



# Guarding curvilinear art galleries with vertex or point guards

Menelaos Karavelas, Elias Tsigaridas

## ► To cite this version:

Menelaos Karavelas, Elias Tsigaridas. Guarding curvilinear art galleries with vertex or point guards. [Research Report] RR-6132, 2007. inria-00132988v2

**HAL Id: inria-00132988**

**<https://inria.hal.science/inria-00132988v2>**

Submitted on 28 Feb 2007 (v2), last revised 18 Feb 2008 (v8)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

# *Guarding curvilinear art galleries with vertex or point guards*

Menelaos I. Karavelas — Elias P. Tsigaridas

N° ????

February 2007

Thème SYM

A large blue rectangle occupies the lower half of the page. Overlaid on it is the text 'Rapport de recherche' in a white serif font. To the left of the text is a large, light gray stylized 'R'. A horizontal gray brushstroke underline is positioned below the text.

*Rapport  
de recherche*



## Guarding curvilinear art galleries with vertex or point guards

Menelaos I. Karavelas<sup>\* †</sup>, Elias P. Tsigaridas<sup>‡ §</sup>

Thème SYM — Systèmes symboliques  
Projet VEGAS

Rapport de recherche n° 7777 — February 2007 — 30 pages

**Abstract:** One of the earliest and most well known problems in computational geometry is the so-called *art gallery problem*. The goal is to compute the minimum possible number guards placed on the vertices of a simple polygon in such a way that they cover the interior of the polygon.

In this paper we consider the problem of guarding an art gallery which is modeled as a polygon with curved walls. Our main focus is on polygons the edges of which are convex arcs pointing towards the exterior or interior of the polygon (but not both), named piecewise convex and piecewise concave polygons. We prove that, in the case of piecewise convex polygons, if we only allow vertex guards,  $\lfloor \frac{4n}{7} \rfloor - 1$  guards are sometimes necessary, and  $\lfloor \frac{2n}{3} \rfloor$  guards are always sufficient. Moreover, an  $O(n \log n)$  time and  $O(n)$  space algorithm is described that produces a vertex guarding set of size at most  $\lfloor \frac{2n}{3} \rfloor$ . In the special case of monotone piecewise convex polygons we can show that  $\lfloor \frac{n}{2} \rfloor$  vertex guards are always sufficient and sometimes necessary; these bounds remain valid even if we allow point guards.

In the case of piecewise concave polygons, we show that  $2n - 4$  point guards are always sufficient and sometimes necessary, whereas it might not be possible to guard such polygons by vertex guards. We conclude with lower bounds for other types of curved polygons and future work.

**Key-words:** art gallery, curved objects

This work is partially supported by the european project ACS (Algorithms for Complex Shapes, IST FET Open 006413) and ARC ARCADIA (<http://www.loria.fr/~petitjea/Arcadia/>)

<sup>\*</sup> University of Crete, Department of Applied Mathematics, GR-714 09 Heraklion, Greece

<sup>†</sup> Foundation for Research and Technology - Hellas, Institute of Applied and Computational Mathematics P.O. Box 1527, GR-711 10 Heraklion, Greece

<sup>‡</sup> LORIA-INRIA Lorraine, Nancy, FRANCE

<sup>§</sup> The work on this project started while at INRIA Sophia-Antipolis.

## **Guarding curvilinear art galleries with vertex or point guards**

**Résumé :**

**Mots-clés :** Pas de motclef

## 1 Introduction

Consider a simple polygon  $P$  with  $n$  vertices. How many points with omnidirectional visibility are required in order to see every point in the interior of  $P$ ? This problem, known as the *art gallery problem* has been one of the earliest problems in Computational Geometry. Applications areas include robotics [14], [28], motion planning [17], [21], computer vision and pattern recognition [25], [29], [2], [26], graphics [19], [6] and CAD/CAM [3], [10].

The first results on the art gallery problem or its variations date back to the 1970's. Chvátal [7] was the first to prove that a simple polygon with  $n$  vertices can be always guarded with  $\lfloor \frac{n}{3} \rfloor$  vertices; this bound is tight in the worst case. The proof by Chátal was quite tedious and Fisk [12] gave a much simpler proof by means of triangulating the polygon and coloring its vertices using three colors in such a way so that every triangle in the triangulation of the polygon does not contain two vertices of the same color. The algorithm proposed by Fisk runs in  $O(T(n) + n)$  time, where  $T(n)$  is the time to triangulate a simple polygon. Following Chazelle's linear-time algorithm for triangulating a simple polygon [4, 5], the algorithm proposed by Fisk runs in  $O(n)$  time. Lee and Lin [15] showed that computing the minimum number of vertex guards for a simple polygon is NP-hard, which was later on extended to point guards by Aggarwal [1]. Soon afterwards other types of polygons were considered. Kahn, Klawe and Kleitman [13] showed that orthogonal polygons of size  $n$ , i.e., polygons with axes-aligned edges, can be guarded with  $\lfloor \frac{n}{4} \rfloor$  vertex guards, which is also a lower bound. Several  $O(n)$  algorithms have been proposed for this variation of the problem, notably by Sack [23], who gave the first such algorithm, and later on by Lubiw [18]. Edelsbrunner, O'Rourke and Welzl [9] gave a linear time algorithm for guarding orthogonal polygons with  $\lfloor \frac{n}{4} \rfloor$  point guards.

Beside simple polygons and simple orthogonal polygons, polygons with holes, and orthogonal polygons with holes have been considered. As far as the type of guards is concerned, *edge guards* and *mobile guards* have been considered. An edge guard is an edge of the polygon, and a point is visible from it if it is visible from at least one point on the edge; mobile guards are essentially either edges of the polygon, or diagonals of the polygon. Other types of guarding problems have also been considered in the literature, notably, the *fortress problem* (guard the exterior of the polygon against enemy raids) and the *prison yard problem* (guard both the interior and the exterior of the polygon which represents a prison: prisoners must be guarded in the interior of the prison and should be allowed to escape out of the prison). For a detailed discussion of these variations and the corresponding results the interested reader should refer to the book by O'Rourke [22], the survey paper by Shermer [24] and the book chapter by Urrutia [27].

In this paper we consider the original problem, that is the problem of guarding a simple polygon. We are primarily interested in the case of vertex guards, although some results about points guards are also described. In our case, polygons are not required to have linear edges. On the contrary we consider polygons that have smooth curved edges. Clearly, these problems are NP-hard, since they are direct generalizations of the corresponding original art gallery problems. In the most general setting where we impose no restriction on the type of edges of the polygon, it is very easy to see that there exist curved polygons that cannot be guarded with vertex guards, or require an infinite number of point guards (see Fig. 16(b)). Restricting the edges of the polygon to be locally convex curves,

pointing towards the exterior of the polygon (i.e., the polygon is a locally convex set, except possibly at the vertices) we can construct polygons that require a minimum of  $n$  vertex or point guards, where  $n$  is the number of vertices of the polygon (see Fig. 16(a)). The main focus of this paper is the class of polygons that are either locally convex or locally concave (except possibly at the vertices), the edges of which are convex arcs; we call such polygons *piecewise convex* and *piecewise concave polygons*, respectively.

For the first class of polygons we show that it is always possible to guard them with  $\lfloor \frac{2n}{3} \rfloor$  vertex guards, where  $n$  is the number of polygon vertices. On the other hand we describe families of piecewise convex polygons that require a minimum of  $\lfloor \frac{4n}{7} \rfloor - 1$  vertex guards and  $\lfloor \frac{n}{2} \rfloor$  point guards. Aside from the combinatorial complexity type of results, we describe an  $O(n \log n)$  time and  $O(n)$  space algorithm which, given a piecewise convex polygon, computes a guarding set of size at most  $\lfloor \frac{2n}{3} \rfloor$ . Our algorithm should be viewed as a generalization of Fisk's algorithm [12]; in fact, when applied to polygons with linear edges, it produces a guarding set of size at most  $\lfloor \frac{n}{3} \rfloor$ . For the purposes of our complexity analysis and results, we assume, throughout the paper, that the curved edges of our polygons are arcs of algebraic curves of constant degree. The central idea for both obtaining the upper bound as well as for designing our algorithm is to approximate the piecewise convex polygon by a linear polygon (a polygon with line segments as edges). Additional auxiliary vertices are added on the boundary of the curved polygon in order to achieve this. The resulting linear polygon has the same topology as the original polygon and captures the essentials of the geometry of the piecewise convex polygon; for obvious reasons we term this linear polygon the *polygonal approximation*. Once the polygonal approximation has been constructed, we compute a guarding set for it by applying a slight modification of Fisk's algorithm [12]. The guarding set just computed for the polygonal approximation turns out to be a guarding set for the original curved polygon. The final step of both the proof and our algorithm consists in mapping the guarding set of the polygonal approximation to another vertex guarding set consisting of vertices of only the original polygon.

If we further restrict ourselves to *monotone piecewise convex polygons*, i.e., piecewise convex polygons that have the property that there exists a line  $L$ , such that any line  $L^\perp$  perpendicular to  $L$  intersects the polygon at most twice, we can show that  $\lfloor \frac{n}{2} \rfloor$  vertex or point guards are always sufficient and sometimes necessary. If the line  $L$  is known, it is very easy to compute the a vertex guarding set of size  $\lfloor \frac{n}{2} \rfloor$ : the problem of computing such a guarding set essentially reduces to sorting, thus taking  $O(n \log n)$  time and  $O(n)$  space. This result should be contrasted against the case of monotone linear polygons where the corresponding upper and lower bound on the number of vertex or point guards required to guard the polygon matches that of general (i.e., not necessarily monotone) linear polygons. In other words, monotonicity seems to play a crucial role in the case of piecewise convex polygons, which is not the case for linear polygons.

For the second class of polygons, i.e., the class of piecewise concave polygons, vertex guards may not be sufficient in order to guard the interior of the polygon (see Fig. 15(a)). We thus turn our attention to point guards, and we show that  $2n - 4$  point guards are always sufficient and sometimes necessary. Our method for showing the sufficiency result is similar to the technique used to illuminate sets of disjoint convex objects on the plane [11]. Given a piecewise concave polygon  $P$ , we construct a new piecewise concave polygon  $Q$ , contained inside  $P$ , and such that the tangencies between edges of  $Q$  are maximized. The problem of guarding  $P$  then reduces to the problem of

guarding  $Q$ , which essentially consists of a number of faces with pairwise disjoint interiors. The faces of  $Q$  require, each, two point guards in order to be guarded, and are in 1–1 correspondence with the triangles of an appropriately defined triangulation graph  $\mathcal{T}(R)$  of a polygon  $R$  with  $n$  vertices. Thus the number point guards required to guard  $P$  is at most two times the number of faces of  $\mathcal{T}(R)$ , i.e.,  $2n - 4$ .

The rest of the paper is structured as follows. In Section 2 we introduce some notation and provide various definitions. In Section 3 we present our algorithm for computing a guarding set, of size  $\lfloor \frac{2n}{3} \rfloor$  for a piecewise convex polygon with  $n$  vertices. Section 3 is further subdivided into five subsections. In Subsection 3.1 we define the polygonal approximation of our curved polygon and prove some geometric and combinatorial properties. In Subsection 3.2 we show how to construct a, properly chosen, *constrained* triangulation of the polygonal approximation. In Subsection 3.3 we describe how to compute the guarding set for the original curved polygon from the guarding set of the polygonal approximation due to Fisk’s algorithm and prove the upper bound on the cardinality of the guarding set. In Subsection 3.4 we show how to compute the guarding set in  $O(n \log n)$  time and  $O(n)$  space. Finally, in Subsection 3.5 is devoted to the presentation of the family of polygons that attains the lower bound of  $\lfloor \frac{4n}{7} \rfloor - 1$  vertex guards. The special case of guarding piecewise monotone piecewise convex polygons is discussed in Section 4. We show that  $\lfloor \frac{n}{2} \rfloor$  vertex (or point) guards are always necessary and sometimes sufficient, and present the  $O(n \log n)$  time and  $O(n)$  space algorithm for computing such a guarding set. In Section 5 we present our results for piecewise concave polygons, namely, that  $2n - 4$  point guards are always necessary and sometimes sufficient for this class of polygons. Section 6 contains further initial results. More precisely, we present lower bound constructions for two major variations of our main problem. The first kind of variation concerns the fact that we allow point guards. The second kind of variation refers to the type of polygons considered. The final section of the paper, Section 7 summarizes our results and discusses open problems.

## 2 Definitions

**Curved arcs.** Let  $S$  be a sequence of points  $v_1, \dots, v_n$  and  $E$  a set of curved arcs  $a_1, \dots, a_n$ , such that  $a_i$  has as endpoints the points  $v_i$  and  $v_{i+1}$ <sup>1</sup>. We will assume that the arcs  $a_i$  and  $a_j$ ,  $i \neq j$  do not intersect, except when  $j = i - 1$  or  $j = i + 1$ , in which case they intersect only at the points  $v_i$  and  $v_{i+1}$ , respectively. We define a *curved polygon*  $P$  to be the closed region delimited by the arcs  $a_i$ . The points  $v_i$  are called the vertices of  $P$ . An arc  $a_i$  is a *convex arc* if every line on the plane intersects  $a_i$  at at most two points. If  $q$  is a point in the interior of  $a_i$ , an  $\varepsilon$ -neighborhood  $n_\varepsilon(q)$  of  $q$  is defined to be the intersection of  $a_i$  with a disk centered at  $q$  with radius  $\varepsilon$ . An arc  $a_i$  is a *locally convex arc* if for every point  $q$  in the interior of  $a_i$ , there exists an  $\varepsilon_q$  such that for every  $0 < \varepsilon \leq \varepsilon_q$ , the  $\varepsilon$ -neighborhood of  $q$  lies entirely in one of the two halfspaces defined by the line  $\ell$  on the plane tangent to  $a_i$  at  $q$ . Note that a convex arc is also a locally convex arc.

