Altitude Terrain Guarding and Guarding Uni-Monotone Polygons

Stephan Friedrichs[1,2], Valentin Polishchuk[3], and Christiane Schmidt[3]

1 **Max Planck Institute for Informatics, Saarbrücken, Germany**
sfriedri@mpi-inf.mpg.de
2 **Saarbrücken Graduate School of Computer Science, Saarbrücken, Germany**
3 **Communications and Transport Systems, ITN, Linköping University, Sweden,**
{valentin.polishchuk,christiane.schmidt}@liu.se

Abstract

We show that the problem of guarding an x-monotone terrain from an altitude line and the problem of guarding a uni-monotone polygon are equivalent. We present a polynomial time algorithm for both problems, and show that the cardinality of a minimum guard set and the cardinality of a maximum witness set coincide. Thus, uni-monotone polygons are perfect.

1 Introduction

Both the Art Gallery Problem (AGP) and the 1.5D Terrain Guarding Problem (TGP) are well known problems in Computational Geometry. We are given a polygon P (AGP) or an x-monotone chain T of line segments in $\mathbb{R}^2$ (1.5D TGP) and need to place a minimum number of point-shaped guards in P or on T, such that they cover all of P or T, respectively. Both problems have been shown to be NP-hard: Krohn and Nilsson [3] proved the AGP to be hard even for monotone polygons, and King and Krohn [2] established the NP-hardness of both the discrete and the continuous TGP (with guards restricted to the terrain vertices or guards located anywhere on the terrain).

The problem of guarding a uni-monotone polygon (an x-monotone polygon with a single horizontal segment as one of its two chains) and the problem of guarding a terrain with guards placed on a horizontal line above the terrain appear to be problems somewhere between the 1.5D TGP and the AGP in monotone polygons. We show that, surprisingly, both problems allow for a polynomial time algorithm: a simple sweep.

Moreover, we are able to construct a maximum witness set of the same cardinality as the minimum guard set for uni-monotone polygons. Hence, we establish the first non-trivial class of perfect polygons (earlier only proven for "rectilinear"[5] and "staircase" visibility [4]).

One application of guarding a terrain with guards placed on a horizontal line above the terrain, the Altitude Terrain Guarding Problem (ATGP), comes from the idea of using drones to surveil a complete geographical area. Usually, these drones will not be able to fly arbitrarily high, which motivates to cap the allowed height for guards (and without this restriction a single sufficiently high guard above the terrain will be enough). Of course, eventually we are interested in working in two dimensions and a height, the 2.5D ATGP—one dimension and height is a natural starting point for this.

2 Notation and Preliminaries

A polygon P is *x-monotone* if any line orthogonal to the x-axis has a simply connected intersection with P. Its leftmost and rightmost vertex split the boundary in two x-monotone polygonal chains. A *uni-monotone polygon* P is an x-monotone polygon, such that one of

34th European Workshop on Computational Geometry, Berlin, Germany, March 21–23, 2018.
This is an extended abstract of a presentation given at EuroCG'18. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear eventually in more final form at a conference with formal proceedings and/or in a journal.

its two chains is a single horizontal segment. W.l.o.g. we will assume the single horizontal segment to be the upper chain for the remainder of this paper; we denote this segment by $\mathcal{H}$. The lower chain of P, $LC(P)$, is defined by its *vertices* $V(P) = \{v_1, \ldots, v_n\}$ and has *edges* $E(P) = \{e_1, \ldots, e_{n-1}\}$ with $e_i = \overline{v_i v_{i+1}}$. Due to uni-monotonicity the vertices of P are totally ordered w.r.t. their x-coordinates. A point $p \in P$ *sees* or *covers* $q \in P$ if and only if $\overline{pq}$ is fully contained in P. $\mathcal{V}_P(p)$ is the *visibility polygon* (VP) of p in P with $\mathcal{V}_P(p) := \{q \in P \mid p \text{ sees } q\}$. For $G \subset P$ we abbreviate $\mathcal{V}_P(G) := \bigcup_{g \in G} \mathcal{V}_P(g)$.

