Guarding curvilinear art galleries with vertex or point guards

Menelaos I. Karavelas[*,a,b], Csaba D. Tóth[c], Elias P. Tsigaridas[d]

[a] *University of Crete, Department of Applied Mathematics, GR-714 09 Heraklion, Greece*
[b] *Foundation for Research and Technology - Hellas, Institute of Applied and Computational Mathematics, P.O. Box 1385, GR-711 10 Heraklion, Greece*
[c] *University of Calgary, Department of Mathematics and Statistics, 2500 University Drive NW, Calgary, AB, Canada T2N 1N4*
[d] *INRIA-LORIA Lorraine, 615 rue du Jardin Botanique, BP 101, 54602 Villers-dés-Nancy cedex, France*

Abstract

We study a variant of the classical art gallery problem, where an art gallery is modeled by a polygon with curvilinear sides. We focus on piecewise-convex and piecewise-concave polygons, which are polygons whose sides are convex and concave arcs, respectively. It is shown that for monitoring a piecewise-convex polygon with $n \geq 2$ vertices, $\lfloor \frac{2n}{3} \rfloor$ vertex guards are always sufficient and sometimes necessary. We also present an algorithm for computing at most $\lfloor \frac{2n}{3} \rfloor$ vertex guards in $O(n \log n)$ time and $O(n)$ space. For the number of point guards that can be stationed at any point in the polygon, our upper bound $\lfloor \frac{2n}{3} \rfloor$ carries over and we prove a lower bound of $\lceil \frac{n}{2} \rceil$. For monitoring a piecewise-concave polygon with $n \geq 3$ vertices, $2n - 4$ point guards are always sufficient and sometimes necessary, whereas there are piecewise-concave polygons where some points in the interior are hidden from all vertices, hence they cannot be monitored by vertex guards. We conclude with bounds for some special types of curvilinear polygons.

Key words: art gallery, curvilinear polygons, vertex guards, point guards, piecewise-convex polygons, piecewise-concave polygons
2000 MSC: 68U05, 68W40

1. Introduction

In the classical art gallery problem, an *art gallery* is represented by a simply connected closed polygonal domain (for short *polygon*) P. The art gallery is monitored by a set of guards, each represented by a point in P, if every point in P is visible to at least one of the guards. Two points see each other if they are visible to each other, i.e., if the closed line segment connecting them lies in P. Victor Klee asked what is the minimum number of guards that can monitor any polygon with $n \geq 3$ vertices. Art gallery-type problems have found applications in robotics [1, 2], motion planning [3, 4], computer vision and pattern recognition [5, 6, 7, 8], graphics [9, 10], CAD/CAM [11, 12] and wireless networks [13]. Curvilinear objects were typically modeled with straight-line polygonal approximations. Starting from the late 80s, some geometric algorithms were extended to curvilinear polygons [14]. Refer to the recent book edited by Boissonnat and Teillaud [15] for a collection of computational-geometry results for curves and surfaces. In this context this paper addresses the classical art gallery problem for various classes of polygonal regions bounded by curvilinear edges. To the best of our knowledge this is the first time that the art gallery problem is considered in this context.

The first results on art gallery-type problems date back to the 1970's. Chvátal [16] proved that every simple polygon with n vertices can be monitored by $\lfloor \frac{n}{3} \rfloor$ vertex guards; this bound is tight in the worst case.

[*] Corresponding author
Email addresses: mkaravel@tem.uoc.gr (Menelaos I. Karavelas), cdtoth@math.ucalgary.ca (Csaba D. Tóth), elias.tsigaridas@loria.fr (Elias P. Tsigaridas)

Later Fisk [17] gave an elegant algorithmic proof using a 3-coloring of a triangulation of the polygon. Fisk's algorithm runs in $O(n)$ time for a triangulated polygon with n vertices, and the time complexity of the triangulation is $O(n)$ based on Chazelle's algorithm [18]. Lee and Lin [19] showed that finding the minimum number of vertex guards for a given simple polygon is NP-hard, which was extended to point guards by Aggarwal [20]. Other types of art galleries have also been considered. Kahn, Klawe and Kleitman [21] showed that every simple orthogonal polygon, i.e., simple polygon with axes-aligned edges, with n vertices can be monitored by $\lfloor \frac{n}{4} \rfloor$ vertex guards, and this bound is best possible. Several $O(n)$ time algorithms have been proposed for placing the guards in this variation of the problem, notably by Sack [22] and later by Lubiw [23]. Edelsbrunner, O'Rourke and Welzl [24] gave an $O(n)$ time algorithm for placing $\lfloor \frac{n}{4} \rfloor$ point guards that jointly monitor an orthogonal polygon with n vertices. Other types of guarding problems have also been studied in the literature. For a detailed discussion of these variations and the corresponding results the interested reader should refer to the book by O'Rourke [25], or the survey papers by Shermer [26] and by Urrutia [27].

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 (defined below); we call such polygons *piecewise-convex* and *piecewise-concave polygons*, respectively.

We show that every piecewise-convex polygon with $n \geq 2$ vertices can be monitored by at most $\lfloor \frac{2n}{3} \rfloor$ vertex guards. This bound is tight: there are piecewise-convex polygons with n vertices, for every $n \geq 2$, that cannot be monitored by fewer than $\lfloor \frac{2n}{3} \rfloor$ vertex guards. Our upper bound is based on an algorithm for placing vertex guards, which can be implemented in $O(n \log n)$ time and $O(n)$ space. Our algorithm is a generalization of Fisk's algorithm [17]; in fact, when applied to a straight-line polygon with $n \geq 3$ vertices, it produces at most $\lfloor \frac{n}{3} \rfloor$ vertex guards. For the purposes of our complexity analysis and results, we assume, throughout the paper, that the curvilinear edges of our polygons are arcs of algebraic curves of constant degree. As a result, all predicates required by the algorithms described in this paper take $O(1)$ time in the real RAM model of computation model. The central idea for our upper bound is the approximation of a piecewise-convex polygon by a straight-line polygon by adding Steiner vertices on the boundary of the curvilinear polygon. The resulting polygonal approximation is a simple straight-line polygon. We compute a guard set for the polygonal approximation by a slightly modified version of Fisk's algorithm [17]. This guard set monitors the original curvilinear polygon, however, vertex guards may be located at Steiner vertices. The final step of our algorithm maps the vertex guards of the polygonal approximation to vertex guards of the curvilinear polygon. Our upper bound of $\lfloor \frac{2n}{3} \rfloor$ also applies to point guards. However, it does not match the best lower bound we have found. There are piecewise-convex polygons with n vertices, for every $n \geq 2$, that cannot be monitored by fewer than $\lceil \frac{n}{2} \rceil$ point guards.

Some piecewise-concave polygons have interior points hidden from all vertices (see Fig. 14(a)), and hence they cannot be monitored by vertex guards alone. We thus turn our attention to point guards, and we show that $2n - 4$ point guards are always sufficient and sometimes necessary for monitoring a piecewise-concave polygon with $n \geq 3$ vertices. Our upper bound proof is based on Fejes Tóth's technique for illuminating sets of disjoint convex objects in the plane [28]. Given a piecewise-concave polygon P, we subdivide P into crescents (bounded by a convex and a concave arc), each adjacent to an edge of P, and into convex polygonal holes. Using Fejes Tóth's argument, if we place guards at points incident to at least three crescents, at two vertices of each triangular hole and all vertices at holes with 4 or more vertices, we obtain a guard set that monitors all holes and all crescents, hence the entire piecewise-concave polygon P. Since the intersection graph of the crescents is outerplanar, whose faces correspond to the holes, it is easy to show that the number of point guards is at most $2n - 4$.

The rest of the paper is structured as follows. In Section 2 we define curvilinear polygons, including piecewise-convex and piecewise-concave polygons. In Section 3 we present our algorithm for computing a vertex guard set, of size $\lfloor \frac{2n}{3} \rfloor$, for a piecewise-convex polygon with n vertices, and present families of piecewise-convex polygons that require a minimum of $\lfloor \frac{2n}{3} \rfloor$ vertex or $\lceil \frac{n}{2} \rceil$ point guards in order to be monitored. In Section 4 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. The final section of the paper, Section 5, discusses further results and states open problems.