<sup>1</sup>Indices are considered to be evaluated modulo  $n$ , unless otherwise stated.



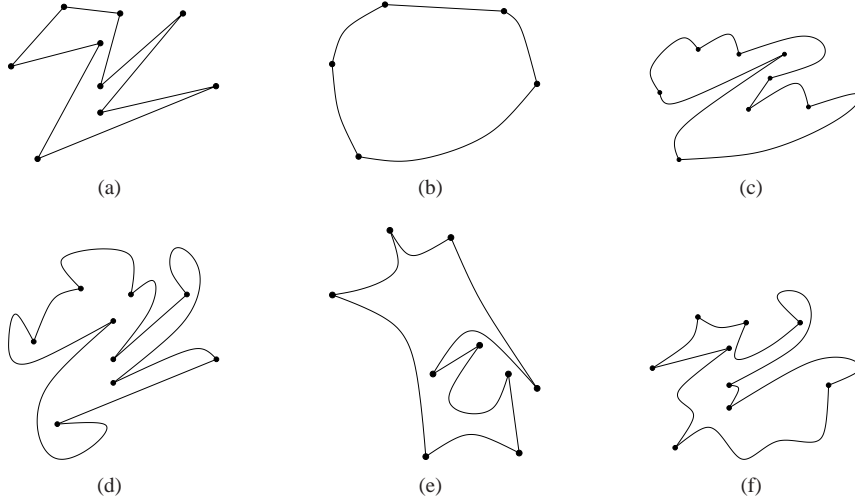


Figure 1: Different types of curved polygons: (a) a linear polygon, (b) a convex polygon, (c) a piecewise convex polygon, (d) a locally convex polygon, (e) a piecewise concave polygon and (f) a general polygon.

**Curved polygons.** A polygon  $P$  is a *linear polygon* if its edges are line segments (see Fig. 1(a)). A polygon  $P$  consisting of curved arcs as segments is called a *convex polygon* if every line on the plane intersects its boundary at most twice (see Fig. 1(b)). A polygon is called a *piecewise convex polygon*, if every arc is a convex arc and for every point  $q$  in the interior of an arc  $a_i$  of the polygon, the interior of the polygon is locally on the same side as the arc  $a_i$  with respect to the line tangent to  $a_i$  at  $q$  (see Fig. 1(c)). A polygon is called a *locally convex polygon* if the boundary of the polygon is a locally convex curve, except possibly at its vertices (see Fig. 1(d)). Note that a convex polygon is a piecewise convex polygon and that a piecewise convex polygon is also a locally convex polygon. A polygon  $P$  is called a *piecewise concave polygon*, if every arc of  $P$  is convex and for every point  $q$  in the interior of a non-linear arc  $a_i$ , the interior of  $P$  lies locally on both sides of the line tangent to  $a_i$  at  $q$  (see Fig. 1(e)). Finally, a polygon is said to be a *general polygon* if we impose no restrictions on the type of its edges (see Fig. 1(f)). We will use the term *curved polygon* to refer to a polygon the edges of which are either line or curve segments.

**Guards and guarding sets.** In our setting, a *guard* or *point guard* is a point in the interior or the boundary of a curved polygon  $P$ . If a guard of  $P$  that is also a vertex of  $P$  is called a *vertex guard*. We say that a curved polygon  $P$  is *guarded* by a set  $G$  of guards if every point in  $P$  is visible from at least one point in  $G$ . The set  $G$  that has this property is called a *guarding set* for  $P$ . A guarding set that consists solely of vertices of  $P$  is called a *vertex guarding set*.

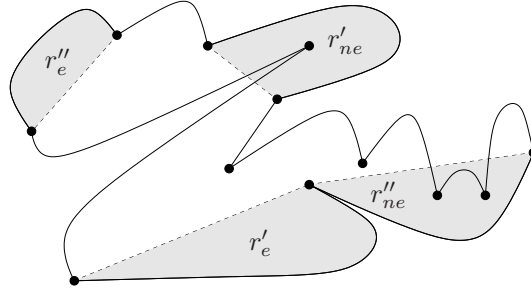


Figure 2: The two types of rooms in a piecewise convex polygon:  $r'_e$  and  $r''_e$  are empty rooms, whereas  $r'_{ne}$  and  $r''_{ne}$  are non-empty rooms.

### 3 Piecewise convex polygons

In this section we present an algorithm which, given a piecewise convex polygon  $P$  of size  $n$ , it computes a guarding set  $G$  of size  $\lfloor \frac{2n}{3} \rfloor$ . The basic steps of the algorithm are as follows:

1. Compute the polygonal approximation  $\tilde{P}$  of  $P$ .
2. Compute a constrained triangulation  $\mathcal{T}(\tilde{P})$  of  $\tilde{P}$ .
3. Compute a guarding set  $G_{\tilde{P}}$  for  $\tilde{P}$ , by coloring the vertices of  $\mathcal{T}(\tilde{P})$  using three colors.
4. Compute a guarding set  $G_P$  for  $P$  from the guarding set  $G_{\tilde{P}}$ .

#### 3.1 Polygonalization of a piecewise convex polygon

Let  $a_i$  be a convex arc with endpoints  $v_i$  and  $v_{i+1}$ . We call the convex region  $r_i$  delimited by  $a_i$  and the line segment  $v_i v_{i+1}$  a *room*. A room is called degenerate if the arc  $a_i$  is a line segment. A line segment  $pq$ , where  $p, q \in a_i$  is called a *chord*, and the region delimited by the chord  $pq$  and  $a_i$  is called a *sector*. The chord of a room  $r_i$  is defined to be the line segment  $v_i v_{i+1}$  connecting the endpoints of the corresponding arc  $a_i$ . A degenerate sector is a sector with empty interior. We distinguish between two types of rooms (see Fig. 2):

1. *empty rooms*: these are non-degenerate rooms that do not contain any vertex of  $P$  in the interior of  $r_i$  or in the interior of the chord  $v_i v_{i+1}$ .
2. *non-empty rooms*: these are non-degenerate rooms that contain at least one vertex of  $P$  in the interior of  $r_i$  or in the interior of the chord  $v_i v_{i+1}$ .

In order to polygonalize  $P$  we are going to add new vertices in the interior of non-linear convex arcs. To distinguish between the two types of vertices, the  $n$  vertices of  $P$  will be called *original vertices*, whereas the additional vertices will be called *auxiliary vertices*.

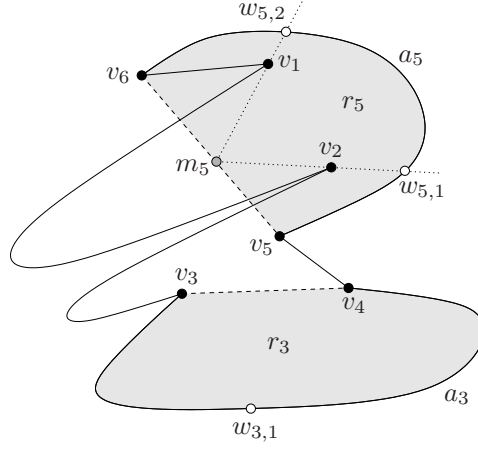


Figure 3: The auxiliary vertices (white points) for rooms  $r_3$  (empty) and  $r_5$  (non-empty).  $w_{3,1}$  is a point in the interior of  $a_3$ .  $m_5$  is the midpoint of  $v_5$  and  $v_6$ , whereas  $w_{3,1}$  and  $w_{5,2}$ , respectively, are the intersections of the lines passing through  $m_5$ ,  $v_2$  and  $m_5$ ,  $v_1$ , with the arc  $a_5$ . In this example  $R_5 = C_5^* = \{v_1, v_2\}$ .

More specifically, for each empty room  $r_i$  we add a vertex  $w_{i,1}$  (anywhere) in the interior of the arc  $a_i$  (see Fig. 3). For each non-empty room  $r_i$ , let  $X_i$  be the set of vertices of  $P$  that lie in the interior of the chord  $v_i v_{i+1}$  of  $r_i$ , and  $R_i$  be the set of vertices of  $P$  that are contained in the interior of  $r_i$  or belong to  $X_i$  (by assumption  $R_i \neq \emptyset$ ). If  $R_i \neq X_i$ , let  $C_i$  be the set of vertices on the convex hull of the vertex set  $(R_i \setminus X_i) \cup \{v_i, v_{i+1}\}$ ; if  $R_i = X_i$ , let  $C_i = X_i \cup \{v_i, v_{i+1}\}$ . Finally, let  $C_i^* = C_i \setminus \{v_i, v_{i+1}\}$ . Clearly,  $v_i$  and  $v_{i+1}$  belong to the set  $C_i$  and, furthermore,  $C_i^* \neq \emptyset$ .

Let  $m_i$  be the midpoint of  $v_i v_{i+1}$  and  $\ell_i^\perp(p)$  the line perpendicular to  $v_i v_{i+1}$  passing through a point  $p$ . If  $C_i^* \neq X_i$ , then, for each  $v_k \in C_i^*$ , let  $w_{i,j_k}$  be the (unique) intersection of the line  $v_k m_i$  with the arc  $a_i$ ; if  $C_i^* = X_i$ , then, for each  $v_k \in C_i^*$ , let  $w_{i,j_k}$  be the (unique) intersection of the line  $\ell_i^\perp(v_k)$  with the arc  $a_i$ .

Now consider the sequence  $\tilde{S}$  of the original vertices of  $P$  augmented by the auxiliary vertices added to empty and non-empty rooms; the order of the vertices in  $\tilde{S}$  is the order in which we encounter them as we traverse the boundary of  $P$  in the counterclockwise order. The linear polygon defined by the sequence  $\tilde{S}$  of vertices is denoted by  $\tilde{P}$  (see Fig. 4(a)). It is easy to show that:

**Lemma 1** *The linear polygon  $\tilde{S}$  is a simple polygon.*

*Proof.* It suffices show that the line segments replacing the curved segments of  $P$  do not intersect other edges of  $P$  or  $\tilde{P}$ .

Let  $r_i$  be an empty room, and let  $w_{i,1}$  be the point added in the interior of  $a_i$ . The interior of the line segments  $v_i w_{i,1}$  and  $w_{i,1} v_{i+1}$  lie in the interior of  $r_i$ . Since  $P$  is a piecewise convex

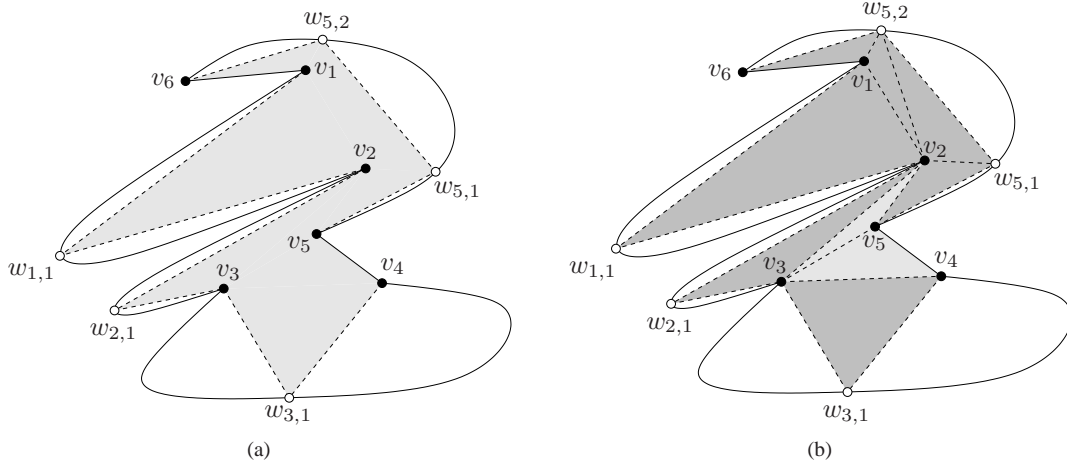


Figure 4: (a) The polygonal approximation  $\tilde{P}$ , shown in gray, of the piecewise convex polygon  $P$  with vertices  $v_i, i = 1, \dots, 6$ . (b) The constrained triangulation  $\mathcal{T}(\tilde{P})$  of  $\tilde{P}$ . The dark gray triangles are the constrained triangles. The polygonal region  $v_5 w_{5,1} w_{5,2} v_6 v_1 v_2 v_5$  is a crescent. The triangles  $w_{5,1} v_2 v_5$  and  $v_1 w_{5,2} v_6$  are boundary crescent triangles. The triangle  $v_2 w_{5,2} v_1$  is an upper crescent triangle, whereas the triangle  $v_2 w_{5,1} w_{5,2}$  is a lower crescent triangle.

polygon, and  $r_i$  is an empty room, no edge of  $P$  could potentially intersect  $v_i w_{i,1}$  or  $w_{i,1} v_{i+1}$ . Hence replacing  $a_i$  by the polyline  $v_i w_{i,1} v_{i+1}$  gives us a new piecewise convex polygon.