A *terrain* T is an x-monotone chain of line segments in $\mathbb{R}^2$ defined by its *vertices* $V(T) = \{v_1, \ldots, v_n\}$ that has *edges* $E(T) = \{e_1, \ldots, e_{n-1}\}$ with $e_i = \overline{v_i v_{i+1}}$; and $\text{int}(e_i) := e_i \setminus \{v_i, v_{i+1}\}$ is e_i's *interior.* Due to monotonicity the points on T are totally ordered w.r.t. their x-coordinates. For $p, q \in T$, we write $p \leq q$ ($p < q$) if p is (strictly) left of q.

An *altitude line* $\mathcal{A}$ at height a_h for a terrain T is a horizontal line located a_h above the lowest (y-)coordinate of all vertices of T, with the leftmost point vertically above v_1 and the rightmost point vertically above v_n. For this abstract we consider only the case where the altitude line lies completely above T. The points on $\mathcal{A}$ are totally ordered as well w.r.t. their x-coordinates, and we adapt the same notation as for two points on T. A point $p \in \mathcal{A}$ *sees* or *covers* $q \in T$ if and only if $\overline{pq}$ is nowhere below T (i.e. $\overline{pq}$ lies on or above T). $\mathcal{V}_T(p)$ is the *visibility region* of p with $\mathcal{V}_T(p) := \{q \in T \mid p \text{ sees } q\}$. For $G \subseteq \mathcal{A}$ we abbreviate $\mathcal{V}_T(G) := \bigcup_{g \in G} \mathcal{V}_T(g)$. We also define the *visibility region* for $p \in T$: $\mathcal{V}_T(p) := \{q \in \mathcal{A} \mid p \text{ sees } q\}$.

For an edge $e \in P$ or $e \in T$ the *strong VP* (*weak VP*) is the set of points that see all of e (at least one point of e): $\mathcal{V}^s_P(e) := \{p \in P : \forall q \in e \; p \text{ sees } q\}$ and $\mathcal{V}^s_T(e) := \{p \in \mathcal{A} : \forall q \in e \; p \text{ sees } q\}$ ($\mathcal{V}^w_P(e) := \{p \in P : \exists q \in e \; p \text{ sees } q\}$ and $\mathcal{V}^w_T(e) := \{p \in \mathcal{A} : \exists q \in e \; p \text{ sees } q\}$).

▶ **Definition 2.1** (Altitude Terrain Guarding Problem)**.** In the *Altitude Terrain Guarding Problem (ATGP)*, abbreviated $\text{ATGP}(T, \mathcal{A})$, we are given a terrain T and an altitude line $\mathcal{A}$. A guard set $G \subset \mathcal{A}$ is optimal w.r.t. $\text{ATGP}(T, \mathcal{A})$ if G is feasible, that is, $T \subseteq \mathcal{V}_T(G)$, and $|G| = \text{OPT}(T, \mathcal{A}) := \min\{|C| \mid C \subseteq \mathcal{A} \text{ is feasible w.r.t. ATGP}(T, \mathcal{A})\}$.

▶ **Definition 2.2** (Art Gallery Problem)**.** In the *Art Gallery Problem (AGP)*, abbreviated $\text{AGP}(G, W)$, we are given a polygon P and sets of guard candidates and points to cover $G, W \subseteq P$. A guard set $C \subseteq G$ is optimal w.r.t. $\text{AGP}(G, W)$ if C is feasible, that is, $W \subseteq \mathcal{V}_P(C)$, and $|C| = \text{OPT}(G, W) := \min\{|C| \mid C \subseteq G \text{ is feasible w.r.t. AGP}(G, W)\}$. In general, we want to solve the AGP for $G = P$ and $W = P$, that is, $\text{AGP}(P, P)$.

A set $W \subset P$ ($W \subset T$) is a *witness set* if $\forall\, w_i \neq w_j \in W$ we have $\mathcal{V}_P(w_i) \cap \mathcal{V}_P(w_j) = \emptyset$. A polygon class $\mathcal{P}$ is *perfect* if the cardinality of an optimum guard set and the cardinality of a maximum witness set coincide for all polygons $P \in \mathcal{P}$.