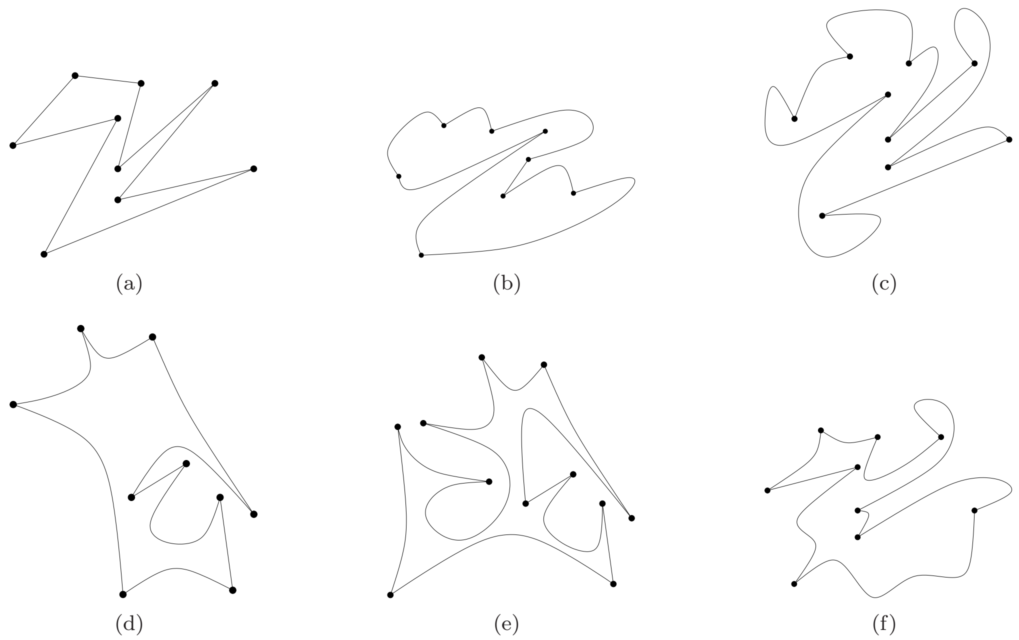

Figure 1: Different types of curvilinear polygons: (a) a straight-line polygon, (b) a piecewise-convex polygon, (c) a locally convex polygon, (d) a piecewise-concave polygon, (e) a locally concave polygon and (f) a general polygon.

2. Definitions

Types of curvilinear polygons. Let V be a sequence of points $v_1, \ldots, v_n$, $n \geq 2$, and A a set of curvilinear arcs $a_1, \ldots, a_n$, such that the endpoints of a_i are v_i and v_{i+1}[1]. We 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 *curvilinear 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 either at most two points or along a line segment.

A polygon P is a *straight-line polygon* if its edges are line segments (see Fig. 1(a)). A polygon P is *locally convex* (see Fig. 1(c)), (resp., *locally concave* (see Fig. 1(e))), if for every point p on the boundary of P, with the possible exception of P's vertices, there exists a disk centered at p, say D_p, such that $P \cap D_p$ is convex (resp., concave). A polygon P is *piecewise-convex* (see Fig. 1(b)), (resp., *piecewise-concave* (see Fig. 1(d))), if it is locally convex (resp., concave), and the portion of the boundary between every two consecutive vertices is a convex arc. 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 use the term *curvilinear polygon* to refer to a polygon the edges of which are either line or curve segments.

Guards and guard sets. In our setting, a *guard* or *point guard* is a point in the interior or on the boundary of a curvilinear polygon P. A guard of P that is also a vertex of P is called a *vertex guard*. We say that a curvilinear polygon P is *monitored* by a set G of guards if every point in P is visible from at least one point in G, where two points p and q in P are visible from each other if the line segment pq lies entirely in P. The set G that has this property is called a *guard set* for P. A guard set that consists solely of vertices of P is called a *vertex guard set*.

[1]Indices are evaluated modulo n.

3. Piecewise-convex polygons

In this section we present an algorithm which, given a piecewise-convex polygon P with n vertices, computes a vertex guard 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 guard set $G_{\tilde{P}}$ for $\tilde{P}$, by 3-coloring the vertices of $\mathcal{T}(\tilde{P})$.
4. Compute a guard set G_P for P from the guard 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_iv_{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_iv_{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. a room is *empty* if it is non-degenerate and does not contain any vertex of P in its interior or in the interior of its chord.
2. a room is *non-empty* if it is non-degenerate and contains at least one vertex of P in its interior or in the interior of its chord.

In order to polygonalize P we add Steiner vertices in the interior of non-linear convex arcs. 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_iv_{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_iv_{i+1} and $\ell_i^{\perp}(p)$ the line perpendicular to v_iv_{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}, $1 \leq j_k \leq |C_i^*|$, be the (unique) intersection of the line m_iv_k with the arc a_i; if $C_i^* = X_i$, then, for each $v_k \in C_i^*$, let w_{i,j_k}, $1 \leq j_k \leq |C_i^*|$, be the (unique) intersection of the line $\ell_i^{\perp}(v_k)$ with the arc a_i.

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

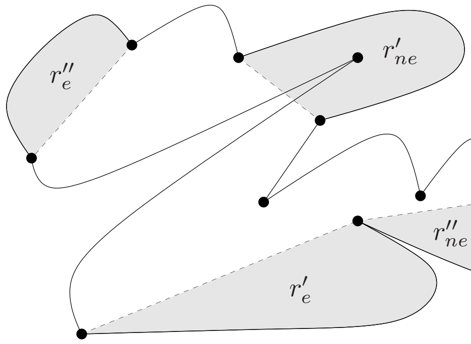

Figure 2: The two types of rooms in a piecewise-convex polygon: r_e' and r_e'' are empty rooms, whereas r_{ne}' and r_{ne}'' are non-empty rooms.

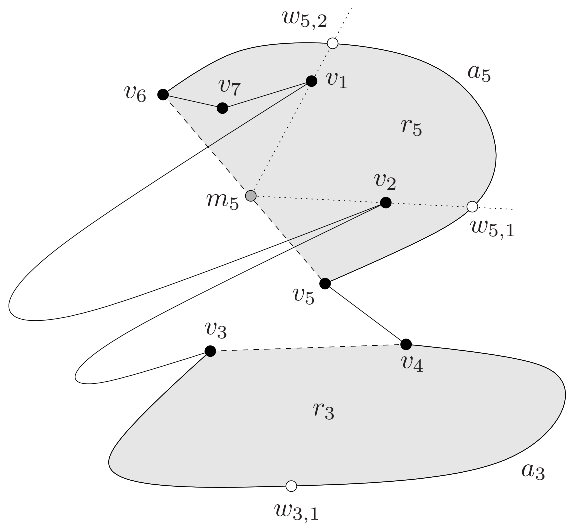

Figure 3: The Steiner vertices (white points) for rooms r_3 (empty) and r_5 (non-empty). $w_{3,1}$ is a point in the interior of a_3. m_5 is the midpoint of the line segment v_5v_6, whereas $w_{5,1}$ and $w_{5,2}$ are the intersections of the lines m_5v_2 and m_5v_1 with the arc a_5, respectively. In this example $R_5 = \{v_1, v_2, v_7\}$, whereas $C_5^* = \{v_1, v_2\}$.

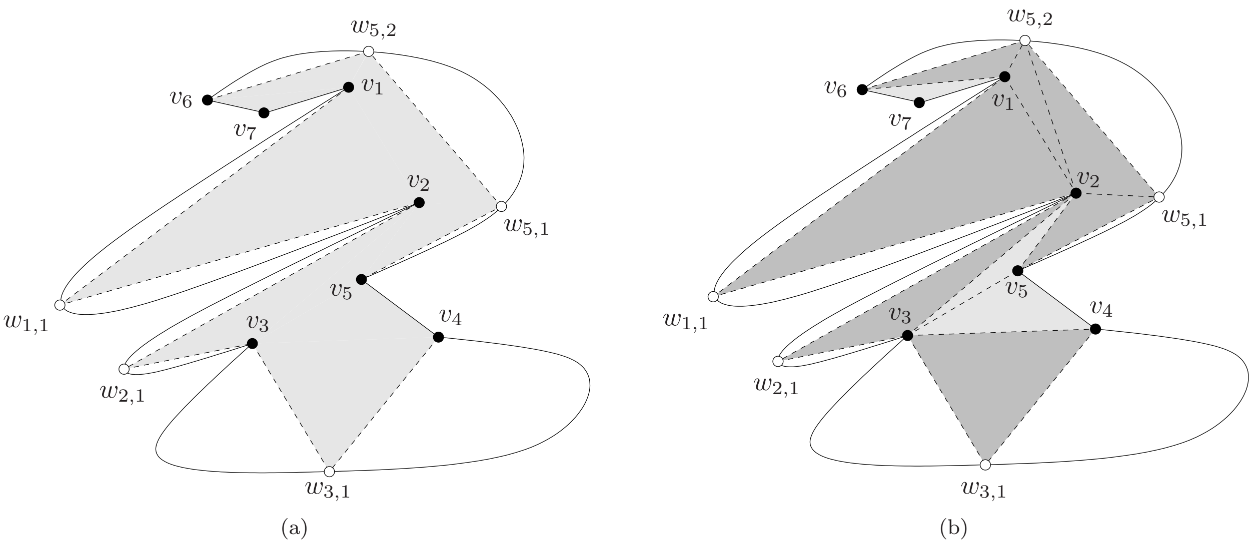

Figure 4: (a) The polygonal approximation $\tilde{P}$, shown in gray, of the piecewise-convex polygon P with vertices v_i, $i = 1, \ldots, 7$. (b) The constrained triangulation $\mathcal{T}(\tilde{P})$ of $\tilde{P}$. The dark gray triangles are the constrained triangles. The polygonal region $v_5w_{5,1}w_{5,2}v_6v_1v_2v_5$ is a crescent. The triangles $w_{5,1}v_2v_5$ and $v_1w_{5,2}v_6$ are boundary crescent triangles. The triangle $v_2w_{5,2}v_1$ is an upper crescent triangle, whereas the triangle $v_2w_{5,1}w_{5,2}$ is a lower crescent triangle.