Let  $r_i$  be a non-empty room. Let  $w_{i,1}, \dots, w_{i,K}$  be the points added on  $a_i$ , where  $K$  is the cardinality of  $C_i^*$ . By construction, every point  $w_{i,k}$  is visible from  $w_{i,k+1}$ ,  $k = 1, \dots, K-1$ , and every point  $w_{i,k}$  is visible to  $w_{i,k-1}$ ,  $k = 2, \dots, K$ . Moreover,  $w_{i,1}$  is visible from  $v_i$  and  $w_{i,K}$  is visible from  $v_{i+1}$ . Therefore, the interior of the segments in the polyline  $v_i w_{i,1} \dots w_{i,K} v_{i+1}$  lie in the interior of  $r_i$  and do not intersect any arc in  $P$ . Hence, substituting  $a_i$  by the polyline  $v_i w_{i,1} \dots w_{i,K} v_{i+1}$  gives us a new piecewise convex polygon.

As a result, the linear polygon  $\tilde{P}$  is a simple polygon.  $\square$

We call the linear polygon  $\tilde{P}$ , defined by  $\tilde{S}$ , the *straight-line polygonal approximation* of  $P$ , or simply the *polygonal approximation* of  $P$ . An obvious result for  $\tilde{P}$  is the following:

**Corollary 2** *If  $P$  is a piecewise convex polygon the polygonal approximation  $\tilde{P}$  of  $P$  is a linear polygon that is contained inside  $P$ .*

We end this section by proving a tight upper bound on the size of the polygonal approximation of a piecewise convex polygon. We start by stating and proving an intermediate result, namely that the sets  $C_i^*$  are pairwise disjoint.

**Lemma 3** *Let  $i, j$ , with  $1 \leq i < j \leq n$ . Then  $C_i^* \cap C_j^* = \emptyset$ .*

*Proof.* If one of the rooms  $r_i$  and  $r_j$  is a degenerate or an empty room, the result is obvious.

Consider two non-empty rooms  $r_i$  and  $r_j$ . Suppose that there exists a vertex  $u \in P$  that is contained in  $C_i^* \cap C_j^*$ . Let  $v_i, v_{i+1}$ , and  $v_j, v_{j+1}$  be the endpoints of the arcs  $a_i$  and  $a_j$ , and  $m_i, m_j$  the midpoints of the chords  $v_i v_{i+1}$ ,  $v_j v_{j+1}$ , respectively. Let  $u_i$  be the intersection of the line  $m_i u$  with the convex arc  $a_i$  and  $u_j$  be the intersection of the line  $m_j u$  with the convex arc  $a_j$ , respectively. Consider the following two cases. The ones that are not listed below are symmetric to one of the cases considered below and analogous arguments can be applied.

1. *The vertices  $v_j$  and  $v_{j+1}$  belong to  $R_i$ .* Since  $u$  belongs to  $C_i^*$ , the line segment  $uu_i$  cannot contain any vertices of  $P$  and it cannot intersect any edge of  $P$  (since otherwise  $u$  would not belong to  $C_i^*$ ). Since  $u$  belongs to  $C_j^*$ , the chord of  $r_j$  has to intersect the line segment  $uu_i$ . We distinguish between the following two cases:
  - (a) *The chord  $v_j v_{j+1}$  intersects the interior of  $uu_i$ .* Let  $x$  be the point of intersection. Depending on the orientation of the triplets of vertices  $\{v_j, v_{j+1}, v_i\}$  and  $\{v_j, v_{j+1}, v_{i+1}\}$ ,  $u$  will either be in the interior of the triangle  $v_i v_{i+1} v_j$  or in the interior of the triangle  $v_i v_{i+1} v_{j+1}$  or inside the convex quadrilateral defined by the four vertices  $v_i, v_{i+1}, v_j$  and  $v_{j+1}$ ; in any of the three cases,  $u$  cannot be a vertex in  $C_i^*$ , hence a contradiction.
  - (b) *The chord  $v_j v_{j+1}$  intersects  $uu_i$  at  $u$ .* We can assume without loss of generality that  $v_{i+1}, v_j$  are to the right and  $v_i, v_{j+1}$  to the left of the oriented line  $u_i u$ . Notice that both  $v_j$  and  $v_{j+1}$  have to belong to  $C_i^*$ , since otherwise  $u$  would not belong to  $C_i^*$ . Let  $v'_j$  and  $v'_{j+1}$  be the intersections of the lines  $m_i v_j$  and  $m_i v_{j+1}$  with  $a_i$ . Consider the path  $\pi$  from  $u$  to  $v_i$  on  $\partial P$ , that does not contain the edge  $a_j$ .  $\pi$  has to intersect either the line segment  $v_j v'_j$  or the line segment  $v_{j+1} v'_{j+1}$ ; either case this yields a contradiction with the fact that both  $v_j$  and  $v_{j+1}$  belong to  $C_i^*$ .
2. *The vertex  $v_i$  belongs to  $R_j$  and the vertex  $v_j$  belongs to  $R_i$ .* Since  $u$  belongs to both  $C_i^*$  and  $C_j^*$ , it has to lie on the line segment  $v_i v_j$ . Moreover, both  $v_i$  and  $v_j$  have to belong to  $C_i^*$  and  $C_j^*$ , respectively, since otherwise  $u$  would not belong to  $C_i^*$  and  $C_j^*$ , respectively. Let  $v'_i$  and  $v'_j$  be the intersections of the lines  $m_j v_i$  and  $m_i v_j$  with the arcs  $a_j$  and  $a_i$ , respectively. Consider the path  $\pi$  on  $\partial P$  from  $u$  to  $v_{j+1}$  that does not contain the edge  $a_j$ .  $\pi$  has to either intersect the line segment  $v_i v'_i$  or the line segment  $v_j v'_j$ . In the former case, we get a contradiction with the fact that  $v_i$  belongs to  $C_j^*$ ; in the latter case we get a contradiction with the fact that  $v_j$  belongs to  $C_i^*$ .  $\square$

An immediate consequence of Lemma 3 is the following corollary that gives us a tight bound on the size of the polygonal approximation  $\tilde{P}$  of  $P$ .

**Corollary 4** *If  $n$  is the size of a piecewise convex polygon  $P$ , the size of its polygonal approximation  $\tilde{P}$  is at most  $3n$ . This bound is tight (up to a constant).*

*Proof.* Let  $a_i$  be a convex arc of  $P$ , and let  $r_i$  be the corresponding room. If  $a_i$  is an empty room, then  $\tilde{P}$  contains one auxiliary vertex due to  $a_i$ . Hence  $\tilde{P}$  contains at most  $n$  auxiliary vertices attributed

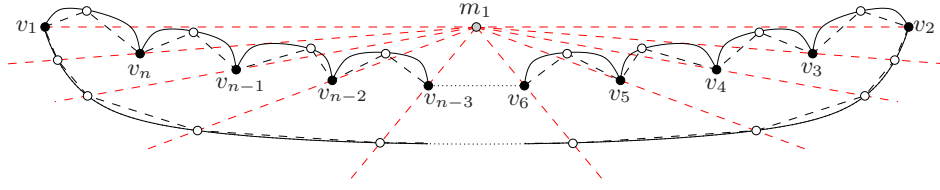


Figure 5: A piecewise convex polygon  $P$  of size  $n$  (black solid curve), the polygonal approximation  $\tilde{P}$  of which consists of  $3n - 3$  vertices (black dashed curve).

to empty rooms in  $P$ . If  $a_i$  is a non-empty room, then  $\tilde{P}$  contains  $|C_i^*|$  auxiliary vertices due to  $a_i$ . By Lemma 3 the sets  $C_i^*$ ,  $i = 1, \dots, n$  are pairwise disjoint, which implies that  $\sum_{i=1}^n |C_i^*| \leq |P| = n$ . Therefore  $\tilde{P}$  contains the  $n$  vertices of  $P$ , contains at most  $n$  vertices due to empty rooms in  $P$  and at most  $n$  vertices due to non-empty rooms in  $P$ . We thus conclude that the size of  $\tilde{P}$  is at most  $3n$ .

The afore-mentioned bound is tight up to a constant. Consider the piecewise convex polygon  $P$  of Fig. 5. It consists of  $n - 1$  empty rooms and one non-empty room  $r_1$ , such that  $|C_1^*| = n - 2$ . It is easy to see that  $|\tilde{P}| = 3n - 3$ .  $\square$

### 3.2 Triangulating the polygonal approximation

Let  $P$  be a piecewise convex polygon and  $\tilde{P}$  is its polygonal approximation. We are going to construct a *constrained triangulation* of  $\tilde{P}$ , i.e., we are going to triangulate  $\tilde{P}$ , while enforcing some triangles to be part of this triangulation. Let  $P^\alpha = \tilde{P} \setminus P$  be the set of auxiliary vertices in  $\tilde{P}$ . The main idea behind the way this particular triangulation is constructed is to enforce that:

1. all triangles of  $\mathcal{T}(\tilde{P})$ , that contain a vertex in  $P^\alpha$ , also contain at least one vertex of  $P$ ,
2. every vertex in  $P^\alpha$  belongs to at least one triangle in  $\mathcal{T}(\tilde{P})$  the other two vertices of which are both vertices of  $P$ , and
3. the triangles of  $\mathcal{T}(\tilde{P})$  that contain vertices of  $\tilde{P}$  can be guarded by vertices of  $P$ .

These properties are going to be exploited in Step 4 of the algorithm presented in Section 3.

More precisely, we are going to enforce the way the triangles of  $\mathcal{T}(\tilde{P})$  are created in the neighborhoods of the vertices in  $P^\alpha$ . By enforcing the triangles in these neighborhoods, we effectively triangulate parts of  $\tilde{P}$ . The remaining untriangulated parts of  $\tilde{P}$  consist of one or more disjoint polygons, which can then be triangulated by means of any polygon triangulation algorithm. In other words, the triangulation of  $\tilde{P}$  that we want to construct is a constrained triangulation, in the sense that we pre-specify some of the edges of the triangulation. In fact, as we will see below we pre-specify triangles, rather than edges, which are going to be referred to as *constrained triangles*.

Let us proceed to define the constrained triangles in  $\mathcal{T}(\tilde{P})$ . If  $r_i$  is an empty room, and  $w_{i,1}$  is the point added on  $a_i$ , add the edges  $v_i v_{i+1}$ ,  $v_i w_{i,1}$  and  $w_{i,1} v_{i+1}$ , thus formulating the constrained

triangle  $v_i w_{i,1} v_{i+1}$  (see Fig. 4(b)). If  $r_i$  is a non-empty room,  $\{c_1, \dots, c_K\}$  the vertices in  $C_i^*$ , and  $\{w_{i,1}, \dots, w_{i,K}\}$  the vertices added on  $a_i$ , add the following edges, if they do not already exist:

1.  $c_k, c_{k+1}, k = 1, \dots, K-1, v_i c_1, c_K v_{i+1}$
2.  $c_i w_{i,k}, k = 1, \dots, K$
3.  $c_i w_{i,k+1}, k = 1, \dots, K-1$
4.  $w_{i,k}, w_{i,k+1}, k = 1, \dots, K-1, v_i w_{i,1}, w_{i,K} v_{i+1}$

These edges formulate  $2K$  constrained triangles, namely,  $c_k c_{k+1} w_{i,k+1}, k = 1, \dots, K-1, c_k w_{i,k} w_{i,k+1}, k = 1, \dots, K-1, v_i c_1 w_{i,1}$  and  $v_{i+1} c_K w_{i,K}$ . We call the polygonal region delimited by these triangles a *crescent*. The triangles  $v_i c_1 w_{i,1}$  and  $v_{i+1} c_K w_{i,K}$  are called *boundary crescent triangles*, the triangles  $c_k c_{k+1} w_{i,k+1}, k = 1, \dots, K-1$  are called *upper crescent triangles* and the triangles  $c_k w_{i,k} w_{i,k+1}, k = 1, \dots, K-1$  are called *lower crescent triangles*.

Note that almost all the points in  $P^\alpha$  belong to exactly one triangle the other two points of which are in  $P$ ; the only exception are the points  $w_{i,K}$  which belong to exactly two such triangles.

As we have already mentioned, having created the constrained triangles mentioned above, there may exist additional possibly disjoint polygonal non-triangulated regions of  $\tilde{P}$ . The triangulation procedure continues by triangulating these additional polygonal non-triangulated regions; any polygon triangulation algorithm may be used.

### 3.3 Computing a guarding set for the original polygon

To compute a guarding set for  $P$  we will perform the following two steps:

1. Compute a guarding set  $G_{\tilde{P}}$  for  $\tilde{P}$ . We are going to show that this is also a guarding set for  $P$ .
2. From the guarding set  $G_{\tilde{P}}$  we are going to define a subset  $G_P$  of  $P$ , which consists only of vertices of  $P$ . We are going to show that  $G_P$  is a guarding set for  $P$  and that its size is at most  $\lfloor \frac{2n}{3} \rfloor$ .