▶ **Lemma 2.3.** *Let P be a uni-monotone polygon, with guard set G. Then there exists a guard set $G^{\mathcal{H}}$ with $|G| = |G^{\mathcal{H}}|$ and $g \in \mathcal{H}\ \forall g \in G^{\mathcal{H}}$. That is, if we want to solve the AGP for a uni-monotone polygon, w.l.o.g. we can restrict our guards to be located on $\mathcal{H}$.*

Proof. Let G be an optimal guard set. Consider a point $p \in P$, there exists a guard $g \in G$ that covers p. Let $g^{\mathcal{H}}$ be the point located vertically above g on $\mathcal{H}$. Because of P being uni-monotone the triangle $\Delta(g, p, g^{\mathcal{H}})$ must be empty, hence, also $g^{\mathcal{H}}$ sees p. ◀

An analogous proof shows that we can always place guards on the altitude line $\mathcal{A}$ even if we would be allowed to place them anywhere between the terrain T and $\mathcal{A}$.

▶ **Lemma 2.4.** *Let P be a uni-monotone polygon, G a guard set with $g \in \mathcal{H}\ \forall g \in G$ that covers $LC(P)$, that is, $LC(P) \subset \mathcal{V}_P(G)$. Then G covers all of P, that is, $P \subseteq \mathcal{V}_P(G)$.*

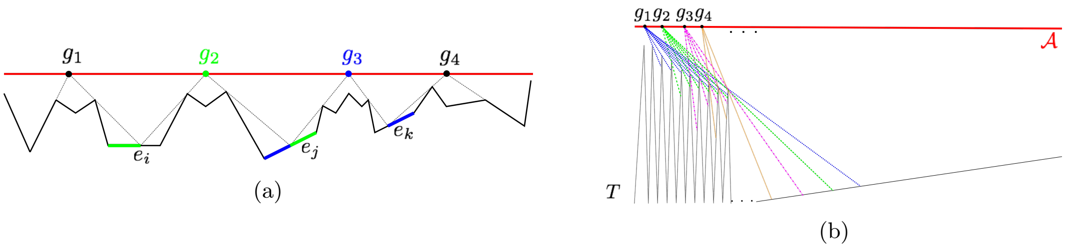

Figure 1 (a) Terrain T and altitude line $\mathcal{A}$ is shown in black and red, resp.. $g_1, \ldots, g_4$ are an optimal guard cover. g_2 and g_3 both cover a critical edge both to their left and to their right. (b) Example: each of the $O(n)$ guards needs to shoot $O(n)$ (colored) rays.

Proof. Assume there is a point $p \in P$, $p \notin LC(P)$ with $p \notin \mathcal{V}_P(G)$. Consider the point p^{LC}, which is located vertically below p on $LC(P)$. Let $g \in G$ be a guard that sees p^{LC}. $LC(P)$ does not intersect the line $\overline{p^{LC}g}$, and because P is uni-monotone the triangle $\Delta(g, p, p^{LC})$ is empty, hence, g sees p; a contradiction. ◀

Consequently, the ATGP and the AGP for uni-monotone polygons are equivalent; we will only refer to the ATGP in the remainder of this paper, with the understanding that all our results can be applied directly to the AGP for uni-monotone polygons.

▶ **Lemma 2.5.** *Let $g \in \mathcal{A}, p \in T, g < p$. If $p \notin \mathcal{V}_T(g)$ then $\forall g' < g, g' \in \mathcal{A} : p \notin \mathcal{V}_T(g')$.*

Before we present our algorithm, we observe that the ATGP is intrinsically different from TGP. We repeat (and extend) a definition from [1]: For a feasible guard cover C of T ($C \subset T$ for TGP and $C \subset \mathcal{A}$ for ATGP), an edge $e \in E$ is *critical w.r.t.* $g \in C$ if $C \setminus \{g\}$ covers some part of, but not all of int(e). If e is critical w.r.t. some $g \in C$, we call e *critical edge* (e is critical iff more than one guard is responsible for covering its interior). $g \in C$ is a *left-guard (right-guard)* of $e_i \in E$ if $g < v_i$ ($v_{i+1} < g$) and e_i is critical w.r.t. g. We call g a *left-guard (right-guard)* if it is a left-guard (right-guard) of some $e \in E$.