Lemma 1. *The straight-line polygon $\tilde{P}$ is a simple polygon.*

Proof. It suffices show that the line segments replacing the curvilinear 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 $v_iw_{i,1}$ and $w_{i,1}v_{i+1}$ lie in the interior of r_i. Since P is a piecewise-convex 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 $w_{i,1}, \ldots, w_{i,K_i}$ be the points added on a_i, where K_i is the cardinality of C_i^*. By construction, every point $w_{i,k}$ is visible from $w_{i,k+1}$, $k = 1, \ldots K_i - 1$, and every point $w_{i,k}$ is visible from $w_{i,k-1}$, $k = 2, \ldots K_i$. Moreover, $w_{i,1}$ is visible from v_i and w_{i,K_i} is visible from v_{i+1}. Therefore, the interior of the segments in the polyline $v_i w_{i,1} \ldots w_{i,K_i} v_{i+1}$ lie in the interior of r_i and do not intersect any arc in P. Hence, substituting a_i by the polyline $v_i w_{i,1} \ldots w_{i,K_i} v_{i+1}$ gives us a new piecewise-convex polygon.

As a result, the straight-line polygon $\tilde{P}$ is a simple polygon. □

We call the straight-line polygon $\tilde{P}$, defined by $\tilde{V}$, 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 straight-line polygon that is contained in P.*

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

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

Proof. Consider an arc a_i of P, delimited by the vertices v_i and v_{i+1} and let π_i denote the shortest path in P between them. Note that π_i is a straight-line polygonal path, the internal vertices of which are the vertices of C_i^*. Since a_i is a convex arc, π_i is also a convex arc. a_i and π_i bound a (curvilinear) polygon, that we denote by Q_i, for which π_i is locally concave. That is, every point in C_i^* is a reflex vertex of Q_i, and so every point in C_i^* is a reflex (i.e., locally concave) vertex of P as well. At every vertex $w \in C_i^*$, the bisector of the internal angle of P enters the polygon Q_i and leaves Q_i (and P) at some point along a_i.

Consider the bisector of the internal angle at every reflex vertex w of P. If the bisector intersects some arc a_j, then w can belong to the set C_j^* only. Since every bisector intersects at most one arc a_j (we are referring to the first intersection of the bisector while walking on it away from w), every vertex w belongs to at most one set C_j^*. □

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

Corollary 4. *The number of vertices of the polygonal approximation $\tilde{P}$ of a piecewise-convex polygon P with n vertices is at most $3n$. This bound is tight (up to an additive constant).*

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

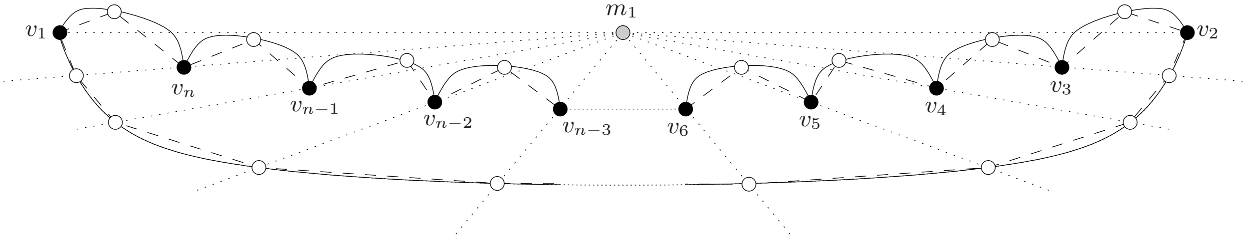

Figure 5: A piecewise-convex polygon P with n vertices (solid curve), the polygonal approximation $\tilde{P}$ of which consists of $3n - 3$ vertices (dashed polyline).

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

The upper bound of the paragraph above is tight up to an additive 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 $|\tilde{V}| = 3n-3$. □

3.2. Triangulating the polygonal approximation

Let P be a piecewise-convex polygon, $\tilde{P}$ its polygonal approximation, and $S_{\tilde{P}}$ the set of Steiner vertices in $\tilde{P}$. We construct a *constrained triangulation* of $\tilde{P}$, i.e., we triangulate $\tilde{P}$, while imposing some triangles to be part of this triangulation. More precisely, we constrain the triangles of $\mathcal{T}(\tilde{P})$ created in the neighborhood of the vertices in $S_{\tilde{P}}$. By constraining the triangles in these neighborhoods, we effectively triangulate parts of $\tilde{P}$. The remaining untriangulated parts of $\tilde{P}$ consist of one or more interior disjoint straight-line polygons, which are then triangulated arbitrarily in linear time and space. We call the pre-specified triangles in $\mathcal{T}(\tilde{P})$ *constrained triangles*. We want the triangulation $\mathcal{T}(\tilde{P})$ to satisfy the following properties:

1. every triangle of $\mathcal{T}(\tilde{P})$, with a vertex in $S_{\tilde{P}}$, also contains at least one vertex of P, i.e., no triangles contain only Steiner vertices,
2. every vertex in $S_{\tilde{P}}$ 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 monitored by vertices of P.

These properties are exploited in Step 4 of the algorithm presented later in this subsection.

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 Steiner vertex 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 forming the constrained triangle $v_i w_{i,1} v_{i+1}$ (see Fig. 4(b)). If r_i is a non-empty room, $c_1, \ldots, c_{K_i}$ the vertices in C_i^*, $K_i = |C_i^*|$, and $w_{i,1}, \ldots, w_{i,K_i}$ the Steiner vertices in a_i ($w_{i,j}$ has been added on a_i due to c_j), add the following edges, if they do not already exist:

1. c_k, c_{k+1}, for $k = 1, \ldots, K_i - 1$, and $v_i c_1$, $c_{K_i} v_{i+1}$;
2. $c_k w_{i,k}$, for $k = 1, \ldots, K_i$;
3. $c_k w_{i,k+1}$, for $k = 1, \ldots, K_i - 1$;
4. $w_{i,k}, w_{i,k+1}$, for $k = 1, \ldots, K_i - 1$, and $v_i w_{i,1}$, $w_{i,K_i} v_{i+1}$.

These edges form $2K_i$ constrained triangles: $c_k c_{k+1} w_{i,k+1}$, for $k = 1, \ldots, K_i - 1$; $c_k w_{i,k} w_{i,k+1}$, for $k = 1, \ldots, K_i - 1$; $v_i c_1 w_{i,1}$ and $v_{i+1} c_{K_i} w_{i,K_i}$. We call the polygonal region formed by these triangles a *crescent*. The triangles $v_i c_1 w_{i,1}$ and $v_{i+1} c_{K_i} w_{i,K_i}$ are called *boundary crescent triangles*, the triangles $c_k c_{k+1} w_{i,k+1}$, $k = 1, \ldots, K_i - 1$, are called *upper crescent triangles*, whereas the triangles $c_k w_{i,k} w_{i,k+1}$, $k = 1, \ldots, K_i - 1$, are called *lower crescent triangles*.

Note that the points $w_{i,j}$, $j < K_i$ (resp., w_{i,K_i}) are vertices of exactly one triangle (resp., exactly two triangles) in $\mathcal{T}(\tilde{P})$, such that the other two vertices of the triangle (resp., of each of the two triangles) belong to P.

3.3. Computing a guard set for the original polygon

Assume that we have colored the vertices $\tilde{V}$ of $\tilde{P}$ with three colors, so that no triangle in $\mathcal{T}(\tilde{P})$ contains two vertices of the same color. This can be easily done by the standard 3-coloring algorithm for straight-line polygons presented in [29, 17]. Let red, green and blue be the three colors, and let K_A, Π_A and M_A be the set of vertices of A of red, green and blue color, respectively, where A stands for either P, $\tilde{P}$ or $S_{\tilde{P}}$. Clearly, all three sets $K_{\tilde{P}}$, $\Pi_{\tilde{P}}$ and $M_{\tilde{P}}$ are guard sets for $\tilde{P}$. In fact, they are also guard sets for P, as the following lemma suggests (see also Fig. 6).