Assume that we have colored the vertices of  $\tilde{P}$  with three colors, so that every triangle in  $\mathcal{T}(\tilde{P})$  does not contain two vertices of the same color. This can be easily done by the standard three-coloring algorithm for linear polygons presented in [20, 12]. Let red, green and blue be the three colors, and let  $K_A$  be the set of vertices of red color,  $\Pi_A$  be the set of vertices of green color and  $M_A$  be the set of vertices of blue color in a subset  $A$  of  $\tilde{P}$ . Clearly, all three sets  $K_{\tilde{P}}, \Pi_{\tilde{P}}$  and  $M_{\tilde{P}}$  are guarding sets for  $\tilde{P}$ . In fact, they are also guarding sets for  $P$ , as the following theorem suggests (see also Fig. 6).

**Theorem 5** *Each one of the sets  $K_{\tilde{P}}, \Pi_{\tilde{P}}$  and  $M_{\tilde{P}}$  is a guarding set for  $P$ .*

*Proof.* Let  $G_{\tilde{P}}$  be one of  $K_{\tilde{P}}, \Pi_{\tilde{P}}$  and  $M_{\tilde{P}}$ . By construction,  $G_{\tilde{P}}$  guards all triangles in  $\mathcal{T}(\tilde{P})$ . To show that  $G_{\tilde{P}}$  is a guarding set for  $P$ , it suffices to show that  $G_{\tilde{P}}$  also guards the non-degenerate sectors defined by the edges in  $\tilde{P}$  and the convex arcs in  $P$ .

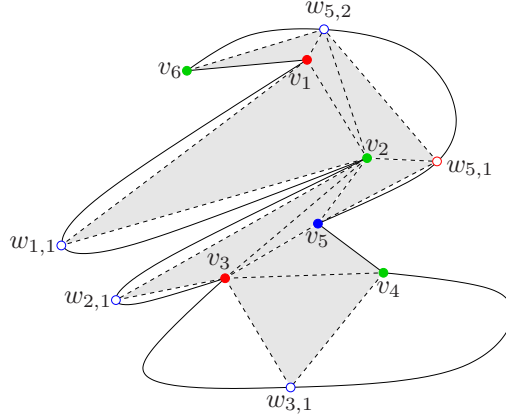


Figure 6: The three guarding sets for  $\tilde{P}$ , are also guarding sets for  $P$ , as Theorem 5 suggests.

Let  $s_i$  be a non-degenerate sector associated with the convex arc  $a_i$ . We consider the following two cases:

1. The room  $r_i$  is an empty room. Then  $s_i$  is adjacent to the triangle  $v_i w_{i,1} v_{i+1}$  of  $\mathcal{T}(\tilde{P})$ . Note that since  $a_i$  is a convex arc, all three points  $v_i$ ,  $v_{i+1}$  and  $w_{i,1}$  guard  $s_i$ . Since one of them has to be in  $G_{\tilde{P}}$ , we conclude that  $G_{\tilde{P}}$  guards  $s_i$ .
2. The room  $r_i$  is a non-empty room. Then  $s_i$  is adjacent to either a boundary crescent triangle or a lower crescent triangle in  $\mathcal{T}(\tilde{P})$ . Let  $T$  be this triangle, and let  $x$ ,  $y$  and  $z$  be its vertices. Since  $a_i$  is a convex arc, all three  $x$ ,  $y$  and  $z$  guard  $s_i$ . Therefore, since one of the three vertices  $x$ ,  $y$  and  $z$  is in  $G_{\tilde{P}}$ , we conclude that  $G_{\tilde{P}}$  guards  $s_i$ .

Therefore every non-degenerate sector in  $P^\alpha$  is guarded by at least one vertex in  $G_{\tilde{P}}$ , which implies that  $G_{\tilde{P}}$  is a guarding set for  $P$ .  $\square$

Let us now assume, without loss of generality that, among  $K_P$ ,  $\Pi_P$  and  $M_P$ ,  $K_P$  has the smallest cardinality and that  $\Pi_P$  has the second smallest cardinality, i.e.,  $|K_P| \leq |\Pi_P| \leq |M_P|$ . We are going to define a mapping  $f$  from  $K_{\tilde{P}} \setminus K_P$  to the power set  $2^{\Pi_P}$  of  $\Pi_P$ . Intuitively,  $f$  maps a vertex  $x$  in  $K_{\tilde{P}} \setminus K_P$  to all the neighboring vertices of  $x$  in  $\mathcal{T}(\tilde{P})$  that belong to  $\Pi_P$ . We are going to give a more precise definition of  $f$  in the below (also consult Fig. 7). Let  $x \in K_{\tilde{P}} \setminus K_P$ . We distinguish between the following cases:

1.  $x$  is an auxiliary vertex added to an empty arc  $a_i$  (see Fig. 7(a)). Then  $x$  is one of the vertices of the constrained triangle  $v_i v_{i+1} x$  contained inside  $r_i$ . One of  $v_i$ ,  $v_{i+1}$  must be a vertex in  $\Pi_P$ , say  $v_{i+1}$ . Then we set  $f(x) = \{v_{i+1}\}$ .
2.  $x$  is an auxiliary vertex added to a non-empty arc  $a_i$ . Consider the following subcases:



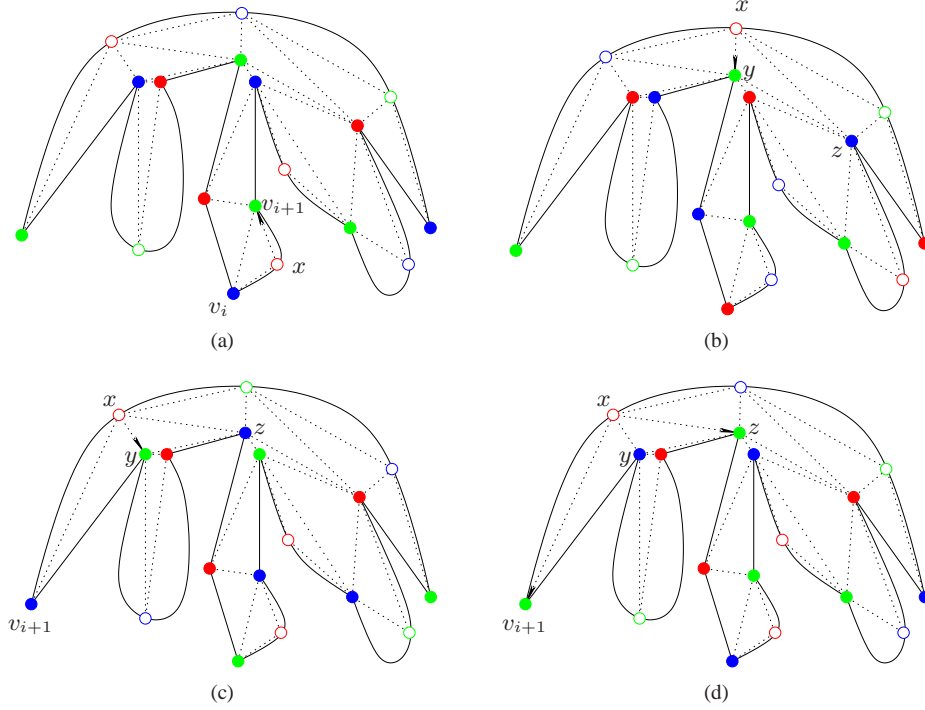


Figure 7: The three cases in the definition of the mapping  $f$ . Case (a):  $x$  is an auxiliary vertex in an empty room. Case (b):  $x$  is an auxiliary vertex in a non-empty room and is not the last auxiliary vertex added on the curved arc. Cases (c) and (d):  $x$  is the last auxiliary vertex added on the curved arc of a non-empty room (in (c) only one of its neighbors in  $P$  is green, whereas in (d) two of its neighbors in  $P$  are green).

- (a)  $x$  is not the last auxiliary vertex on  $a_i$ , as we walk along  $a_i$  in the counterclockwise sense (see Fig. 7(b)). Then  $x$  is incident to a single triangle in  $\mathcal{T}(\tilde{P})$  the other two vertices of which are vertices in  $P$ . Let  $y$  and  $z$  be these other two vertices. One of  $y$  and  $z$  has to be a green vertex, say  $y$ . Then we set  $f(x) = \{y\}$ .
- (b)  $x$  is the last auxiliary vertex on  $a_i$  as we walk along  $a_i$  in the counterclockwise sense (see Figs. 7(c) and 7(d)). Then  $x$  is incident to a boundary crescent triangle and an upper crescent triangle. Let  $xv_{i+1}y$  be the boundary crescent triangle and  $xyz$  the upper crescent triangle. Clearly, all three vertices  $v_{i+1}$ ,  $y$  and  $z$  are vertices of  $P$ . If  $y \in \Pi_P$  (this is the case in Fig. 7(c)), then we set  $f(x) = \{y\}$ . Otherwise (this is the case in Fig. 7(d)), both  $v_{i+1}$  and  $z$  have to be green vertices, in which case we set  $f(x) = \{v_{i+1}, z\}$ .

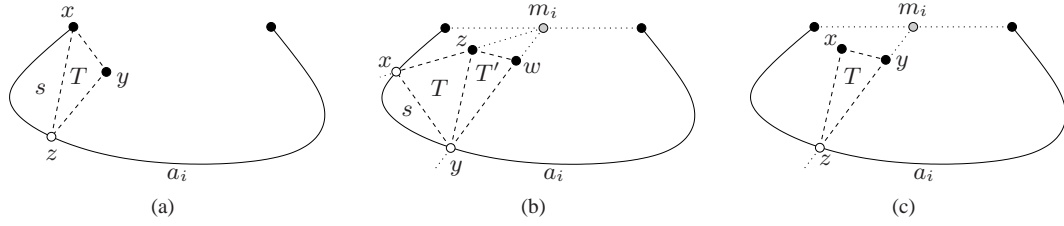


Figure 8: Three of the five cases in the proof of Lemma 6: (a) the triangle  $T$  is a boundary crescent triangle; (b) the triangle  $T$  is a lower crescent triangle; (c) the triangle  $T$  is an upper crescent triangle.

Now define the set  $G_P = K_P \cup \left( \bigcup_{x \in K_{P^\alpha}} f(x) \right)$ . We claim that  $G_P$  is a guarding set for  $P$ .

**Lemma 6** *The set  $G_P = K_P \cup \left( \bigcup_{x \in K_{P^\alpha}} f(x) \right)$  is a guarding set for  $P$ .*

*Proof.* Let us consider the triangulation  $\mathcal{T}(\tilde{P})$  of  $\tilde{P}$ . The regions in  $P^\alpha$  are sectors defined by a curved arc, which is a subarc of an edge of  $P$  and the corresponding chord connecting the endpoints of this subarc. Let us consider the set of triangles in  $\mathcal{T}(\tilde{P})$  and the set  $\mathcal{S}(P)$  of sectors in  $P^\alpha$ . To show that  $G_P$  is a guarding set for  $P$ , it suffices show that every triangle in  $\mathcal{T}(\tilde{P})$  and every sector in  $\mathcal{S}(P)$  is guarded by at least one vertex in  $G_P$ .

If  $T$  is a triangle in  $\mathcal{T}(\tilde{P})$  that is defined over vertices of  $P$ , one of its vertices is colored red and belongs to  $K_P \subseteq G_P$ . Hence,  $T$  is guarded.

Consider now a triangle  $T$  that is defined inside an empty room  $r_i$ . If the auxiliary vertex of  $T$  is not red, then one of the two endpoints of  $a_i$  has to be red, and thus it belongs to  $G_P$ . Hence both  $T$  and the two sectors adjacent to it in  $r_i$  are guarded. If the auxiliary vertex is red, then one of the other two vertices of  $T$  is green and belongs to  $G_P$ ; again,  $T$  is guarded.

Suppose now that  $T$  is a boundary crescent triangle, and let  $s$  be the sector adjacent to it (consult Fig. 8(a)). Let  $x$  be the endpoint of  $a_i$  contained in  $T$ ,  $y$  be the second point of  $T$  that belongs to  $P$  and  $z$  the point in  $P^\alpha$ . Note that all three vertices guard the sector  $s$ . If  $x$  (resp.  $y$ ) is a red vertex it will also be a vertex in  $G_P$ . Hence, in this case both  $T$  and  $s$  are guarded by  $x$  (resp.  $y$ ). If  $z$  is the red vertex in  $T$ , either  $x$  or  $y$  has to be a green vertex. Hence either  $x$  or  $y$  will be in  $G_P$ , and thus again both  $T$  and  $s$  will be guarded.

If  $T$  is a lower crescent triangle, let  $s$  be the sector adjacent to it (consult Fig. 8(b)). Let  $x, y$  be the endpoints of the chord of  $s$  on  $a_i$  and let  $z$  be the point of  $P$  in  $T$ . Let us also assume we encounter  $x$  and  $y$  in that order as we walk along  $a_i$  in the counterclockwise sense, which implies that  $x$  is the intersection of the line  $zm_i$  and the arc  $a_i$ . Finally, let  $T'$  be the upper crescent triangle incident to the edge  $yz$ , and let  $w$  be the third vertex of  $T'$ , beyond  $y$  and  $z$ . It is interesting to note that all four vertices  $x, y, z$  and  $w$  guard  $T, T'$  and  $s$ . Moreover,  $x$  and  $w$  have to be of the same color. In order to show that  $T$  and  $s$  are guarded by  $G_P$ , it suffices to show that one of  $x, y, z$  and  $w$  belongs to  $G_P$ . Consider the following cases:

1.  $z$  is a red vertex. Since  $z \in K_P$ , we get that  $z \in G_P$ .
2.  $x$  is a red vertex. But then  $w$  is also a red vertex. Since  $w \in K_P$ , we conclude that  $w$  belongs to  $G_P$  as well.
3.  $y$  is a red vertex. Then either  $z$  is a green vertex or both  $x$  and  $w$  are green vertices. If  $z$  is a green vertex, then  $\{z\} \subseteq f(y)$ , which implies that  $z \in G_P$ . If  $z$  is a blue vertex, then both  $x$  and  $w$  are green vertices, and in particular  $\{w\} \subseteq f(y)$ . Hence  $w \in G_P$ .