▶ Observation 2.6. For the TGP we have: Let C be finite and cover T, then no $g \in C \setminus V(T)$ is both a left- and a right-guard, that is, no guard that is not on a vertex is responsible to cover critical edges to its left and right, see Friedrichs et al. [1]. However, for the ATGP, any guard g on $\mathcal{A}$ may be responsible to cover critical edges both to its left and to its right, that is, guards may be both a left- and a right-guard, see Figure 1(a).

3 Sweep Algorithm

Our algorithm is a sweep, and informally it can be described as follows (the pseudocode for our algorithm, using definitions from Section 3.1, is presented in Algorithm 1):

- We start with an empty set of guards, $G = \emptyset$, and at the leftmost point of $\mathcal{A}$; all edges $E(T)$ are completely unseen.
- We sweep along $\mathcal{A}$ from left to right and place a guard g_i whenever we could no longer see all of an edge e' if we would move more to the right.
- We compute the visibility polygon of g_i, $\mathcal{V}_T(g_i)$, and for each edge $e = \{v, w\}$ partially seen by g_i, we split the edge, and only keep the open interval that is not yet guarded.
- Thus, whenever we insert a new guard g_i we have a new set of “edges” $E_i(T)$ that are still completely unseen, and $\forall f \in E_i(T)$ we have $f \subseteq e \in E(T)$.
- We continue placing new guards until $T \subseteq \mathcal{V}_T(G)$.
- As we can define a witness set of $|G|$ our guard set is optimal: we place a point witness on e' at the point p we would lose coverage of, if we had not placed guard g_i.

3.1 How to Split the Partly Seen Edges

For each edge in the initial set of edges, $e \in E(T)$, we need to determine the point p_e^c that closes the interval on $\mathcal{A}$ from which all of e is visible. We denote the set of points $p_e^c\ \forall e \in E(T)$ as the set of closing points $\mathcal{C}$, that is, $\mathcal{C} = \cup_{e \in E(T)}\{p_e^c \in \mathcal{A} : (e \subseteq \mathcal{V}_T(p_e^c)) \wedge (e \nsubseteq \mathcal{V}_T(p)\ \forall p > p_e^c,\ p \in \mathcal{A})\}$. The points in $\mathcal{C}$ are the rightmost points on $\mathcal{A}$ in the strong visibility polygon of the edge e, for all edges. Analogously, we define the set of opening points $\mathcal{O}$: $\mathcal{O} = \cup_{e \in E(T)}\{p_e^o \in \mathcal{A} : (e \subseteq \mathcal{V}_T(p_e^o)) \wedge (e \nsubseteq \mathcal{V}_T(p)\ \forall p < p_e^o,\ p \in \mathcal{A})\}$. For each edge e the point in $\mathcal{O}$ is the leftmost point on $\mathcal{A}$ in the strong visibility polygon of e.

Moreover, whenever we place a new guard, we need to split partly seen edges to obtain the new, completely unseen, possibly open, interval, and determine the point on $\mathcal{A}$ where we would lose coverage of this edge (interval). That is, whenever we split an edge we need to add the appropriate point to $\mathcal{C}$.

To be able to easily identify whether an edge e of the terrain needs to be split due to a new guard g, we define the set of "soft openings" $\mathcal{S}$: the leftmost point on $\mathcal{A}$ in the weak visibility polygon of e (if g is to the right of this point—and to the left of the closing point—it can see at least parts of e). We define $\mathcal{S} = \cup_{e \in E(T)}\{p_e^s \in \mathcal{A} : (\exists q \in e, q \in \mathcal{V}_T(p_e^s)) \wedge (\nexists q \in e,\ q \in \mathcal{V}_T(p)\ \forall p < p_e^s, p \in \mathcal{A})\}$.