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

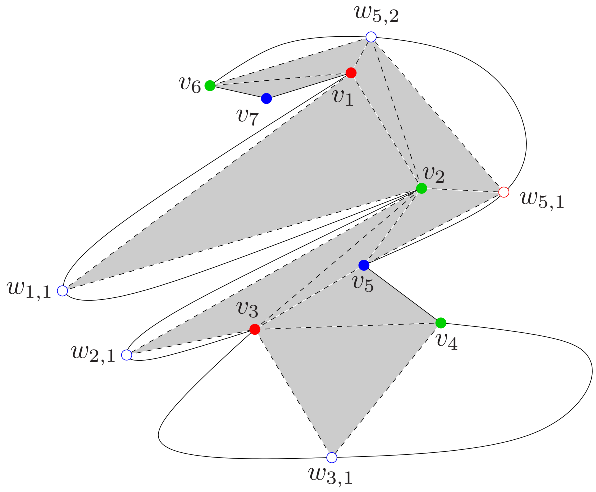

Figure 6: The three guard sets for $\tilde{P}$, are also guard sets for P, as Lemma 5 suggests.

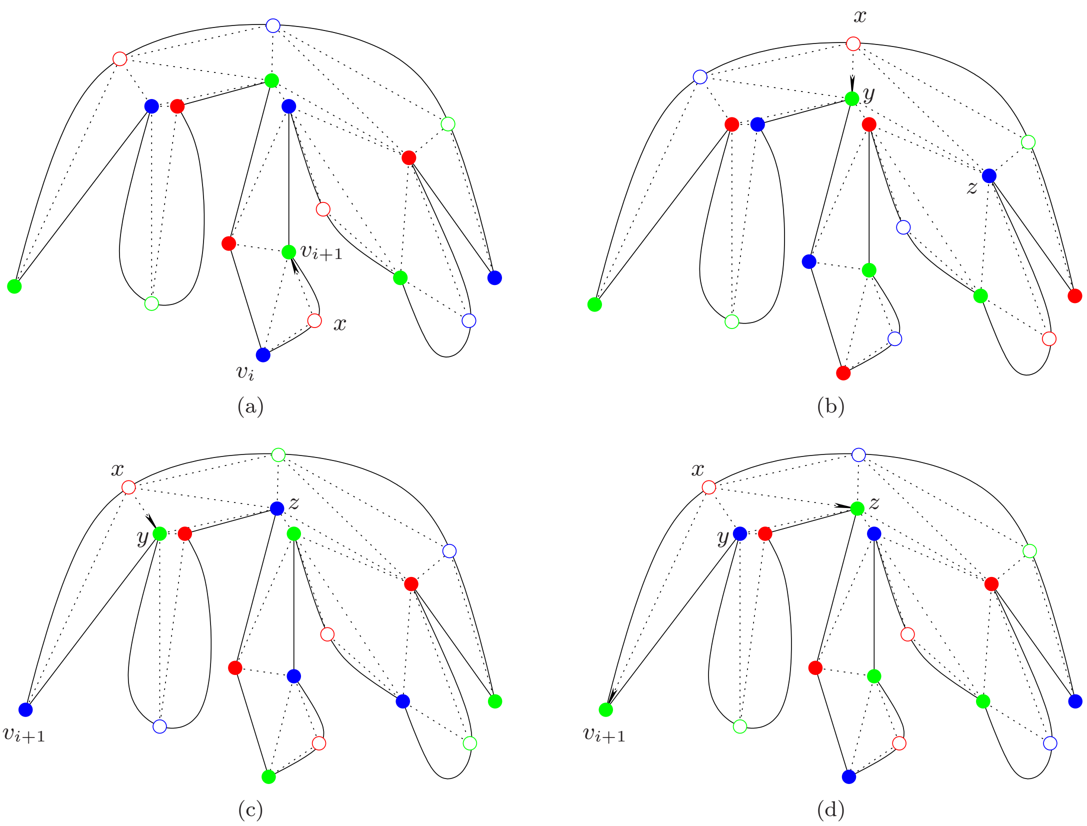

Figure 7: The three cases in the definition of the mapping f. Case (a): x is a Steiner vertex in an empty room. Case (b): x is an Steiner vertex in a non-empty room and is not the last Steiner vertex added on the curvilinear arc. Cases (c) and (d): x is the last Steiner vertex added on the curvilinear arc of a non-empty room (in (c) $|f(x)| = 1$, whereas in (d) $|f(x)| = 2$).

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

Indeed, let s be a non-degenerate sector associated with the convex arc a_i, and let $T \in \mathcal{T}(\tilde{P})$ be the triangle incident to the chord of s. If r_i is an empty room, each of the three vertices of T monitors r_i (and therefore also s). If r_i is a non-empty room, the vertex of T that is not an endpoint of the chord of s is a vertex in C_i^* and monitors s by construction. Clearly, one of the three vertices of T belongs to $G_{\tilde{P}}$. □

Let as now assume, without loss of generality, that $|K_P| \leq |\Pi_P| \leq |M_P|$. Define the mapping f from $K_{S_{\tilde{P}}}$ to the power set 2^{Π_P} of Π_P by mapping a vertex x in $K_{S_{\tilde{P}}}$ to all the neighboring vertices of x in $\mathcal{T}(\tilde{P})$ that belong to Π_P (see Fig. 7 for the three possible cases for x). Notice that $1 \leq |f(x)| \leq 2$.

Finally, define the set $G_P = K_P \cup f(K_{S_{\tilde{P}}})$, where $f(K_{S_{\tilde{P}}}) = \bigcup_{x \in K_{S_{\tilde{P}}}} f(x)$. We claim that G_P is a guard set for P.

Lemma 6. *The set $G_P = K_P \cup f(K_{S_{\tilde{P}}})$ is a guard set for P.*

Proof. The regions in $P \setminus \tilde{P}$ are sectors bounded by a curvilinear arc, which is a subarc of an edge of P, and the corresponding chord connecting the endpoints of this subarc. To show that G_P is a guard set for P, it suffices show that every triangle in $\mathcal{T}(\tilde{P})$ and every sector in $P \setminus \tilde{P}$ is monitored by at least one vertex in G_P.

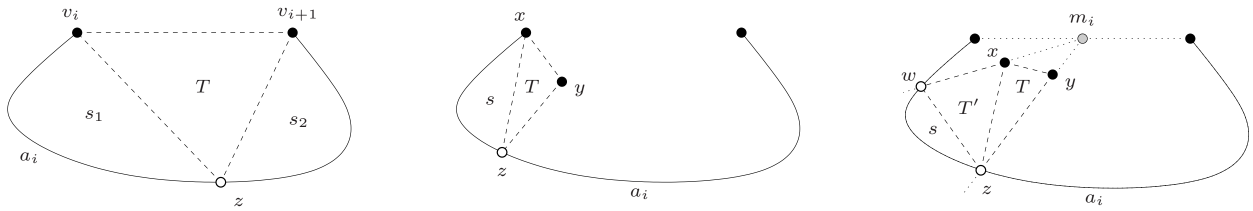

Figure 8: Proof of Lemma 6. From left to right: the case of empty rooms; the case of boundary crescent triangles; the case of upper and lower crescent triangles.

If all three vertices of a triangle $T \in \mathcal{T}(\tilde{P})$ are vertices of P, one of the vertices of T is in $K_P \subseteq G_P$. If T is a triangle in an empty room (see Fig. 8(left)), or a boundary crescent triangle (see Fig. 8(middle)), either the unique Steiner vertex z of T is in $K_{S_{\tilde{P}}}$, in which case one of the other two vertices of T belongs to $f(K_{S_{\tilde{P}}})$, or z is not in $K_{S_{\tilde{P}}}$, in which case one of the other two vertices of T belongs to K_P. Moreover, the sector/sectors adjacent to an edge of T in r_i is/are visible by both vertices of T in P and thus monitored by one of them. Finally, upper and lower crescent triangles come in pairs. Let T be an upper crescent triangle in a non-empty room r_i (see Fig. 8(right)). Let x, y be the vertices of T in P, and let z be its vertex in $S_{\tilde{P}}$; it is assumed here that z is the intersection of $m_i y$ with a_i. Let T' be the lower crescent triangle adjacent to T along the edge xz, w be the third vertex of T', and s be the sector in $P \setminus \tilde{P}$ adjacent to zw. Since x and y belong to C_i^*, either x or y monitors T, T' and s. We end the proof by claiming that either x or y belongs to G_P: if x or y belongs to K_P the claim is obvious; if neither x nor y belongs to K_P, then $z \in K_{S_{\tilde{P}}}$ in which case one of x and y belongs to $f(K_{S_{\tilde{P}}})$. □

Since $f(K_{S_{\tilde{P}}}) \subseteq \Pi_P$ we get that $G_P \subseteq K_P \cup \Pi_P$. Since K_P and Π_P are the two sets of smallest cardinality among K_P, Π_P and M_P, we conclude that $|G_P| \leq |K_P| + |\Pi_P| \leq \lfloor \frac{2n}{3} \rfloor$, and thus arrive at the following theorem.