Finally, consider the case that  $T$  is an upper crescent triangle, let  $x$  and  $y$  be the vertices of  $P$  in  $T$  and let  $z$  be the vertex of  $T$  in  $P^\alpha$  (consult Fig. 8(c)). Let us also assume that  $z$  is the intersection of the line  $ym_i$  with  $a_i$ . To show that  $T$  is guarded by  $G_P$ , it suffices to show that one of  $x$ ,  $y$  and  $z$  belongs to  $G_P$ . Consider the following cases:

1.  $x$  is red vertex. Since  $x \in K_P$  we have that  $x \in G_P$ .
2.  $y$  is red vertex. Since  $y \in K_P$  we have that  $y \in G_P$ .
3.  $z$  is a red vertex. If  $x$  is a green vertex, then  $\{x\} \subseteq f(z)$ . Hence  $x \in G_P$ . If  $x$  is blue vertex, then  $y$  has to be a green vertex, and  $\{y\} \subseteq f(z)$ . Therefore,  $y \in G_P$ .  $\square$

Since  $f(x) \subseteq \Pi_P$  for every  $x$  in  $K_{P^\alpha}$  we get that  $\bigcup_{x \in K_{P^\alpha}} f(x) \subseteq \Pi_P$ . But this, in turn implies that  $G_P \subseteq K_P \cup \Pi_P$ . Since  $K_P$  and  $\Pi_P$  are the two sets of smallest cardinality among  $K_P$ ,  $\Pi_P$  and  $M_P$ , we can easily verify that  $|K_P| + |\Pi_P| \leq \lfloor \frac{2n}{3} \rfloor$ . Hence,  $|G_P| \leq |K_P| + |\Pi_P| \leq \lfloor \frac{2n}{3} \rfloor$ , which yields the following theorem.

**Theorem 7** *Let  $P$  be a piecewise convex polygon with  $n \geq 2$  vertices.  $P$  can be guarded with  $\lfloor \frac{2n}{3} \rfloor$  vertex guards.*

### 3.4 Time and space complexity

In this section we will show how to compute a guarding set  $G_P$ , of size  $\lfloor \frac{2n}{3} \rfloor$ , for  $P$  in  $O(n \log n)$  time and  $O(n)$  space. The algorithm presented at the beginning of this section consists of four phases:

1. The computation of the polygonal approximation  $\tilde{P}$  of  $P$ .
2. The computation of the constrained triangulation  $\mathcal{T}(\tilde{P})$  of  $\tilde{P}$ .
3. The computation of a guarding set  $G_{\tilde{P}}$  for  $\tilde{P}$ .
4. The computation of a guarding set  $G_P$  for  $P$  from the guarding set  $G_{\tilde{P}}$ .

Step 2 of the algorithm presented above can be done in  $O(n)$  time and space. We need linear time and space to create the constrained triangles of  $\mathcal{T}(\tilde{P})$  and linear total time and space to triangulate the subpolygons created after the introduction of the constrained triangles.

Step 3 of the algorithm takes also linear time and space with respect to the size of the polygon  $\tilde{P}$ . By Corollary 4  $|\tilde{P}| \leq 3n$ , which implies that the guarding set  $G_{\tilde{P}}$  can be computed in  $O(n)$  time and space.

Step 4 also requires  $O(n)$  time. Computing  $G_P$  from  $G_{\tilde{P}}$  requires determining for each vertex  $v$  of  $K_{\tilde{P}} \setminus K_P$  all the vertices of  $\Pi_P$  adjacent to it. This takes time proportional to the degree  $\deg(v)$  of  $v$  in  $\mathcal{T}(\tilde{P})$ , i.e., a total of  $\sum_{v \in K_{\tilde{P}} \setminus K_P} \deg(v) = O(|\tilde{P}|) = O(n)$  time. Clearly, the space requirements for performing Step 4 is  $O(n)$ .

To complete our time and space complexity analysis, we need to show how to compute the polygonal approximation  $\tilde{P}$  of  $P$  in  $O(n \log n)$  time and linear space. Before proceeding with the analysis let us recall some definitions from Subsection 3.1. Given a convex arc  $a_i$  of  $P$ , with endpoints the vertices  $v_i v_{i+1}$ , the convex region delimited by the arc  $a_i$  and the line segment  $v_i v_{i+1}$  is called the room  $r_i$  associated with  $a_i$ . We denote by  $R_i$  the vertices of  $P$  that are contained in the interior of  $r_i$  or on the line segment  $v_i v_{i+1}$ . The set of vertices on the convex hull of the set  $R_i \cup \{v_i, v_{i+1}\}$  is denoted by  $C_i$ , and, finally,  $C_i^* = C_i \setminus \{v_i, v_{i+1}\}$ .

In order to compute the polygonal approximation  $\tilde{P}$  of  $P$ , it suffices to determine for each room  $r_i$  of  $P$ , whether it is an empty room or a non-empty room. In other words we need to compute for each room  $r_i$  the set of vertices  $C_i^*$ . If  $C_i^* = \emptyset$ , then  $r_i$  is empty, otherwise we have the set of vertices we wanted. From  $C_i^*$  we can compute the points  $w_{i,k}$  and the linear polygon  $\tilde{P}$  in  $O(n)$  time and space.

The underlying idea is to split  $P$  into  $y$ -monotone piecewise convex subpolygons. For each room  $r_i$  within each such  $y$ -monotone subpolygon, corresponding to a convex arc  $a_i$  with endpoints  $v_i$  and  $v_{i+1}$ , we will then compute the corresponding set  $C_i^*$ . This will be done by first computing a subset  $S_i$  of the set  $R_i$  of the points inside the room  $r_i$ , such that  $S_i \supseteq C_i^*$ , and then apply an optimal time and space convex hull algorithm to the set  $S_i \cup \{v_i, v_{i+1}\}$  in order to compute  $C_i$ , and subsequently from that  $C_i^*$ . In the discussion that follows, we assume that for each convex arc  $a_i$  of  $P$  we associate a set  $S_i$ , which is initialized to be the empty set. The sets  $S_i$  will be progressively filled with vertices of  $P$ , so that in the end they fulfill the containment property mentioned above.

Splitting  $P$  into  $y$ -monotone piecewise convex subpolygons can be done in two steps:

1. First we need to split each convex arc  $a_i$  into  $y$ -monotone pieces. Let  $P'$  be the piecewise convex polygon we get by introducing the  $y$ -extremal points for each  $a_i$ . Since each  $a_i$  can yield up to three  $y$ -monotone convex pieces, we conclude that  $|P'| \leq 3n$ . Obviously splitting the convex arcs  $a_i$  into  $y$ -monotone pieces takes  $O(n)$  time and space. A vertex added to split a convex arc into  $y$ -monotone pieces will be called an *added external vertex*.
2. Second, we need to apply the standard algorithm for computing  $y$ -monotone subpolygons out of a linear polygon to  $P'$  (cf. [16] or [8]). The algorithm in [16] (or [8]) is valid not only for line segments, but rather for any polygon comprised of  $y$ -monotone edges. Since  $|P'| \leq 3n$ , we conclude that compute the  $y$ -monotone subpolygons of  $P'$  takes  $O(n \log n)$  time and requires  $O(n)$  space.

Note that a non-split arc of  $P$  belongs to exactly one  $y$ -monotone subpolygon.  $y$ -monotone pieces of a split arc of  $P$  may belong to at most three  $y$ -monotone subpolygons (see Fig. 9).

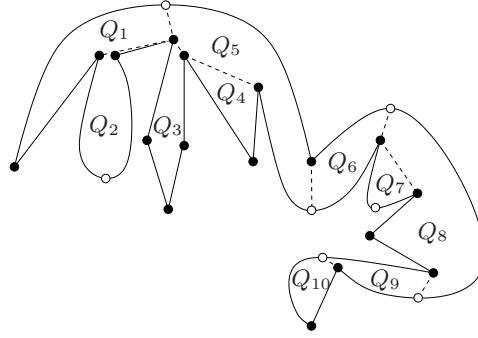


Figure 9: Decomposition of a piecewise convex polygon into ten  $y$ -monotone subpolygons. The white points are added extremal vertices that have been added in order to split non- $y$ -monotone arcs to  $y$ -monotone pieces.

At the beginning of our algorithm we associate to each arc  $a_i$  of  $P$  a set of vertices  $S_i$ , which is initialized to the empty set. Suppose now that we have a  $y$ -monotone polygon  $Q$ . The edges of  $Q$  are either convex arcs of  $P$ , or pieces of convex arcs of  $P$ , or line segments between mutually visible vertices of  $P$ , added in order to form the  $y$ -monotone subpolygons of  $P$ ; we call these line segments *bridges*. For each non-bridge edge  $e_i$  of  $Q$ , we want to compute the set  $C_i^*$ . This can be done by sweeping  $Q$  in the negative  $y$ -direction (i.e., by moving the sweep line from  $+\infty$  to  $-\infty$ ). The events of the sweep correspond to the  $y$  coordinates of the vertices of  $Q$ , which are all known before-hand and can be put in a decreasing sorted list. The first event of the sweep corresponds to the top-most vertex of  $Q$ : since  $Q$  consists of  $y$ -monotone convex arcs, the  $y$ -maximal point of  $Q$  is necessarily a vertex. The last event of the sweep corresponds to the bottom-most vertex of  $Q$ , which is also the  $y$ -minimal point of  $Q$ . We distinguish between four different types of events:

1. the first event: corresponds to the top-most vertex of  $Q$ ,
2. the last event: corresponds to the bottom-most vertex of  $Q$ ,
3. a left event: corresponds to a vertex of the left  $y$ -monotone chain of  $Q$ , and
4. a right event: corresponds to a vertex of the right  $y$ -monotone chain of  $Q$ .

Our sweep algorithm proceeds as follows. Let  $\ell$  be the sweep line parallel to the  $x$ -axis at some  $y$ . For each  $y$  in between the  $y$  maximal and minimal values of  $Q$ ,  $\ell$  intersects  $Q$  at two points which belong to either a left edge  $e_l$  (i.e., an edge on the left  $y$ -monotone chain of  $Q$ ) or is a left vertex  $v_l$  (i.e., a vertex on the left  $y$ -monotone chain of  $Q$ ), and either a right edge  $e_r$  (i.e., an arc on the right  $y$ -monotone chain of  $Q$ ) or a right vertex  $v_r$  (i.e., a vertex on the right  $y$ -monotone chain of  $Q$ ). We are going to associate the current left edge  $e_l$  at position  $y$  to a point set  $S_L$  and the current right edge at position  $y$  to a point set  $S_R$ . If the edge  $e_l$  (resp.  $e_r$ ) is a non-bridge edge, the set  $S_L$

(resp.  $S_R$ ) will contain vertices of  $Q$  that are inside the room of the convex arc of  $P$  corresponding  $e_l$  (resp.  $e_r$ ).

When the  $y$ -maximal vertex  $v_{max}$  is encountered, i.e., during the first event, we initialize  $S_L$  and  $S_R$  to be the empty set. When a left event is encountered due a vertex  $v$ , let  $e_{l,up}$  be the left edge above  $v$  and  $e_{l,down}$  be the left edge below  $v$  and let  $e_r$  be the current right edge (i.e., the right edge at the  $y$ -position of  $v$ ). If  $e_{l,up}$  is a non-bridge edge, and  $a_i$  is the corresponding convex arc of  $P$ , we augment the set  $S_i$  by the vertices in  $S_L$ . Then, irrespectively of whether or not  $e_{l,up}$  is a bridge edge, we re-initialize  $S_L$  to be the empty set. Finally, if  $e_r$  is a non-bridge edge, and  $a_k$  is the corresponding convex arc in  $P$ , we check if  $v$  is inside the room  $r_k$  or lies in the interior of the chord of  $r_k$ ; if this is the case we add  $v$  to  $S_R$ . When a right event is encountered our sweep algorithm behaves symmetrically. If the right event is due to a vertex  $v$ , let  $e_{r,up}$  be right edge of  $Q$  above  $v$  and  $e_{r,down}$  be the right edge of  $Q$  below  $v$  and let  $e_l$  be the current left edge of  $Q$ . If  $e_{l,up}$  is a non-bridge edge, and  $a_i$  is the corresponding convex arc of  $P$ , we augment  $S_i$  by the vertices in  $S_R$ . Then, irrespectively of whether or not  $e_{r,up}$  is a bridge edge or not, we re-initialize  $S_R$  to be the empty set. Finally, If  $e_l$  is a non-bridge edge, and  $a_k$  is the corresponding convex arc of  $P$ , we check if  $v$  is inside the room  $r_k$  or lies in the interior of the chord of  $r_k$ ; if this is the case we add  $v$  to  $S_L$ . When the last event is encountered due to the  $y$ -minimal vertex  $v_{min}$ , let  $e_l$  and  $e_r$  be the left and right edges of  $Q$  above  $v_{min}$ , respectively. If  $e_l$  is a non-bridge edge, let  $a_i$  be the corresponding convex arc in  $P$ . In this case we simply augment  $S_i$  by the vertices in  $S_L$ . Symmetrically, if  $e_r$  is a non-bridge edge, let  $a_j$  be the corresponding convex arc in  $P$ . In this case we simply augment  $S_j$  by the vertices in  $S_R$ .