So, how do we preprocess our terrain such that we can easily identify the point on $\mathcal{A}$ that we need to add to $\mathcal{C}$ when we split an edge? We make an initial sweep from the rightmost to the leftmost vertex; for each vertex we shoot a ray to all other vertices to its left and mark the points, *mark points*, where these rays hit the edges of the terrain. This leaves us with $O(n^2)$ preprocessed intervals. For each mark point m we store the rightmost of the two terrain vertices that defined the ray hitting the terrain at m, let this vertex be denoted by v_m. For each edge $e_j = \{v_j, v_{j+1}\}$ with v_{j+1} convex, this includes v_{j+1} as a mark point.

Whenever the placement of a guard g splits an edge e such that the open interval $e' \subset e$ is not yet guarded, see for example Figure 2(a), we identify the mark, $m_{e'}$ to the right of e' and shoot a ray r from the right endpoint of e' through $v_{m_{e'}}$ (the one we stored with $m_{e'}$). The intersection point of r and $\mathcal{A}$ defines our new closing point $p_{e'}^c$, see Figure 2(b).

3.2 Minimum Guard Set and Perfect Polygons

▶ **Lemma 3.1.** *The set G output by Algorithm 1 is feasible, that is, $T \subseteq \mathcal{V}_T(G)$.*

Proof. Assume there is a point $p \in T$ with $p \notin \mathcal{V}_T(G)$. $p \in e$ for some edge $e \in E(T)$. As p is not covered, there exists no guard in G in the interval $[p_e^o, p_e^c]$ on $\mathcal{A}$. Thus, p_e^c is never the

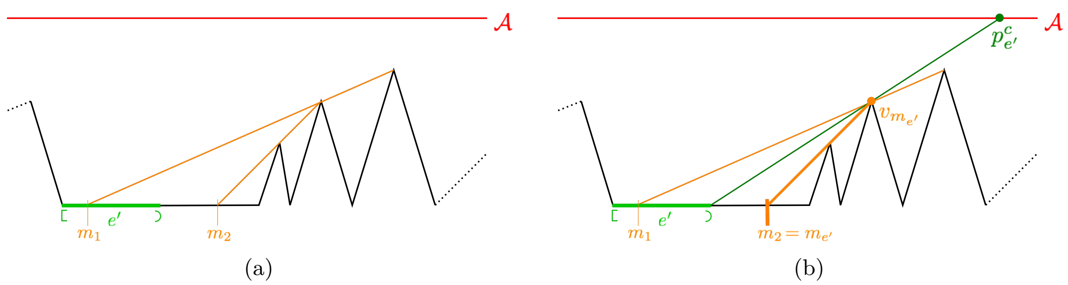

Figure 2 Terrain T and altitude line $\mathcal{A}$ is shown in black and red, resp.. The orange lines show the rays from the preprocessing step, their intersection points with the terrain define the mark points. Assume the open interval e', shown in light green, is still unseen. To identify the closing point for e' we identify the mark to the right of e', $m_{e'}$, and shoot a ray, shown in dark green, from the right end point of e' through $v_{m_{e'}}$. The intersection point of r and $\mathcal{A}$ defines our new closing point $p_{e'}^c$.

```
   INPUT  : Terrain T, altitude line 𝒜, its leftmost point a, sets 𝒞, 𝒪, 𝒮 of closing, opening, and soft
            opening points for all edges in T, all ordered from left to right.
   OUTPUT : An optimal guard set G.
1  E_g = E(T)                                         // set of edges that still need to be guarded
2  i := 1
3  g_0 := a                                 // the point on 𝒜 before the first guard is a
4  while E_g ≠ ∅                                   // as long as there are still unseen edges
5  do
6  |  1. Sweep right from g_{i−1} along 𝒜 until the first closing point c ∈ 𝒞 is hit
7  |  2. Place g_i on c, G = G ∪ {g_i}, i := i + 1
8  |  3. for all e ∈ E_g                                          // g_i ≯ p_e^c by construction
9  |  do
10 |  |  if p_e^o ≤ g_i ≤ p_e^c then
11 |  |  |  E_g = E_g \ {e}                        // if all of e is seen, delete it from E_g
12 |  |  |  𝒞 = 𝒞 \ {p_e^c}               // and delete the closing point from the event queue
13 |  |  else if g_i < p_e^o then
14 |  |  |  if p_e^s ≤ g_i                           // if g_i can see the right point of e
15 |  |  |  then
16 |  |  |  |  Shoot a visibility ray from g_i onto e, let the intersection point be r_e  // all points on e
   |  |  |  |  to the right of r_e (incl. r_e) are seen
17 |  |  |  |  Identify the mark m_e immediately to the right of r_e on e
18 |  |  |  |  Shoot a ray r from r_e through v_{m_e}
19 |  |  |  |  Let p_{e'}^c be the intersection point of r and 𝒜  // p_{e'}^c is the closing point for the still
   |  |  |  |  unseen interval e' ⊂ e
20 |  |  |  |  𝒞 = 𝒞 ∪ {p_{e'}^c} \ {p_e^c}
21 |  |  |  |  Sort 𝒞
22 |  |  |  |  E_g = E_g ∪ {e'} \ {e}
```