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

We close this subsection by making two remarks:

Remark 1. *When the input to our algorithm is a straight-line polygon all rooms are degenerate; consequently, no Steiner vertices are created, and the guard set computed corresponds to the set of colored vertices of smallest cardinality, hence producing a vertex guard set of size at most $\lfloor \frac{n}{3} \rfloor$. In that respect, our algorithm can be viewed as a generalization of Fisk's algorithm [17] to the class of piecewise-convex polygons.*

Remark 2. *Given a straight-line polygon P with $r \geq 2$ reflex vertices, we can view P as a piecewise-convex polygon the edges of which are c convex polylines, where $c \geq r$. In this context Theorem 7 can be "translated" as follows:*

> *If the boundary of a simple straight-line polygon P can be partitioned into $c \geq 2$ convex polylines such that P is a piecewise-convex polygon with its edges being the c convex polylines, then P can be monitored with at most $\lfloor \frac{2c}{3} \rfloor$ vertex guards.*

3.4. Time and space complexity

In this subsection we show how to compute the vertex guard set G_P in $O(n \log n)$ time and $O(n)$ space. It is straightforward to show that Steps 2–4 of our algorithm (see beginning of Section 3) can be implemented in linear time and space. 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. In order to compute $\tilde{P}$, it suffices 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 straight-line 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 we then compute the corresponding set C_i^*. This is done by first computing a subset S_i of the set R_i of the points in the room r_i, such that $S_i \supseteq C_i^*$, and then applying 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 are 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 is done in two steps:

1. First we 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 and let V' be the vertex set of P'. Since each a_i can yield up to three y-monotone convex pieces, we conclude that $|V'| \leq 3n$. 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 are called an *added extremal vertex*.
2. Second, we apply to P' the standard algorithm for computing y-monotone subpolygons of a straight-line polygon (cf. [30] or [31]). The algorithm in [30] (or [31]) is valid not only for line segments, but also for piecewise-convex polygons consisting of y-monotone arcs (such as P'). Since $|V'| \leq 3n$, we conclude that computing 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).

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* (see Fig. 9). For each non-bridge edge e_i of Q, we want to compute the set C_i^*. This is 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. There are 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,

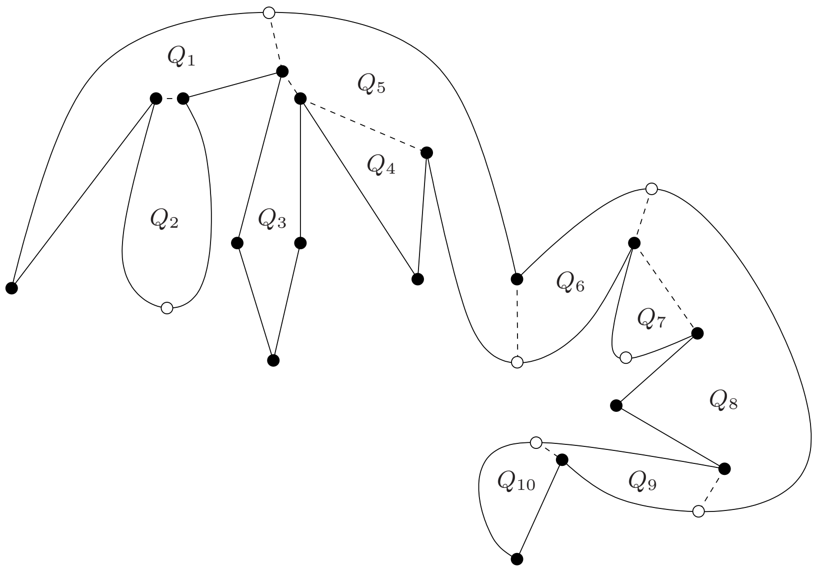

Figure 9: Decomposition of a piecewise-convex polygon into ten y-monotone subpolygons. The white points are added extremal vertices that have been added in order to split non-y-monotone arcs to y-monotone pieces. The bridges are shown as dashed segments.

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 ℓ be the sweep line parallel to the x-axis at some y. For each y in between the y-maximal and y-minimal values of Q, ℓ intersects Q at two points which belong to either a left edge e_l or a left vertex v_l (i.e., an edge or vertex on the left y-monotone chain of Q), and either a right edge e_r or a right vertex v_r (i.e., a edge or vertex on the right y-monotone chain of Q). We 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) contains vertices of Q that are in the room of the convex arc of P corresponding to 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_l, let $e_{l,up}$ be the left edge above v_l and $e_{l,down}$ be the left edge below v_l and let e_r be the current right edge. If $e_{l,up}$ is an 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_l is in the room r_k or lies in the interior of the chord of r_k; if this is the case we add v_l to S_R. When a right event is encountered our sweep algorithm behaves symmetrically. When the last event is encountered due to the y-minimal vertex v_{min}, let e_l (resp., e_r) be the left (resp., right) edge of Q above v_{min}. If e_l (resp., e_r) is a non-bridge edge, let a_i (resp., a_j) be the corresponding convex arc in P. In this case we simply augment S_i (resp., S_j) by the vertices in S_L (resp., 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 room r_i of P, with a_i the corresponding convex arc, the vertices of the set C_i^* belong to the y-monotone subpolygons of P' computed via the algorithm in [30] (or [31]), which either contain the entire arc a_i or y-monotone pieces of a_i.*

Proof. Let u be a vertex of P in C_i^* that is not a vertex of any of the y-monotone subpolygons of P' (computed by the algorithm in [30] or [31]) that contain either the entire arc a_i or y-monotone pieces of a_i.

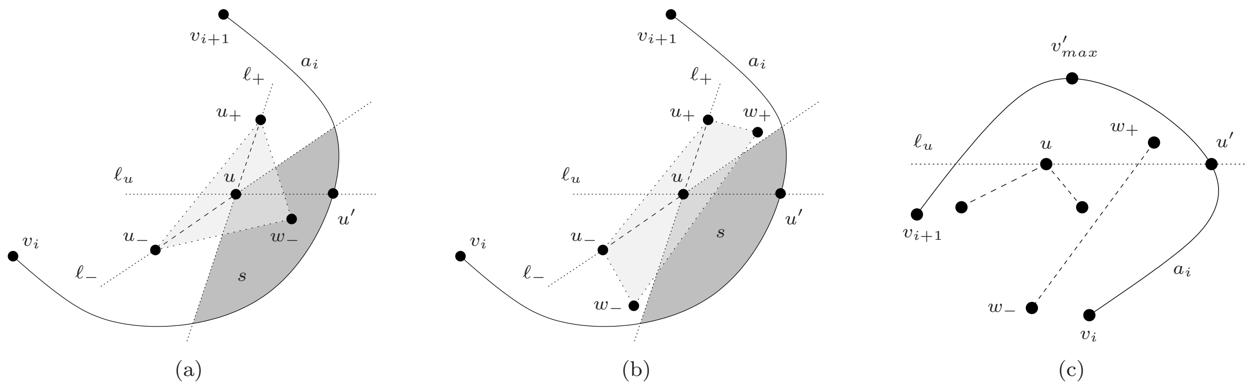

Figure 10: Proof of Lemma 8. (a) The case $u \in C_i^* \setminus \{v_{min}, v_{max}\}$, with $w_- \in s$. (b) The case $u \in C_i^* \setminus \{v_{min}, v_{max}\}$, with $w_+, w_- \notin s$. (c) The case $u \equiv v_{max}$.

Let v_{max} (resp., v_{min}) be the vertex of P of maximum (resp., minimum) y-coordinate in C_i; ties are broken lexicographically. Let ℓ_u be the line parallel to the x-axis passing through u. Consider the following cases:

1. $u \in C_i^* \setminus \{v_{min}, v_{max}\}$. Without loss of generality we can assume that u is a vertex in the right y-monotone chain of C_i (see Figs. 10(a) and 10(b)). Let u' be the intersection of ℓ_u with a_i. Let Q (resp., Q') be the y-monotone subpolygon of P' that contains u (resp., u'); by our assumption $Q \neq Q'$. Finally, let u_+ (resp., u_-) be the vertex of C_i above (resp., below) u in the right y-monotone chain of C_i.
   The line segment uu' cannot intersect any edges of P, since this would contradict the fact that $u \in C_i^*$. Similarly, uu' cannot contain any vertices of P': if v is a vertex of P in the interior of uu', u would be in the triangle vu_+u_-, which contradicts the fact that $u \in C_i^*$, whereas if v is a vertex of $V' \setminus V$ in the interior of uu', P would not be locally convex at v, a contradiction with the fact that P is a piecewise-convex polygon. As a result, and since $Q \neq Q'$, there exists a bridge edge e intersecting uu'. Let w_+, w_- be the two endpoints of e in P', where w_+ lies above the line ℓ_u and w_- lies below the line ℓ_u. In fact neither w_+ nor w_- can be a vertex in $V' \setminus V$, since the algorithm in [30] (or [31]) connects a vertex in $V' \setminus V$ in a room r_k with either the y-maximal or the y-minimal vertex of C_k only. Let ℓ_+ (resp., ℓ_-) be the line passing through the vertices u and u_+ (resp., u and u_-). Finally, let s be the sector delimited by the lines ℓ_+, ℓ_- and a_i. Now, if w_+ or w_- lies in s, then u is in the triangle $w_+u_+u_-$ or in the triangle $w_-u_+u_-$, respectively (see Fig. 10(a)). In either case we get a contradiction with the fact that $u \in C_i^*$. If neither w_+ nor w_- lie in s, then both w_+ and w_- have to be vertices in r_i, and moreover u lies in the convex quadrilateral $w_+u_+u_-w_-$; again this contradicts the fact that $u \in C_i^*$ (see Fig. 10(b)).
2. $u \equiv v_{max}$. By the maximality of the y-coordinate of u in C_i, we have that the y-coordinate of u is larger than or equal to the y-coordinates of both v_i and v_{i+1}. Therefore, the line ℓ_u intersects the arc a_i exactly twice, and, moreover, a_i has a y-maximal vertex of $V' \setminus V$ in its interior, which we denote by v'_{max} (see Fig. 10(c)). Let u' be the intersection of ℓ_u with a_i that lies to the right of u, and let Q (resp., Q') be the y-monotone subpolygon of P' that contains u (resp., u'). By assumption $Q \neq Q'$, which implies that there exists a bridge edge e intersecting the line segment uu'. Notice, that, as in the case $u \in C_i^* \setminus \{v_{min}, v_{max}\}$, the line segment uu' cannot intersect any edges of P, or cannot contain any vertex v of $V' \setminus V$; the former would contradict the fact that $u \in C_i^*$, whereas as the latter would contradict the fact that P is piecewise-convex. Furthermore, uu' cannot contain vertices of P since this would contradict the maximality of the y-coordinate of u in C_i.
   Let w_+ and w_- be the endpoints of e above and below ℓ_u, respectively. Notice that e cannot have v'_{max} as endpoint, since the only bridge edge that has v'_{max} as endpoint is the bridge edge $v'_{max}u$. But then w_+ must be a vertex of P lying in r_i; this contradicts the maximality of the y-coordinate of u among the vertices in C_i.

3. $u \equiv v_{min}$. This case is entirely symmetric to the case $u \equiv v_{max}$. $\square$

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. Then 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. During the course of our algorithm we only perform 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 and the C_i^*'s. The analysis above thus yields the following:

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

3.5. Lower bound constructions

In this subsection we present an n-vertex piecewise-convex polygon, for every $n \geq 2$, that cannot be monitored by fewer than $\lfloor \frac{2n}{3} \rfloor$ vertex guards (resp., $\lceil \frac{n}{2} \rceil$ point guards).

It is clear that a piecewise-convex 2-gon (e.g., Fig. 11(a)) requires 1 vertex guard. Fig. 11(b) depicts a piecewise-convex triangle that cannot be monitored by fewer than 2 vertex or point guards.

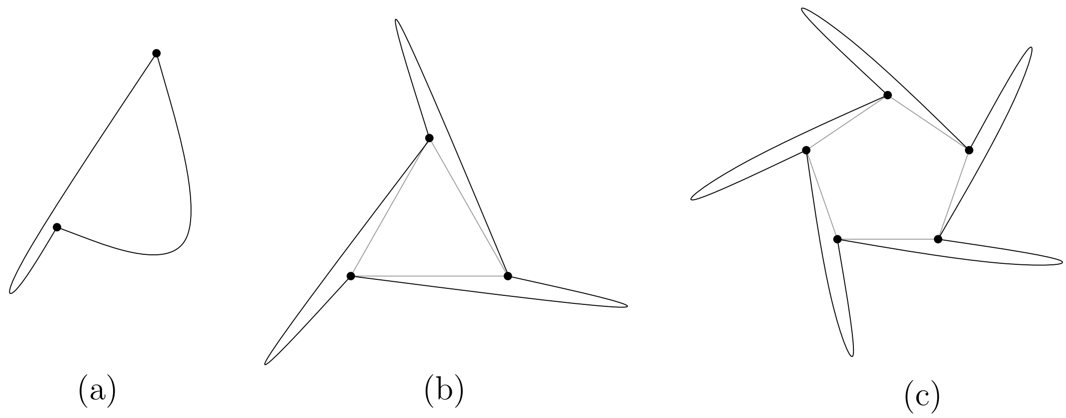

Figure 11: (a) a piecewise-convex 2-gon; (b) a piecewise-convex triangle that requires 2 vertex guards; (c) a piecewise-convex pentagon that requires 3 point guards.

For every integer $n \geq 4$, we give a construction based on a regular k-gon $a_1 a_2 \ldots a_k$, where $k = \lceil \frac{n}{3} \rceil \geq 2$ (in particular, for $k = 2$, a 2-gon is a line segment $a_1 a_2$). First assume that $n = 3k$ for an integer $k \geq 2$. Let κ denote the circumscribed circle of $a_1 a_2 \ldots a_k$. Replace each edge $a_i a_{i+1}$, $i = 1, 2, \ldots, k$, by a piecewise-convex path (a_i, b_i, c_i, a_{i+1}) depicted in Fig. 12(b), to obtain a piecewise-convex n-gon P. The vertices b_i and c_i are in the left open halfplane delimited by the directed line $\overrightarrow{a_i a_{i+1}}$ and they are separated from the polygon $a_1 a_2 \ldots a_k$ by the tangent of κ at a_i. The patterns (a_i, b_i, c_i, a_{i+1}) are designed such that at each vertex of P, the tangents of the two adjacent edges are the same, which we call the *common tangent* at the vertex. The common tangent at a_i is also tangent to the circle κ at a_i; the common tangent at b_i is parallel to the common tangent at a_i; and the common tangent at c_i is perpendicular to the common tangents at a_i and b_i. Let A_i and B_i denote the empty rooms bounded by $a_i b_i$ and $b_i c_i$, respectively. Let C_i denote the part of the (non-empty) room bounded by $c_i a_{i+1}$ that lies on the left side of both directed lines $\overrightarrow{a_i a_{i+1}}$ and $\overrightarrow{a_i c_i}$. Note that the regions A_i, B_i, and C_i are hidden from any vertex of P other than a_i, b_i, c_i. However, none of a_i, b_i, c_i sees all three regions A_i, B_i, C_i entirely (in particular, a_i does not see B_i entirely; b_i doers not see C_i entirely; and c_i does not see A_i entirely). Hence each triple of regions $\{A_i, B_i, C_i\}$ requires at least two vertex guards at $\{a_i, b_i, c_i\}$. This gives a lower bound of $2k = \frac{2n}{3}$, if $n = 3k$, $k \geq 2$.

Now assume that $n = 3k - 2$ for an integer $k \geq 2$. Replace every edge $a_i a_{i+1}$, for $i = 1, 2, \ldots, k-1$, by a piecewise-convex path (a_i, b_i, c_i, a_{i+1}) depicted in Fig. 12(b). The previous argument shows that the resulting piecewise-convex n-gon requires $2(k-1) = \lfloor \frac{2n}{3} \rfloor$ vertex guards. Finally, assume that $n = 3k - 1$ for $k \geq 2$. Replace every edge $a_i a_{i+1}$, for $i = 1, 2, \ldots, k-1$, by a piecewise-convex path (a_i, b_i, c_i, a_{i+1}) depicted in Fig. 12(b); and replace edge $a_k a_1$ by (a_k, b_k, a_1) depicted in Fig. 12(c). The common tangent at b_k in Fig. 12(c) passes through side $a_k a_1$. The empty room bounded by $a_k b_k$ is not visible from any other vertex but a_k and b_k, hence there must be a guard at one of these vertices. Combined with the previous argument, the resulting piecewise-convex n-gon requires $2(k-1) + 1 = 2k - 1 = \lfloor \frac{2n}{3} \rfloor$ vertex guards.

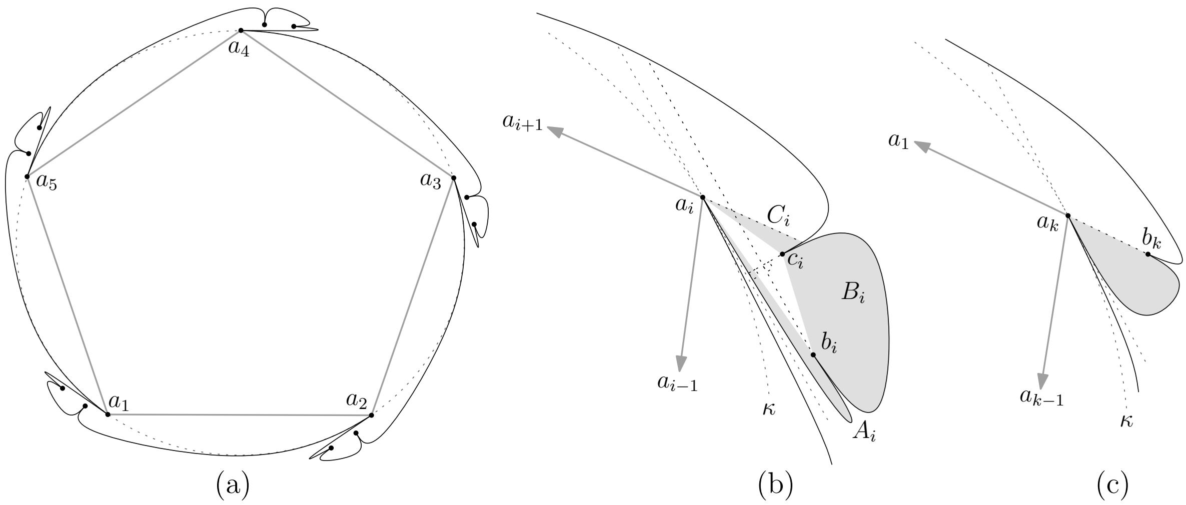

Figure 12: (a) Our lower bound construction for $n = 15$; (b) a pattern with 3 vertices requiring two vertex guards; (c) a pattern with 2 vertices requiring 1 vertex guard.