We claim that our sweep-line algorithm computes a set  $S_i$  such that  $S_i \supseteq C_i^*$ . To prove this we need the following intermediate result:

**Lemma 8** *Given a non-empty convex arc  $a_i$  of  $P$ , the vertices of the set  $C_i^*$  belong to the  $y$ -monotone subpolygons of  $P$ , which either contain the entire arc  $a_i$  or  $y$ -monotone pieces of  $a_i$ .*

*Proof.* Let  $a_i$  be a non-empty convex arc of  $P$  and let  $u$  be a vertex of  $P$  in  $C_i^*$  that is not a vertex of any of  $y$ -monotone subpolygons of  $P$  that contain either the entire arc  $a_i$  or  $y$ -monotone pieces of  $a_i$ . Since  $u$  is inside the room  $r_i$ , the  $x$ -axis parallel line through  $u$  intersects  $a_i$  at either one or two points (respectively, intersects one or two  $y$ -monotone pieces of  $a_i$ ). Let  $u'$  be the unique or one of the two intersection points and consider the corresponding  $y$ -monotone arc  $e$  of  $a_i$  on which  $u'$  lies. Let  $Q$  be the  $y$ -monotone subpolygon of  $P$  that contains  $e$ . We may assume without loss of generality that  $e$  is a right edge in  $Q$ .

Recall that  $u$  is not a vertex of  $Q$  and let  $Q' \neq Q$  be one of the  $y$ -monotone polygons that contain  $u$ . Since we have assumed that  $Q' \neq Q$ , the line segment  $uu'$  either contains a vertex of  $P$  or is intersected by an edge of  $Q'$ .  $uu'$  cannot contain a vertex of  $P$ , since, in such a case,  $u$  could not have been a vertex in  $C_i^*$ . Let  $e'$  be an edge of  $Q'$  that intersects the line segment  $uu'$ . We distinguish between the following two cases:

1.  $e'$  is a bridge edge. In this case  $e'$  is a line segment connecting two vertices  $w$  and  $z$  of  $P'$ . Moreover,  $w$  and  $z$  lie on different sides with respect to the line  $uu'$ . We can assume, without loss of generality that  $w$  lies above  $uu'$  and that  $z$  lies below  $uu'$ .

If both  $w$  and  $z$  are vertices of  $P$ , then  $u$  cannot belong to  $C_i^*$ , since it is contained either in the interior of the triangle  $v_i w v_{i+1}$ , or in the interior of the triangle  $v_i z v_{i+1}$  or in the convex quadrilateral formed by the four vertices  $v_i, v_{i+1}, w$  and  $z$ . Thus, we reach a contradiction.

Consider now the case where  $w \in P' \setminus P$  and  $z \in P'$ . Since  $w$  is an added external vertex, it has to be a  $y$ -maximal vertex ( $y$ -minimal vertices cannot be endpoints of a bridge edge with lower  $y$ -coordinate than the other endpoint of the bridge edge). Consider the triangle  $uwz$ . The edge  $uw$  cannot contain any vertices and cannot intersect any edges of  $P$ , since then  $u$  would not belong to  $C_i^*$ . Hence the points  $u$  and  $w$  are mutually visible. But the  $y$ -coordinate of  $u$  is larger than the  $y$  coordinate of  $z$ , which implies that  $z$  would not have been connected to  $w$  via a bridge edge in the  $y$ -monotone decomposition of  $P$ , a contradiction.

There one final case to consider:  $w \in P$  and  $z \in P' \setminus P$ . Since  $z$  is an added external vertex, it has to be a  $y$ -minimal vertex ( $y$ -minimal vertices cannot be endpoints of a bridge edge with higher  $y$ -coordinate than the other endpoint of the bridge edge). Again, consider the triangle  $uwz$ . The edge  $uz$  cannot contain any vertices and cannot intersect any edges of  $P$ , since then  $u$  would not belong to  $C_i^*$ . Hence the points  $u$  and  $z$  are mutually visible. However, the  $y$ -coordinate of  $u$  is smaller than the  $y$ -coordinate of  $w$ , which implies that  $w$  would not have been connected to  $z$  via a bridge edge in the  $y$ -monotone decomposition of  $P$ ; again, a contradiction.

2.  $e'$  is not a bridge edge. Among all edges of  $P$  that intersect  $uu'$ , consider the one “closest” to  $u'$ ; the distance here from  $u'$  is understood as the distance of the point of intersection of  $e'$  and  $uu'$  from  $u'$ . Let  $e'$  be this closest edge and let  $w$  and  $z$  be its endpoints in  $P'$ . Let  $p$  be the point of intersection of  $e'$  and  $uu'$ , and let  $t_p$  be the tangent line of  $e'$  at  $p$ . Clearly,  $t_p$  cannot coincide with the line  $uu'$ , since then  $P$  would not be locally convex at  $p$ . Hence  $t_p$  separates  $u$  and  $u'$ , i.e., they have to lie on different halfspaces with respect to  $t_p$ . Let  $H_u$  and  $H_{u'}$  be the two halfspaces defined by  $t_p$  that contain  $u$  and  $u'$ , respectively. Clearly,  $e'$  cannot lie on  $H_u$ , since then  $P$  would not be locally convex at  $p$ . Therefore,  $e'$  has to belong entirely to  $H_{u'}$ , which further implies that both  $w$  and  $z$  also have to belong to  $H_{u'}$ . Since  $e'$  is either a convex arc  $a_j$  of  $P$  or a  $y$ -monotone piece of a convex arc  $a_j$  of  $P$ , we conclude that the endpoints  $v_j$  and  $v_{j+1}$  of  $a_j$  also have to belong to the halfspace  $H_{u'}$ . Notice that  $v_j$  and  $v_{j+1}$  cannot both lie on the same side of the line  $uu'$ , since such a configuration would contradict the fact that  $p$  is the closest point on  $uu'$  to  $u'$ . Hence  $v_j$  and  $v_{j+1}$  lie on different sides of the line  $uu'$ . This, in turn, yields that  $u$  lies either in the interior of the triangle  $v_i v_j v_{i+1}$ , or in the interior of the triangle  $v_i v_{j+1} v_{i+1}$ , or in the interior of the convex quadrilateral defined by the four points  $v_i, v_{i+1}, v_j$  and  $v_{j+1}$ . This contradicts the fact that  $u$  belongs to  $C_i^*$ .  $\square$

An immediate corollary of the above lemma is the following:

**Corollary 9** *For each convex arc  $a_i$  of  $P$ , the set  $S_i$  computed by the sweep algorithm described above is a superset of the set  $C_i^*$ .*

Let us now analyze the time and space complexity of Step 1 of the algorithm sketched at the beginning of this subsection. Computing the polygonal approximation  $\tilde{P}$  of  $P$  requires subdividing



$P$  into  $y$ -monotone subpolygons. This subdivision takes  $O(n \log n)$  time and  $O(n)$  space. Once we have the subdivision of  $P$  into  $y$ -monotone subpolygons we need to compute the sets  $S_i$  for each convex arc  $a_i$  of  $P$ . The sets  $S_i$  can be implemented as red-black trees. Inserting an element in some  $S_i$  takes  $O(\log n)$  time. During the course of our algorithm we perform only insertions on the  $S_i$ 's. A vertex  $v$  of  $P$  is inserted at most  $\deg(v)$  times in some  $S_i$ , where  $\deg(v)$  is the degree of  $v$  in the  $y$ -monotone decomposition of  $P$ . Since the sum of the degrees of the vertices of  $P$  in the  $y$ -monotone decomposition of  $P$  is  $O(n)$ , we conclude that the total size of the  $S_i$ 's is  $O(n)$  and that we perform  $O(n)$  insertions on the  $S_i$ 's. Therefore we need  $O(n \log n)$  time and  $O(n)$  space to compute the  $S_i$ 's. Finally, since  $\sum_{i=1}^n |S_i| = O(n)$ , the sets  $C_i^*$  can also be computed in total  $O(n \log n)$  time and  $O(n)$  space. The analysis above thus yields the following:

**Theorem 10** *Let  $P$  be a piecewise convex polygon with  $n$  vertices. We can compute a guarding set for  $P$  of size  $\lfloor \frac{2n}{3} \rfloor$  in  $O(n \log n)$  time and  $O(n)$  space.*

### 3.5 The lower bound construction

In this section we are going to present a piecewise convex polygon which requires a minimum of  $\frac{4n}{7}$  vertex guards in order to be guarded.

Let us first consider the windmill-like piecewise convex polygon  $W$  with seven vertices of Fig. 10(a), a detail of which is shown in Fig. 10(b). The *double ear* defined by the vertices  $v_3$ ,  $v_4$  and  $v_5$  and the convex arcs  $a_3$  and  $a_4$  is constructed in such a way so that none of the three vertices  $v_3$ ,  $v_4$  and  $v_5$  can guard both rooms  $r_3$  and  $r_4$  by itself. This is achieved by ensuring that  $a_3$  (resp.  $a_4$ ) intersects the line  $v_4v_5$  (resp.  $v_3v_4$ ) twice. Note that both  $a_3$  and  $a_4$  intersect the line  $mv_4$  only at  $v_4$ , where  $m$  is the midpoint of the line segment  $v_3v_5$ . The double ear defined by the vertices  $v_5$ ,  $v_6$  and  $v_7$  and the convex arcs  $a_5$  and  $a_6$  is constructed in an analogous way. Moreover, the vertices  $v_1$ ,  $v_2$ ,  $v_4$  and  $v_6$  are placed in such a way so that they do not (collectively) guard the interior of the triangle  $v_3v_5v_7$  (for example the lengths of the edges  $v_1v_7$  and  $v_2v_3$  are considered to be big enough, so that  $v_2$  does not see too much of the triangle  $v_3v_5v_7$ ). As a result of this construction  $W$  cannot be guarded by two vertex guards, but can be guarded with three. There are actually only five possible guarding triplets:  $\{v_3, v_4, v_6\}$ ,  $\{v_3, v_5, v_6\}$ ,  $\{v_3, v_5, v_7\}$ ,  $\{v_4, v_5, v_7\}$  and  $\{v_4, v_6, v_7\}$ . Any guarding set that contains either  $v_1$  or  $v_2$  has cardinality at least four. The vertices  $v_1$  and  $v_2$  will be referred to as *base vertices*.

Consider now the crescent-like polygon  $C$  with  $n$  vertices of Fig. 11. The vertices of  $C$  are in strictly convex position. This fact has the following implication: if  $v_i$ ,  $v_{i+1}$ ,  $v_{i+2}$  and  $v_{i+3}$  are four consecutive vertices of  $C$ , and  $u$  is the point of intersection of the lines  $v_iv_{i+1}$  and  $v_{i+2}v_{i+3}$ , then the triangle  $v_{i+1}uv_{i+2}$  is guarded if and only if either  $v_{i+1}$  or  $v_{i+2}$  is in the guarding set of  $C$ . As a result, it is easy to see that  $C$  cannot be guarded with less than  $\lfloor \frac{n}{2} \rfloor$  vertices, since in this case there will be at least one edge both endpoints of which would not be in the guarding set for  $C$ .

In order to construct the piecewise convex polygon, that gives us the lower bound mentioned at the beginning of this section, we are going to merge several copies of  $W$  with  $C$ . More precisely, consider the piecewise convex polygon  $P$  with  $n = 7k$  vertices of Fig. 12. It consists of copies of the polygon  $W$  merged with  $C$  at every other linear edge of  $C$ , through the base points of the  $W$ 's.



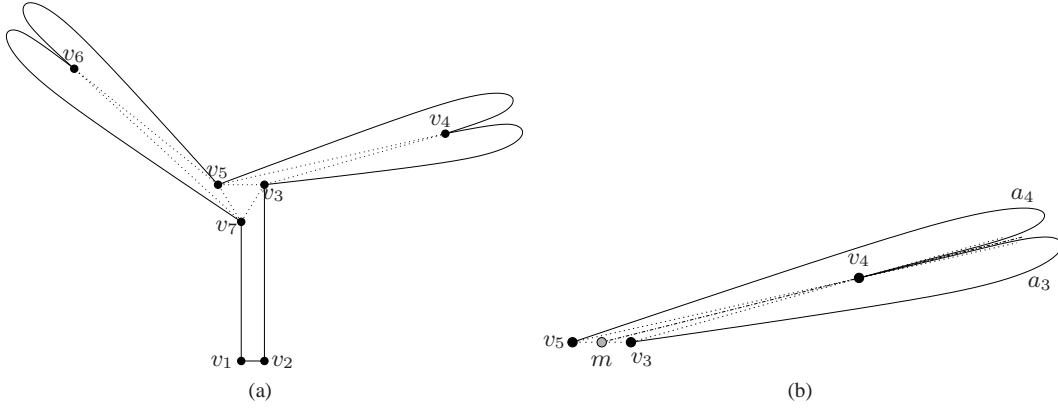


Figure 10: The windmill-like piecewise convex polygon  $W$  that requires at least three vertex guards in order to be guarded. The only triplets of guards that guard  $W$  are  $\{v_3, v_4, v_6\}$ ,  $\{v_3, v_5, v_7\}$  and  $\{v_4, v_5, v_7\}$ .

