, as root in a shortest rectilinear path tree T_R to each of the other vertices v_i, for $1 \leq i < k$, that can be computed in linear time [10]. The shortest rectilinear path tree consists of paths that are shortest in the L_1-metric connecting v_k to all the other vertices v_i, for $1 \leq i < k$.

To each rectilinear path $SP_R(v_i, v_k)$ connecting v_i with v_k we define a vertical and a horizontal *histogram expansion*. The horizontal histogram expansion $\mathbf{H}_i^H$ consists of those points in $\mathbf{P}$ that can be connected to the path $SP_R(v_i, v_k)$ with vertical line segments contained in $\mathbf{P}$. We define the vertical histogram expansion $\mathbf{H}_i^V$ in a similar manner. A histogram expansion can be computed in linear time using an algorithm by Levcopoulos [20]. Each horizontal histogram expansion

consists of a number of x-monotone polygons with the property that no guard in one monotone polygon can see anything in any of the others. Similarly a vertical histogram expansion subdivides into y-monotone pieces with the same property; see Figure 8.
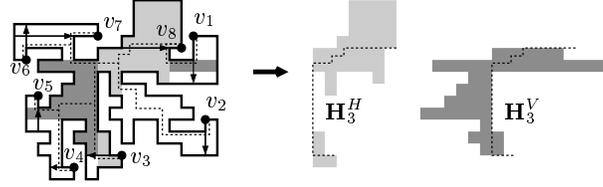

Fig. 8. Illustrating the algorithm.

Lemma 5. *If* **P** *can be guarded with OPT guards, then a histogram expansion can also be guarded with at most OPT guards interior to the region.*

Proof. Let p be a point that sees into a monotone piece **R** of a histogram expansion. Assume that **R** is x-monotone and that p lies in a region adjacent to the lower boundary D of **R**. Let l be the line segment that separates **R** from the piece containing p. Consider the intersection $\mathbf{VP}(p) \cap \mathbf{R}$. The intersection subdivides **R** into left pockets and right pockets. Traversing the boundary of $\mathbf{VP}(p)$ clockwise starting at a point outside **R** will first reach the edges that are incident to left pockets, then a boundary chain C of **R**, and finally the edges that are incident to right pockets of **R**. Take any point q of C and let p' be the intersection of the line segment between p and q with l. Any point in **R** seen by p will also be seen by p', which proves the lemma. □
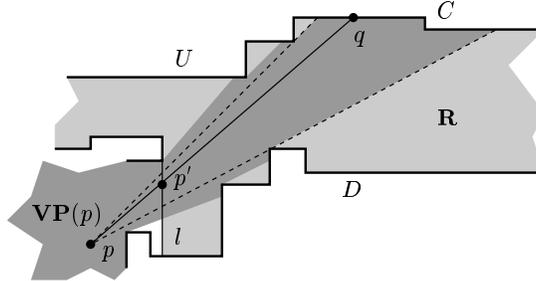

Fig. 9. Illustrating the proof of Lemma 5.

We use the *Guard-Monotone-Polygon* algorithm of the previous section to guard each monotone piece with at most $12m$ guards, where m is the smallest guard cover for the monotone piece. From Lemma 5 we know that each histogram expansion can be guarded with OPT guards interior to the histogram expansion, and hence, our algorithm guards it with at most $12\,OPT$ guards.

Furthermore, one guard can see at most two of the vertices v_i, for $1 \leq i \leq k$, hence, $k/2 \leq OPT$. Since we construct a total of $2k$ horizontal and vertical

histogram expansions, the union $\bigcup_{i=1}^{k-1} \mathbf{H}_i^H \cup \mathbf{H}_i^V$ can be guarded by at most $48OPT^2$ guards.

The set $\mathbf{P} \setminus (\bigcup_{i=1}^{k-1} \mathbf{H}_i^H \cup \mathbf{H}_i^V)$ partitions into a number of connected regions. These *extra pieces*; see Figure 10; are monotone with respect to both the x- and the y-axis and we can guard each of them with one extra guard. An extra piece is also adjacent to one horizontal and one vertical histogram expansion and each monotone piece in a histogram expansion can be adjacent to at most two extra pieces. Hence, to count the number of extra pieces, i.e., the number of additional guards we place, we associate each extra piece with the horizontal or vertical monotone piece of a histogram expansion that is closer to v_k. Thus, the number of guards placed to see all of $\mathbf{P}$ has at most doubled.
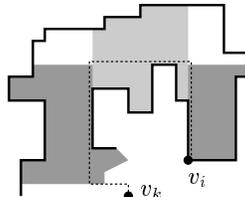

Fig. 10. Handling the extra pieces.