Theorem 11. *For every integer $n \geq 2$, there is a piecewise-convex polygon with n vertices that cannot be monitored by fewer than $\lfloor \frac{2n}{3} \rfloor$ vertex guards.*

The lower bound for point guards can be established much more easily. Consider the n-vertex piecewise-convex polygon C shown in Fig. 11(c). It can be readily seen that we need one point guard for any two consecutive prongs of C; since C contains n prongs, a minimum of $\lceil \frac{n}{2} \rceil$ point guards are necessary for monitoring C.

Theorem 12. *For every integer $n \geq 2$, there is a piecewise-convex polygon with n vertices that cannot be monitored by fewer than $\lceil \frac{n}{2} \rceil$ point guards.*

4. Piecewise-concave polygons

In this section we address the problem of finding the minimum number of guards that can jointly monitor any piecewise-concave polygon with $n \geq 3$ vertices. Monitoring a piecewise-concave polygon with vertex guards may be impossible even for very simple configurations (see Fig. 14(a)). In particular we prove the following:

Theorem 13. *For every integer $n \geq 3$, the minimum number of point guards that can jointly monitor any piecewise-concave polygon with n vertices is $2n - 4$.*

To prove the sufficiency of $2n-4$ point guards we adapt a technique due to Fejes Tóth [28] to our case. Fejes Tóth proved that the free space around n pairwise disjoint compact convex sets can be monitored by $\max(2n, 4n-7)$ point guards. The edges of a piecewise-concave polygon P are the boundaries of compact convex sets in the plane; these sets however are not necessarily disjoint. The proof in [28] is based on a tessellation of the free space; here we compute a tessellation restricted to P.

Proof. We are given a piecewise-concave polygon P with n vertices and n concave arcs (see Fig. 13). Successively replace each concave arc a_i by another concave arc κ_i with the same endpoints that decreases the polygon maximally. Formally, we construct a sequence of piecewise-concave polygons $P_0 = P, P_1, P_2, \ldots, P_n$. For $i = 1, 2, \ldots, n$, we obtain P_i from P_{i-1} by replacing the concave arc a_i by a concave arc κ_i between v_i and v_{i+1} such that P_i is minimal (for containment), that is, there is no piecewise-concave polygon P_i' with n vertices such that $P_i' \subsetneq P_i$ and the boundary of P_i' differs from P_i only in the edge between v_i and v_{i+1}. Let $\mathcal{K} = \{\kappa_i : 1 \leq i \leq n\}$.

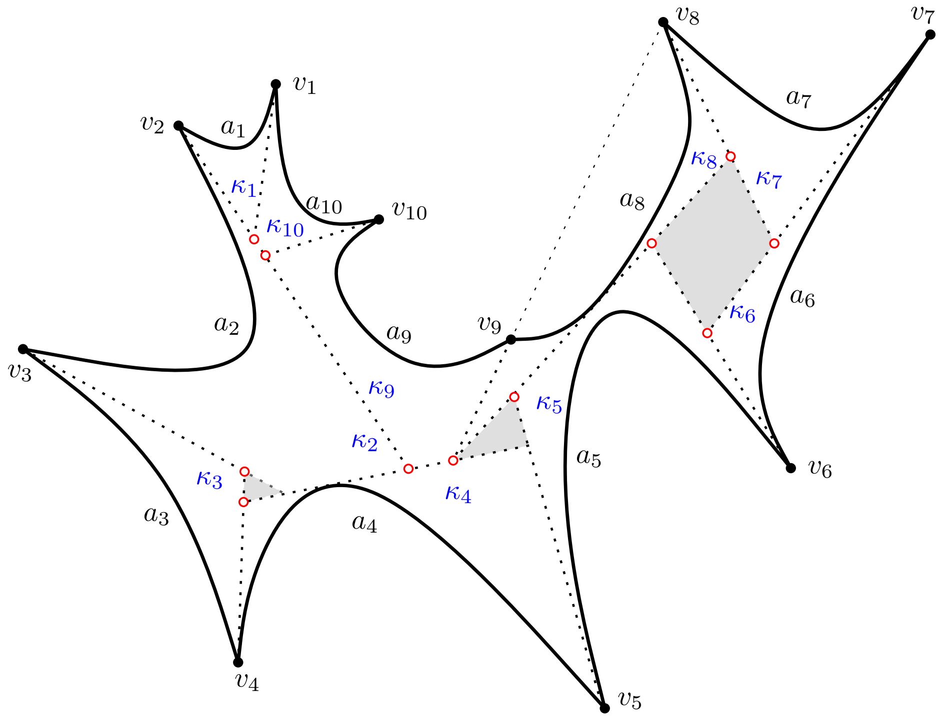

Figure 13: A piecewise-concave 10-gon, the concave arcs κ_i for $i = 1, 2, \ldots, 10$, the resulting tessellation into 10 crescents and 3 convex gaps, and the locations of 11 point guards.

Let us call the region bounded by a_i and κ_i the *crescent* of edge a_i. Fejes Tóth proved that each arc κ_i is a polygonal path, and the arcs κ_i partition P into n crescents (one for each edge) and convex polygons, which he called *gaps*. The crescents and convex gaps are the *faces* of a *tessellation* T of P. A *vertex* of this tessellation is a point incident to at least three faces. Note that every vertex of a gap is a vertex of T. Fejes Tóth showed that we can monitor all crescents and all gaps (hence, the entire P) if we place point guards as follows:

- place a point guard at every vertex of T incident to at least 3 crescents;
- place two guards at two arbitrary vertices of every triangular gap;

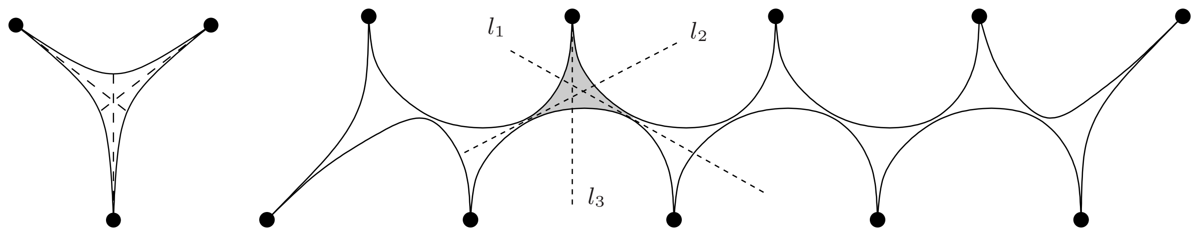

Figure 14: (a) A piecewise-concave polygon P that cannot be monitored solely by vertex guards. Two consecutive edges of P have a common tangent at the common vertex and as a result the three vertices of P see only the points along the dashed segments. (b) A piecewise-concave polygon P that requires $2n-4$ point guards in order to be monitored.

- place a guard at each vertex of every gap with 4 or more vertices.

Construct, now, a planar graph Γ with vertex set $\mathcal{K}$. Two vertices κ_i and κ_j of Γ are connected via an edge if κ_i and κ_j are adjacent. The graph Γ is a planar graph combinatorially equivalent to an outerplanar graph R with n vertices. The edges of Γ connecting consecutive arcs κ_i, κ_{i+1}, $1 \leq i \leq n$, correspond to the boundary edges of R, whereas all other edges of Γ correspond to diagonals in R. Every gap of the tessellation incident to k crescents corresponds to bounded k-gon face of R. Every ordinary vertex of the tessellation which is incident to k crescents but no gap corresponds to a bounded k-gon face of R.

Denote by d_k the number of k-gon faces of R. Every triangular face of R corresponds to at most 2 point guards, and every k-gon face, $k \geq 4$ corresponds to at most k point guards. The total number of point guards is $2d_3 + \sum_{k=4}^{n} kd_k$. This quantity does not decrease if we subdivide a bounded face with $k \geq 4$ vertices into $k-2$ triangles. In the worst case, all faces are triangles. An outerplanar graph with n vertices has at most $n-2$ triangular faces, hence the number of point guards is bounded by $2(n-2)$.