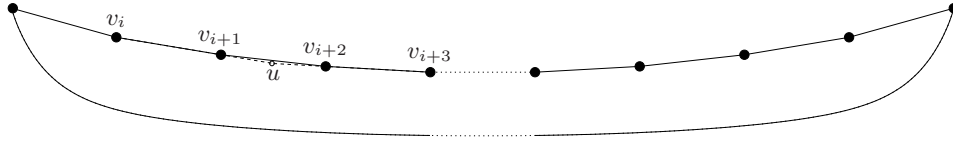


Figure 11: The crescent-like piecewise convex polygon  $C$ , that requires a guarding set of at least  $\lfloor \frac{n}{2} \rfloor$  vertex guards.

In order to guard any of the windmill-like subpolygons, we need at least three vertices per such polygon, none of which can be a base point. This gives a total of  $3k$  vertices. On the other hand, in order to guard the crescent-like part of  $P$  we need at least  $k - 1$  guards among the base points. To see that, notice that there are  $k - 1$  linear segments connecting base points; if we were to use less than  $k - 1$  guards, we would have at least one such line segment  $e$ , both endpoints of which would not participate in the guarding set of  $G$ . But then, as in the case of  $C$ , there would be a triangle, adjacent to  $e$ , which would not be guarded. Therefore, in order to guard  $P$  we need a minimum of  $4k - 1$  guards, which yields the following theorem.

**Theorem 11** *There exists a family of piecewise convex polygons with  $n$  vertices any vertex guarding set of which has cardinality at least  $\lfloor \frac{4n}{7} \rfloor - 1$ .*

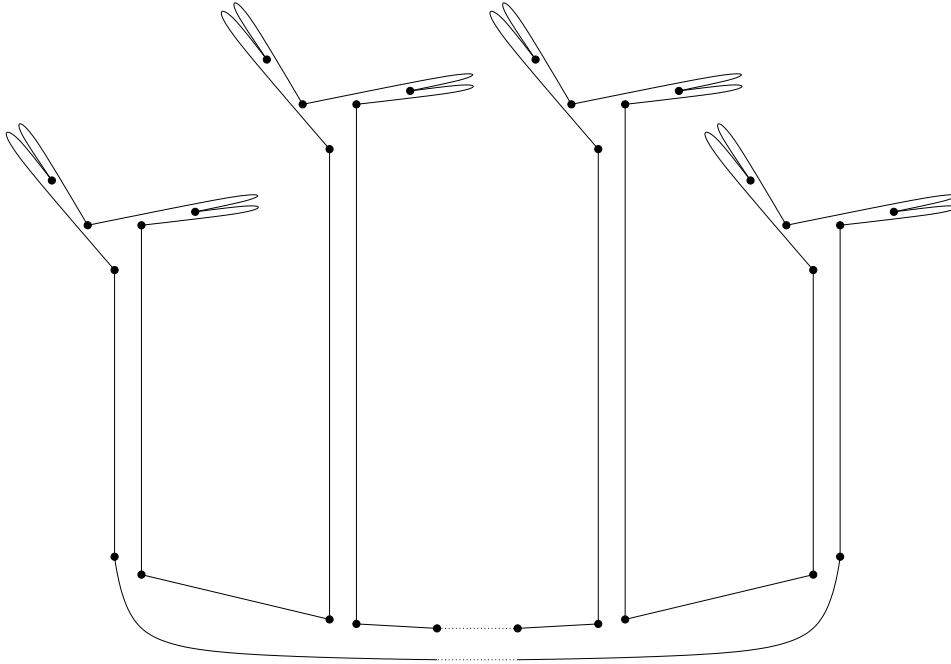


Figure 12: The lower bound construction.

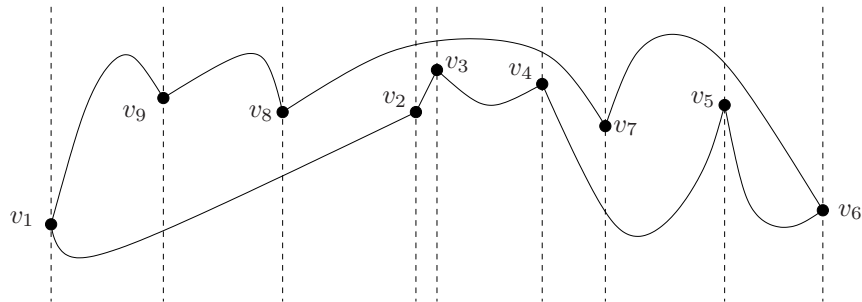


Figure 13: A monotone piecewise convex polygon  $P$  with  $n = 9$  vertices and its vertical decomposition into four-sided convex slabs.  $P$  can be guarded with  $\lfloor \frac{n}{2} \rfloor = 4$  vertices, namely the vertices  $v_9$ ,  $v_2$ ,  $v_4$  and  $v_5$ .

## 4 Monotone piecewise convex polygons

In this section we focus on the subclass of piecewise convex polygons that are monotone. Let us recall the definition of monotone polygons from Section 1. A curved polygon  $P$  is called monotone if there exists a line  $L$  such that any line  $L^\perp$  perpendicular to  $L$  intersects  $P$  at most twice.

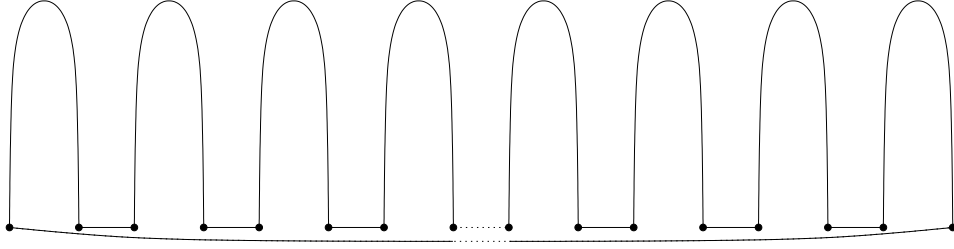


Figure 14: A comb-like monotone piecewise convex polygon that requires  $\lfloor \frac{n}{2} \rfloor$  point guards in order to be guarded: one point guard is required per prong.

In the case of linear polygons monotonicity does not yield better bounds on the worst case number of point or vertex guards needed in order to guard the polygon. In both cases, monotone or possibly non-monotone linear polygons,  $\lfloor \frac{n}{3} \rfloor$  point or vertex guards are always sufficient and sometimes necessary. In the context of piecewise convex polygons the situation is different. Unlike general (i.e., not necessarily monotone) piecewise convex polygons, which require at least  $\lfloor \frac{4n}{7} \rfloor - 1$  vertex guards and can always be guarded with  $\lfloor \frac{2n}{3} \rfloor$  vertex guards, monotone piecewise convex polygons can always be guarded with  $\lfloor \frac{n}{2} \rfloor$  vertex guards. This bound is actually tight, since there exist monotone piecewise convex polygons that require that many vertex or point guards. This section is devoted to the presentation of the afore-mentioned tight bounds.

Let us consider a monotone piecewise convex polygon  $P$ , and let us assume without loss of generality that  $P$  is monotone with respect to the  $x$ -axis (see also Fig. 13). Let  $u_j$  be the  $j$ -th vertex of  $P$  when considered in the list of vertices sorted with respect to their  $x$ -coordinate. Let  $\ell_j$  be the vertical line passing through the vertex  $u_j$  of  $P$ , and let  $\mathcal{L} = \{\ell_1, \ell_2, \dots, \ell_n\}$  be the collection of these lines. An immediate consequence of the fact that  $P$  is monotone and piecewise convex is the following lemma:

**Lemma 12** *The collection of lines  $\mathcal{L}$  decomposes the interior of  $P$  into  $n - 1$  convex regions  $\kappa_j$ ,  $j = 1, \dots, n - 1$ , that are free of vertices or edges of  $P$ .*

In addition to the fact that the regions  $\kappa_j$  are convex,  $\kappa_j$  has on its boundary both  $u_j$  and  $u_{j+1}$ . This immediately implies that both  $u_j$  and  $u_{j+1}$  see the entire region  $\kappa_j$ . Hence, in order to guard  $P$  it suffices to take every other vertex  $u_j$ , starting from  $u_2$ . The set  $G = \{u_2, u_4, u_6, \dots, u_{2\lfloor n/2 \rfloor}\}$  is, thus, a vertex guarding set for  $P$ . As a final remark, if the line  $L$  with respect to which  $P$  is monotone is known, it is possible to compute the vertex guarding set  $G$  for  $P$  in  $O(n \log n)$  time and  $O(n)$  space: sort vertices of  $P$  with respect to their projections on  $L$ , and then report every other vertex in the sorted list, starting from the second one.

The polygon  $M$  yielding the lower bound is shown in Fig. 14. Notice that is very similar to the well known comb-like linear polygon that establishes the lower bound on the number of point or vertex guards required to guard a linear polygon. In our case it is easy to see that we need at least

one point or vertex guard per prong of the polygon, and since there are  $\lfloor \frac{n}{2} \rfloor$  prongs we conclude that we need at least  $\lfloor \frac{n}{2} \rfloor$  vertex or point guards in order to guard  $M$ .

We are now ready to state the following theorem that summarizes the results of this section.

**Theorem 13** *Given a monotone piecewise convex polygon  $P$  with  $n \geq 2$  vertices,  $\lfloor \frac{n}{2} \rfloor$  vertex or point guards are always sufficient and sometimes necessary. Moreover, if the line  $L$  with respect to which  $P$  is monotone is known, we can compute a vertex guarding set for  $P$  of size  $\lfloor \frac{n}{2} \rfloor$  in  $O(n \log n)$  time and  $O(n)$  space.*

## 5 Piecewise concave polygons

In this section we deal with the problem of guarding piecewise concave polygons using point guards. Guarding a piecewise concave polygon with vertex guards may be impossible even for very simple configurations (see Fig. 15(a)). In particular we prove the following:

**Theorem 14** *Let  $P$  be a piecewise concave polygon with  $n$  vertices.  $2n - 4$  point guards are always sufficient and sometimes necessary in order to guard  $P$ .*

To prove the sufficiency of  $2n - 4$  point guards we essentially apply the technique in [11] for illuminating disjoint compact convex sets. Let  $A_i$  be the convex object defined by the edge  $a_i$  and the chord  $v_i v_{i+1}$  of  $p_i$ . Let  $t_i(v_j)$  be the tangent line to  $a_i$  at  $v_j$ ,  $j = i, i + 1$ , and let  $b_{i+1}$  be the bisecting ray of  $t_i(v_{i+1})$ ,  $t_{i+1}(v_{i+1})$  pointing towards the interior of  $P$ . Construct a set of convex arcs  $\mathcal{K} = \{\kappa_1, \kappa_2, \dots, \kappa_n\}$  that lie entirely inside  $P$  as such that (cf. [11]): (a) the endpoints of  $\kappa_i$  are  $v_i, v_{i+1}$ , (b)  $\kappa_i$  is tangent to  $b_i$  (resp.  $b_{i+1}$ ) at  $v_i$  (resp.  $v_{i+1}$ ), (c) if  $S_i$  is the convex object defined by  $\kappa_i$  and its chord  $v_i v_{i+1}$ , then  $S_i \subseteq A_i$ ,  $1 \leq i \leq n$ , (d) the arcs  $\kappa_i$  are pairwise non-crossing, and (e) the number of tangencies between the elements of  $\mathcal{K}$  is maximized. Let  $Q$  be the piecewise concave polygon defined by the sequence of the arcs in  $\mathcal{K}$ .

Suppose now that  $\kappa_i$  and  $\kappa_{\sigma(j)}$  are tangent,  $1 \leq j \leq m$ , and let  $\ell_{i,\sigma(j)}$  be the common tangent to  $\kappa_i$  and  $\kappa_{\sigma(j)}$ . Let  $\Pi_i$  be the polygonal region defined by the intersection of the half-planes bounded by the  $\ell_{i,\sigma(j)}$ 's that contain  $\kappa_i$  and  $a_i$ .  $\Pi_i$  is a bounded polygonal region one edge of which is the chord of  $a_i$ . It is easy to see that placing guards on the vertices of the  $\Pi_i$ 's guards both  $P$  and  $Q$ . Let  $G_Q$  be the guarding set of  $P$  constructed this way. Construct, now, a planar graph  $\Gamma$  with vertex set  $\mathcal{K}$ . Two vertices  $\kappa_i$  and  $\kappa_j$  of  $\Gamma$  are connected via an edge if  $\kappa_i$  and  $\kappa_j$  are tangent. The graph  $\Gamma$  is a planar graph combinatorially equivalent to the triangulation graph  $\mathcal{T}(R)$  of a polygon  $R$  with  $n$  vertices. The edges of  $\Gamma$  connecting the polygons  $\kappa_i, \kappa_{i+1}$ ,  $1 \leq i \leq n$ , are the edges of  $R$ , whereas all other edges of  $\Gamma$  correspond to diagonals in  $\mathcal{T}(R)$ . Let  $Q^\circ$  denote the interior of  $Q$ . Observing that  $Q^\circ$  consists of a number of faces that are in 1–1 correspondence with the triangles in  $\mathcal{T}(R)$ , we conclude that  $Q^\circ$  consists of  $n - 2$  faces, each containing three guards of  $G_Q$ . In fact, each face of  $Q^\circ$  can actually be guarded by only two of the three guards it contains and thus we can eliminate one of them per face of  $Q^\circ$ . The new guarding set  $G$  of  $Q$  constructed above is also a guarding set for  $P$  and contains  $2(n - 2)$  point guards.