We have proved the following theorem.

Theorem 2. *There is a polynomial time algorithm that computes a guard cover of size $96OPT^2$ in a rectilinear polygon $\mathbf{P}$, where OPT is the size of the smallest guard cover for $\mathbf{P}$.*

5 Conclusions

We have proved polynomial time algorithms for approximate interior guarding of monotone and rectilinear polygons. Our contribution is that the approximation factors for both algorithms is independent of the size of the polygon. Interesting open problems are to improve the approximation bounds for monotone and rectilinear polygons, to find approximation algorithms for other classes of polygons, and ultimately approximate guarding of the general class of simple polygons.

References

1. A. AGGARWAL. *The Art Gallery Theorem: Its Variations, Applications and Algorithmic Aspects.* PhD thesis, Johns Hopkins University, 1984.
2. B. BEN-MOSHE, M.J. KATZ, J.S.B. MITCHELL. A Constant-Factor Approximation Algorithm for Optimal Terrain Guarding. In *Proc. 16th ACM-SIAM Symposium on Discrete Algorithms, SODA'05*, 2005.
3. B. BRODÉN, M. HAMMAR, B.J. NILSSON. Guarding Lines and 2-Link Polygons is APX-hard. In *Proc. 13th Canadian Conference on Computational Geometry, CCCG'01*, pages 45–48, 2001.

4. B. CHAZELLE. Triangulating a Simple Polygon in Linear Time. In *Proc. 31st Symposium on Foundations of Computer Science*, pages 220–230, 1990.
5. D.Z. CHEN, V. ESTIVILL-CASTRO, J. URRUTIA. Optimal Guarding of Polygons and Monotone Chains. In *Proc. 7th Canadian Conference on Computational Geometry, CCCG'95*, pages 133–138, 1995.
6. W. CHIN, S. NTAFOS. Optimum Watchman Routes. *Information Processing Letters*, 28:39–44, 1988.
7. V. CHVÁTAL. A Combinatorial Theorem in Plane Geometry. *Journal of Combinatorial Theory B*, 13(6):395–398, 1975.
8. K.L. CLARKSON, K. VARADARAJAN. Improved Approximation Algorithms for Geometric Set Cover. In *Proc. 21st ACM Symposium on Computational Geometry*, 2005.
9. J.C. CULBERSON, R.A. RECKHOW. Covering Polygons is Hard. In *Proc. 29th Symposium on Foundations of Computer Science*, pages 601–611, 1988.
10. M. DE BERG. On Rectilinear Link Distance. *Computational Geometry: Theory and Applications*, 1(1):13–34, 1991.
11. A. EFRAT, S. HAR-PELED. Locating Guards in Art Galleries. In *Proc. 2nd IFIP International Conference on Theoretical Computer Science*, 2002.
12. S. EIDENBENZ. Inapproximability Results for Guarding Polygons without Holes. In *Proc. 9th Annual International Symposium on Algorithms and Computation*, pages 427–436, 1998.
13. S. EIDENBENZ. *Inapproximability of Visibility Problems on Polygons and Terrains*. PhD thesis, ETH, Zurich, 2000.
14. S. FISK. A Short Proof of Chvátal's Watchman Theorem. *Journal of Combinatorial Theory B*, 24:374, 1978.
15. S.K. GHOSH. Approximation Algorithms for Art Gallery Problems. In *Proceedings of the Canadian Information Processing Society Congress*, 1987.
16. F. HOFFMANN, M. KAUFMANN, K. KRIEGEL. The Art Gallery Theorem for Polygons with Holes. In *Proc. 32nd IEEE Symposium on the Foundations of Computer Science*, pages 39–48, 1991.
17. B. JOE, R.B. SIMPSON. Correction to Lee's Visibility Polygon Algorithm. *BIT*, 27:458–473, 1987.
18. D.T. LEE. Visibility of a Simple Polygon. *Computer Vision, Graphics, and Image Processing*, 22:207–221, 1983.
19. D.T. LEE, A.K. LIN. Computational Complexity of Art Gallery Problems. *IEEE Transactions on Information Theory*, IT-32:276–282, 1986.
20. C. LEVCOPOULOS. *Heuristics for Minimum Decompositions of Polygons*. PhD thesis, University of Linköping, Linköping, Sweden, 1987.
21. B.J. NILSSON. *Guarding Art Galleries — Methods for Mobile Guards*. PhD thesis, Lund University, 1995.
22. J. O'ROURKE. *Art Gallery Theorems and Algorithms*. Oxford University Press, 1987.