Algorithm 1: Optimal Guard Set for ATGP

event point that defines the placement of a guard in lines 6,7 of Algorithm 1. Moreover, as $\nexists g_i : p_e^o \leq g_i \leq p_e^c$, e is never completely deleted from E_g in lines 10–12. Consequently, for some i we have $g_i < p_e^o$ and $p_e^s \leq g_i$ (lines 14–22). As $p \notin \mathcal{V}_T(G)$, we have $p \in e' \subset e$.

Again, because $p \notin \mathcal{V}_T(G)$, $\nexists g_j \in [p_e^o, p_{e'}^c] \subset \mathcal{A}$, $j \geq i$. Due to line 6 no guard may be placed to the left of $p_{e'}^c$, hence, there is no guard placed in $[p_e^o, b]$ (b being the right end point of $\mathcal{A}$). So, e' is never deleted from E_g, a contradiction to G being the output of Alg. 1. ◀

▶ **Theorem 3.2.** *The set G output by Algorithm 1 is optimal.*

Proof. To show that G is optimal, we need to show that G is feasible and that G is minimum. Feasibility follows from Lemma 3.1, it remains to show that G is minimum. Given a witness set W and a guard set G, $|W| \leq |G|$ holds. Thus, if we can find a witness set W with $|W| = |G|$, we can show that G is minimum. We will place a witness for each guard Algorithm 1 places. First, we need an auxiliary lemma (and omit the proof):

▶ **Lemma 3.3.** *Let $c \in \mathcal{C}$ be the closing point for a complete edge (and not just an edge interval) in line 6 of Algorithm 1 that enforces the placement of a guard g_i. Then there exists an edge $e_j = \{v_j, v_{j+1}\} \in E(T)$ for which c is the closing point, such that v_{j+1} is a reflex vertex, and v_j is a convex vertex.*

Now we can define our witness set: we consider the edges or edge intervals, which define the closing point $c \in \mathcal{C}$ that leads to a placement of guard g_i in lines 6,7 of Algorithm 1.

If c is defined by some complete edge $e_j \in E(T)$, let $E_c \subseteq E_g$ be the set of edges for which c is the closing point. We pick the rightmost edge $e_j \in E_c$ such that v_j is a convex vertex and v_{j+1} is a reflex vertex, which exists by Lemma 3.3, and choose $w_i = v_j$.

Otherwise, that is, if c is only defined by edge intervals, we pick the rightmost such edge interval $e' \subset e_j$. Then $e' = [v_j, q)$ for some point $q \in e_j, q \neq v_{j+1}$, and we place a witness at q^ε, a point ε_i to the left of q on T: $w_i = q^\varepsilon$. We define $W = \cup_{i=1}^{|G|} w_i$. By definition $|W| = |G|$, and we still need to show that W is indeed a witness set.