To prove the necessity, refer to the piecewise concave polygon  $P$  in Fig. 15(b). Each one of the “triangular” regions in the interior of  $P$  requires exactly two point guards in order to be guarded.

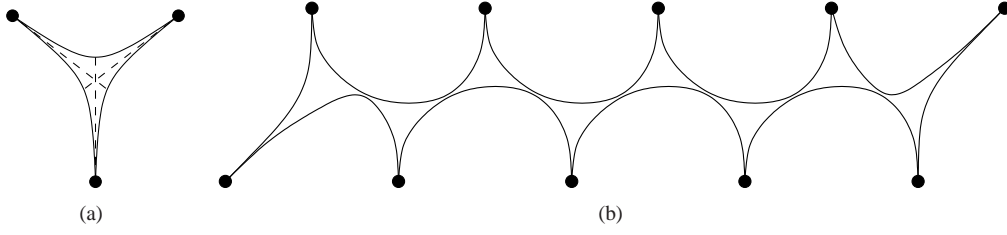


Figure 15: (a) A piecewise concave polygon  $P$  that cannot be guarded solely by vertex guards. Two consecutive edges of  $P$  have a common tangent at the common vertex and as a result the three vertices of  $P$  see only the points along the dashed segments. (b) A piecewise concave polygon  $P$  that requires  $2n - 4$  point guards in order to be guarded.

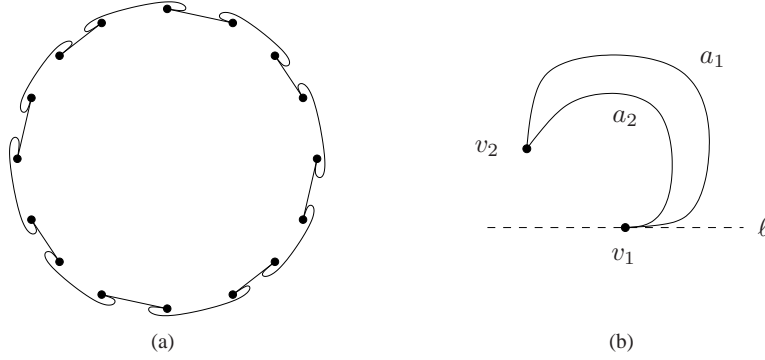


Figure 16: (a) A locally convex polygon with  $n$  vertices that requires  $n$  vertex or point guards in order to be guarded. (b) A non-convex polygon that cannot be guarded by vertex guards, and which requires an infinite number of point guards.

The number of such triangular regions is exactly  $n - 2$ , thus we need a total of  $2n - 4$  point guards to guard  $P$ .

It is interesting to note that the piecewise concave polygon  $P$  in Fig. 15(b) is monotone with respect to the  $x$ -axis. This implies that, unlike the class of piecewise convex polygons, monotonicity does not induce better bounds on the number of guards required to guard piecewise concave polygons.

## 6 Other lower bounds

We have so far been dealing with the case of piecewise convex polygons. In this section we will present some initial results about locally convex and non-convex polygons.

For locally convex polygons the situation is much less interesting, in the sense that there exist locally convex polygons that require  $n$  vertex guards in order to be guarded. Consider for example the locally convex polygon of Fig. 16(a). Every room in this polygon cannot be guarded by a single guard, but rather it requires both vertices of every locally convex edge to be in any guarding set in order for the corresponding room to be guarded. As a result it requires  $n$  vertex guards. Clearly, these  $n$  guards are also sufficient, since any one of them guards also the central convex part of the polygon. More interestingly, even if we do not restrict ourselves to vertex guards, but rather allow guards to be any point in the interior or the boundary of the polygon, then the locally convex polygon in Fig. 16(a) still requires  $n$  guards. This stems from the fact that the rooms of this polygon have been constructed in such a way so that the kernel of each room is the empty set (i.e., they are not star-shaped objects). However, we can guard each room with two guards, which can actually be chosen to be the endpoints of the locally convex arcs.

The case of non-convex polygons poses even more difficulties. Consider the non-convex polygon  $N$  of Fig. 16(b), which consists of two vertices  $v_1$  and  $v_2$  and two convex arcs  $a_1$  and  $a_2$ . The two arcs are tangent to a common line  $\ell$  at  $v_1$ . It is readily visible that  $v_1$  and  $v_2$  cannot guard the interior of  $N$ . In fact,  $v_1$  cannot guard any point of  $N$  other than itself. Even worse, any finite number of guards, placed anywhere in  $N$ , cannot guard the polygon. To see that, consider the vicinity of  $v_1$ . Assume that  $N$  can be guarded by a finite number of guards, and let  $g \neq v_1$  be the guard closest to  $v_1$  with respect to shortest paths within  $N$ . Consider the line  $\ell_g$  passing through  $g$  that is tangent to  $a_2$  (among the two possible tangents we are interested in the one the point of tangency of which is closer to  $v_1$ ). Consider now the sector  $s_g$  of  $N$  delimited by  $a_1$ ,  $a_2$  and  $\ell_g$ .  $s_g$  cannot contain any guarding point, since such a vertex would be closer to  $v_1$  than  $g$ . Since  $s_g$  is not guarded by  $v_1$ , we conclude that  $s_g$  is not guarded at all, which contradicts our assumption that  $N$  is guarded by a finite set of guards.

## 7 Summary and future work

In this paper we have considered the problem of guarding a polygonal art gallery, the walls of which are allowed to be arcs of curves. We have demonstrated that if we allow these arcs to be locally convex arcs, we need  $n$  guards in the worst case. If these arcs are allowed to be non-convex, then an infinite number of guards may be required. In the case of piecewise convex curved polygons with  $n$  vertices, we have shown that it is always possible to guard the polygon with  $\lfloor \frac{2n}{3} \rfloor$  vertex guards and we have provided an  $O(n \log n)$  time and  $O(n)$  space algorithm for computing a guarding set. Finally, we have described a class of piecewise convex polygons with  $n$  vertices that requires a minimum of  $\lfloor \frac{4n}{7} \rfloor + O(1)$  vertex guards.

Up to now we have not found a piecewise convex polygon that requires more than  $\lfloor \frac{4n}{7} \rfloor + O(1)$ , nor have we devised a polynomial time algorithm for guarding a piecewise convex polygon with less than  $\lfloor \frac{2n}{3} \rfloor$  guards. Closing the gap between then two complexities remains an open problem.

Another open problem is what is the worst case maximum number of point guards required to guard a piecewise convex polygon. In this case our lower bound construction fails, since it is possible to guard the corresponding polygon with  $\lfloor \frac{3n}{7} \rfloor + O(1)$  guards. On the other hand, the comb-like polygon shown in Fig. 14, requires  $\lfloor \frac{n}{2} \rfloor$  points guards. Clearly, our algorithm that computes a guarding set of at most  $\lfloor \frac{2n}{3} \rfloor$  vertex guards is still applicable.

Other types of guarding problems have been studied in the literature, which either differ on the type of guards (e.g., edge or mobile guards), the topology of the polygons considered (e.g., polygons with holes) or the guarding model (e.g., guard the exterior of the polygon – also known as the *fortress problem*, or both the interior and exterior – also known as the *prison yard problem*); see the book by O’Rourke [22], the survey paper by Shermer [24] of the book chapter by Urrutia [27] for an extensive list of the variations of the art gallery problem with respect to the guard types of the guarding model. It would be interesting to extend these results to the families of curved polygons presented in this paper.

Last but not least, in the case of non-convex polygons, is it possible to devise an algorithm for computing a guarding set with a finite number of guards, if the polygon does not contain cusp-like configurations such as the one in Fig. 16(b)?

## Acknowledgements

The authors wish to thank Ioannis Z. Emiris and Hazel Everett for useful discussions about the problem. Work partially supported by the IST Programme of the EU (FET Open) Project under Contract No IST-006413 – (ACS - Algorithms for Complex Shapes with Certified Numerics and Topology).

## References

- [1] A. Aggarwal. *The art gallery problem: Its variations, applications, and algorithmic aspects*. PhD thesis, Dept. of Comput. Sci., Johns Hopkins University, Baltimore, MD, 1984.
- [2] D. Avis and H. ElGindy. A combinatorial approach to polygon similarity. *IEEE Trans. Inform. Theory*, IT-2:148–150, 1983.
- [3] W. Bronsvoort. Boundary evaluation and direct display of CSG models. *Computer-Aided Design*, 20:416–419, 1988.
- [4] B. Chazelle. Triangulating a simple polygon in linear time. In *Proc. 31st Annu. IEEE Sympos. Found. Comput. Sci.*, pages 220–230, 1990.
- [5] B. Chazelle. Triangulating a simple polygon in linear time. *Discrete Comput. Geom.*, 6(5):485–524, 1991.
- [6] B. Chazelle and J. Incerpi. Triangulation and shape-complexity. *ACM Trans. Graph.*, 3(2):135–152, 1984.

- [7] V. Chvátal. A combinatorial theorem in plane geometry. *J. Combin. Theory Ser. B*, 18:39–41, 1975.
- [8] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, Berlin, Germany, 2nd edition, 2000.
- [9] H. Edelsbrunner, J. O’Rourke, and E. Welzl. Stationing guards in rectilinear art galleries. *Comput. Vision Graph. Image Process.*, 27:167–176, 1984.
- [10] K. Eo and C. Kyung. Hybrid shadow testing scheme for ray tracing. *Computer-Aided Design*, 21:38–48, 1989.
- [11] L. Fejes Tóth. Illumination of convex discs. *Acta Math. Acad. Sci. Hungar.*, 29(3–4):355–360, 1977.
- [12] S. Fisk. A short proof of Chvátal’s watchman theorem. *J. Combin. Theory Ser. B*, 24:374, 1978.
- [13] J. Kahn, M. M. Klawe, and D. Kleitman. Traditional galleries require fewer watchmen. *SIAM J. Algebraic Discrete Methods*, 4:194–206, 1983.
- [14] R. Kuc and M. Siegel. Efficient representation of reflecting structures for a sonar navigation model. In *Proc. 1987 IEEE Int. Conf. Robotics and Automation*, pages 1916–1923, 1987.
- [15] D. Lee and A. Lin. Computational complexity of art gallery problems. *IEEE Trans. Inform. Theory*, 32(2):276–282, 1986.
- [16] D. T. Lee and F. P. Preparata. Location of a point in a planar subdivision and its applications. *SIAM J. Comput.*, 6(3):594–606, 1977.
- [17] T. Lozano-Pérez and M. A. Wesley. An algorithm for planning collision-free paths among polyhedral obstacles. *Commun. ACM*, 22(10):560–570, 1979.
- [18] A. Lubiw. Decomposing polygonal regions into convex quadrilaterals. In *Proc. 1st Annu. ACM Sympos. Comput. Geom.*, pages 97–106, 1985.
- [19] M. McKenna. Worst-case optimal hidden-surface removal. *ACM Trans. Graph.*, 6:19–28, 1987.
- [20] G. Meisters. Polygons have ears. *Amer. Math. Monthly*, 82:648–651, 1975.
- [21] J. S. B. Mitchell. An algorithmic approach to some problems in terrain navigation. In D. Kapur and J. Mundy, editors, *Geometric Reasoning*. MIT Press, Cambridge, MA, 1989.
- [22] J. O’Rourke. *Art Gallery Theorems and Algorithms*. The International Series of Monographs on Computer Science. Oxford University Press, New York, NY, 1987.
- [23] J.-R. Sack. *Rectilinear computational geometry*. Ph.D. thesis, School Comput. Sci., Carleton Univ., Ottawa, ON, 1984. Report SCS-TR-54.



- [24] T. C. Shermer. Recent results in art galleries. *Proc. IEEE*, 80(9):1384–1399, Sept. 1992.
- [25] J. Stenstrom and C. Connolly. Building wire frames for multiple range views. In *Proc. 1986 IEEE Conf. Robotics and Automation*, pages 615–650, 1986.
- [26] G. T. Toussaint. Pattern recognition and geometrical complexity. In *Proc. 5th IEEE Internat. Conf. Pattern Recogn.*, pages 1324–1347, 1980.
- [27] J. Urrutia. Art gallery and illumination problems. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 973–1027. North-Holland, 2000.
- [28] S. Xie, T. Calvert, and B. Bhattacharya. Planning views for the incremental construction of body models. In *Proc. Int. Conf. Pattern Recongition*, pages 154–157, 1986.
- [29] M. Yachida. 3-D data acquisition by multiple views. In *Robotics Research: Third Int. Symp.*, pages 11–18, 1986.



---

Unité de recherche INRIA Lorraine  
LORIA, Technopôle de Nancy-Brabois - Campus scientifique  
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes  
4, rue Jacques Monod - 91893 ORSAY Cedex (France)

Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

---

Éditeur  
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)  
<http://www.inria.fr>  
ISSN 0249-6399