To prove the necessity, refer to the piecewise-concave polygon P in Fig. 14(b). Each one of the pseudo-triangular regions in the interior of P requires exactly two point guards in order to be monitored. Consider for example the pseudo-triangle τ shown in gray in Fig. 14(b). We need one point along each one of the lines l_1, l_2 and l_3 in order to monitor the regions near the corners of τ, which implies that we need at least two points in order to monitor τ (two out of the three points of intersection of the lines l_1, l_2 and l_3). The number of such pseudo-triangular regions is exactly $n-2$, thus we need a total of $2n-4$ point guards to monitor P. $\square$

5. Discussion and open problems

Every piecewise-convex polygon with $n \geq 3$ vertices can be monitored by $\lfloor \frac{2n}{3} \rfloor$ vertex guards, which is best possible. Furthermore, we presented an $O(n \log n)$ time and $O(n)$ space algorithm for computing a vertex guard set of size at most $\lfloor \frac{2n}{3} \rfloor$. Every piecewise-concave polygons with $n \geq 3$ vertices can be monitored by $2n-4$ point guards, which is also best possible. We have not found a piecewise-convex polygon that requires more than $\lceil \frac{n}{2} \rceil$ point guards. Closing the gap between the upper and lower bounds, for the case of point guards, remains an open problem.

Beyond the two classes of polygons considered in this paper, it is straightforward to prove the following results (the details are available in a preliminary version of this paper [32]):

1. Given a *monotone* piecewise-convex polygon P with n vertices (i.e., a piecewise-convex polygon P for which there exists a line L such that any line $L^{\perp}$ perpendicular to L intersects the boundary of P at most twice), $\lfloor \frac{n}{2} \rfloor + 1$ vertex (resp., $\lfloor \frac{n}{2} \rfloor$ point) guards are always sufficient and sometimes necessary in order to monitor P.
2. Given a locally convex polygon P with n vertices, n point guards are always sufficient and sometimes necessary in order to monitor P. In particular, the n vertices of P are a guard set for P.

3. Given a *monotone* locally convex polygon (defined in direct analogy to monotone piecewise-convex polygons), $\lfloor \frac{n}{2} \rfloor + 1$ vertex or point guards are always sufficient and sometimes necessary.
4. Finally, there exist *general polygons* that cannot be monitored with a finite number of point guards.

Karavelas [33, 34] has recently shown that every piecewise-convex polygon with n vertices can be monitored by $\lfloor \frac{2n+1}{5} \rfloor$ edge guards or by $\lfloor \frac{n+1}{3} \rfloor$ guards each of which is either an edge or a straight-line diagonal of the polygon; whereas $\lfloor \frac{n}{3} \rfloor$ edges or straight-line diagonals are sometimes necessary. Other types of guarding problems have been studied in the literature, which either differ on the type of guards, the topology of the polygons considered (e.g., polygons with holes) or the guarding model; see the book by O'Rourke [25], the surveys by Shermer [26] and by Urrutia [27] for an extensive list of the variations of the art gallery problem with respect to the types of guards or the guarding model. It would be interesting to extend these results to the families of curvilinear polygons presented in this paper.

Acknowledgements

The authors wish to thank Ioannis Z. Emiris, Hazel Everett and Günter Rote for useful discussions about the problem. We are also thankful to Valentin Polishchuk as well as an anonymous referee for their comments and suggestions. 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] R. Kuc, M. Siegel, Efficient representation of reflecting structures for a sonar navigation model, in: Proc. 1987 IEEE Int. Conf. Robotics and Automation, 1987, pp. 1916–1923.
[2] S. Xie, T. Calvert, B. Bhattacharya, Planning views for the incremental construction of body models, in: Proc. Int. Conf. Pattern Recongition, 1986, pp. 154–157.
[3] T. Lozano-Pérez, M. A. Wesley, An algorithm for planning collision-free paths among polyhedral obstacles, Commun. ACM 22 (10) (1979) 560–570.
[4] J. S. B. Mitchell, An algorithmic approach to some problems in terrain navigation, in: D. Kapur, J. Mundy (Eds.), Geometric Reasoning, MIT Press, Cambridge, MA, 1989, pp. 171–201.
[5] J. Stenstrom, C. Connolly, Building wire frames for multiple range views, in: Proc. 1986 IEEE Conf. Robotics and Automation, 1986, pp. 615–650.
[6] M. Yachida, 3-D data acquisition by multiple views, in: Robotics Research: Third Int. Symp., 1986, pp. 11–18.
[7] D. Avis, H. ElGindy, A combinatorial approach to polygon similarity, IEEE Trans. Inform. Theory IT-2 (1983) 148–150.
[8] G. T. Toussaint, Pattern recognition and geometrical complexity, in: Proc. 5th IEEE Internat. Conf. Pattern Recogn., 1980, pp. 1324–1347.
[9] M. McKenna, Worst-case optimal hidden-surface removal, ACM Trans. Graph. 6 (1987) 19–28.
[10] B. Chazelle, J. Incerpi, Triangulation and shape-complexity, ACM Trans. Graph. 3 (2) (1984) 135–152.
[11] W. Bronsvoort, Boundary evaluation and direct display of CSG models, Computer-Aided Design 20 (1988) 416–419.
[12] K. Eo, C. Kyung, Hybrid shadow testing scheme for ray tracing, Computer-Aided Design 21 (1989) 38–48.
[13] D. Eppstein, M. T. Goodrich, N. Sitchinava, Guard placement for efficient point-in-polygon proofs, in: Proc. 23rd Annu. ACM Sympos. Comput. Geom., ACM Press, 2007, pp. 27–36.
[14] D. P. Dobkin, D. L. Souvaine, Computational geometry in a curved world, Algorithmica 5 (1990) 421–457.
[15] J.-D. Boissonnat, M. Teillaud (Eds.), Effective Computational Geometry for Curves and Surfaces, Mathematics and Visualization, Springer, 2007.
[16] V. Chvátal, A combinatorial theorem in plane geometry, J. Combin. Theory Ser. B 18 (1975) 39–41.
[17] S. Fisk, A short proof of Chvátal's watchman theorem, J. Combin. Theory Ser. B 24 (1978) 374.
[18] B. Chazelle, Triangulating a simple polygon in linear time, Discrete Comput. Geom. 6 (5) (1991) 485–524.
[19] D. Lee, A. Lin, Computational complexity of art gallery problems, IEEE Trans. Inform. Theory 32 (2) (1986) 276–282.
[20] A. Aggarwal, The art gallery problem: Its variations, applications, and algorithmic aspects, Ph.D. thesis, Dept. of Comput. Sci., Johns Hopkins University, Baltimore, MD (1984).
[21] J. Kahn, M. M. Klawe, D. Kleitman, Traditional galleries require fewer watchmen, SIAM J. Algebraic Discrete Methods 4 (1983) 194–206.
[22] J.-R. Sack, Rectilinear computational geometry, Ph.D. thesis, School Comput. Sci., Carleton Univ., Ottawa, ON, report SCS-TR-54 (1984).
[23] A. Lubiw, Decomposing polygonal regions into convex quadrilaterals, in: Proc. 1st ACM Sympos. Comput. Geom., ACM Press, 1985, pp. 97–106.
[24] H. Edelsbrunner, J. O'Rourke, E. Welzl, Stationing guards in rectilinear art galleries, Comput. Vision Graph. Image Process. 27 (1984) 167–176.

[25] J. O'Rourke, Art Gallery Theorems and Algorithms, The International Series of Monographs on Computer Science, Oxford University Press, New York, NY, 1987.
[26] T. C. Shermer, Recent results in art galleries, Proc. IEEE 80 (9) (1992) 1384–1399.
[27] J. Urrutia, Art gallery and illumination problems, in: J.-R. Sack, J. Urrutia (Eds.), Handbook of Computational Geometry, North-Holland, 2000, pp. 973–1027.
[28] L. Fejes Tóth, Illumination of convex discs, Acta Math. Acad. Sci. Hungar. 29 (3–4) (1977) 355–360.
[29] G. Meisters, Polygons have ears, Amer. Math. Monthly 82 (1975) 648–651.
[30] D. T. Lee, F. P. Preparata, Location of a point in a planar subdivision and its applications, SIAM J. Comput. 6 (3) (1977) 594–606.
[31] M. de Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf, Computational Geometry: Algorithms and Applications, 2nd Edition, Springer-Verlag, Berlin, Germany, 2000.
[32] M. I. Karavelas, E. P. Tsigaridas, Guarding curvilinear art galleries with vertex or point guards, `arXiv:0802.2594v1 [cs.CG]` (February 2008).
[33] M. I. Karavelas, Guarding curvilinear art galleries with edge or mobile guards via 2-dominance of triangulation graphs, `arXiv:0802.1361v1 [cs.CG]` (2008).
[34] M. I. Karavelas, Guarding curvilinear art galleries with edge or mobile guards, in: Proc. ACM Sympos. Solid and Physical Modeling, ACM Press, 2008, pp. 339–345.