Let S_i be the strip of all points with x-coordinates between $x(g_{i-1}) + \varepsilon$ and $x(g_i)$. Let p_T and $p_{\mathcal{A}}$ be the vertical projection of a point p onto T and $\mathcal{A}$, respectively. $S_i = \{p \in \mathbb{R}^2 : (x(g_{i-1}) + \varepsilon \leq x(p) \leq x(g_i)) \wedge (y(p_T) \leq y(p) \leq y(p_{\mathcal{A}}))\}$. We show that $\mathcal{V}_T(w_i) \subseteq S_i \forall i$, hence, $\mathcal{V}_T(w_k) \cap \mathcal{V}_T(w_\ell) = \emptyset \; \forall w_k \neq w_\ell \in W$, which shows that W is a witness set.

If $w_i = v_j$ for an edge $e_j \in E(T)$, $\mathcal{V}_T(w_i)$ contains the guard g_i, but no other guard: If g_{i-1} could see v_j, we would have $\angle(g_{i-1}, v_j, v_j + 1) \leq 180°$ because v_j is a convex vertex, thus, g_{i-1} could see all of e_j, a contradiction to $e_j \in E_g$. Moreover, assume w_i could see some point p with $x(p) \leq x(g_{i-1})$. The terrain does not intersect the line $\overline{w_i p}$, and because the terrain is monotone the triangle $\Delta(w_i, p, g_{i-1})$ would be empty, a contradiction to g_{i-1} not seeing w_i. Hence, no guard $g_j, j < i$ sees w_i; a similar argument can be given for $g_j, j > i$.

If $w_i = q^\varepsilon$ for $e' = [v_j, q)$, $\mathcal{V}_T(w_i)$ contains the guard g_i, but no other guard: If g_{i-1} could see w_i, q would not be the endpoint of the edge interval, a contradiction. Moreover, assume w_i could see some point p with $x(p) \leq x(g_{i-1})$. T does not intersect the line $\overline{w_i p}$, and because T is monotone the triangle $\Delta(w_i, p, g_{i-1})$ would be empty, a contradiction. Thus, again, no guard $g_j, j < i$ sees p (and the case $j > i$ can be shown similarly). ◀

We showed that there exists a maximum witness set $W \subset T$ and a minimum guard set $G \subset \mathcal{A}$ with $|W| = |G|$. By Lemma 2.3 and 2.4 the ATGP and the AGP for uni-monotone polygons are equivalent. Thus, for a uni-monotone polygon P we can find a maximum witness set $W \subset LC(P) \subset P$ and a minimum guard set $G \subset \mathcal{H} \subset P$ with $|W| = |G|$:

▶ **Theorem 3.4.** *Uni-monotone polygons are perfect.*

3.3 Algorithm Runtime

The preprocessing step to compute the mark points costs $O(n^2)$, based on these we can compute the closing points for all edges of the terrain. Similarly, we compute the mark points from the left to compute the opening points (using the left vertex of an edge to shoot the ray) and the soft opening points (using the right vertex of an edge to shoot the ray). Then, whenever we insert a guard (of which we might add $O(n)$), we need to shoot $O(n)$ rays, see Figure 1(b), which altogether costs $O(n^2 \log n)$. Similarly, for each of the intersection points r_e, we need to shoot a ray through v_{m_e}. This gives a total runtime of $O(n^2 \log n)$. In fact, when we stepwise build the convex hull of the terrain vertices from the right and only shoot rays through vertices of this CH, we can reduce the preprocessing step to $O(n)$.

References

1 S. Friedrichs, M. Hemmer, J. King, and C. Schmidt. The continuous 1.5D terrain guarding problem: Discretization, optimal solutions, and PTAS. *JoCG*, 7(1):256–284, 2016.

2 J. King and E. Krohn. Terrain guarding is NP-hard. *SIAM Journal on Computing*, 40(5):1316–1339, 2011.

3 E. Krohn and B. J. Nilsson. The complexity of guarding monotone polygons. In *Proc. of the 24th Canadian Conference on Comp. Geometry*, pages 167–172, 2012.

4 R. Motwani, A. Raghunathan, and H. Saran. Covering orthogonal polygons with star polygons: The perfect graph approach. *J. Comput. Syst. Sci.*, 40(1):19–48, 1990.

5 C. Worman and J. M. Keil. Polygon decomposition and the orthogonal art gallery problem. *Int. J. Comput. Geometry Appl.*, 17(2):105–138, 2007.