On Guarding Orthogonal Polygons with Bounded Treewidth

Therese Biedl and Saeed Mehrabi*

Abstract

There exist many variants of guarding an orthogonal polygon in an orthogonal fashion: sometimes a guard can see an entire rectangle, or along a staircase, or along an orthogonal path with at most k bends. In this paper, we study all these guarding models in the special case of orthogonal polygons that have bounded treewidth in some sense. Exploiting algorithms for graphs of bounded treewidth, we show that the problem of finding the minimum number of guards in these models becomes linear-time solvable in polygons of bounded treewidth.

1 Introduction

In this paper, we study orthogonal variants of the well-known art gallery problem. In the standard art gallery problem, we are given a polygon P and we want to guard P with the minimum number of point guards, where a guard g sees a point p if the line segment $\overline{gp}$ lies entirely inside P. This problem was introduced by Klee in 1973 [15] and has received much attention since. $\lfloor n/3 \rfloor$ guards are always sufficient and sometimes necessary [5], minimizing the number of guards is NP-hard on arbitrary polygons [13], orthogonal polygons [16], and even on simple monotone polygons [12]. The problem is APX-hard on simple polygons [9] and several approximation algorithms have been developed [11, 12].

Since the problem is hard, attention has focused on restricting the type of guards, their visibility or the shape of the polygon. In this paper, we consider several models of "orthogonal visibility", and study orthogonal polygons that have bounded treewidth in some sense. Treewidth (defined in Section 2.1) is normally a parameter of a graph, but we can define it for a polygon P as follows. Obtain the *standard pixelation* of P by extending a horizontal and a vertical ray inwards at every reflex vertex until it hits the boundary of P (see Figure 1). We can interpret this subdivision into rectangles as a planar straight-line graph by placing a vertex at any place incident to at least two segments, and define the treewidth of a polygon P to be the treewidth of the graph of the standard pixelation.

*David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, Canada. {biedl,smehrabi}@uwaterloo.ca

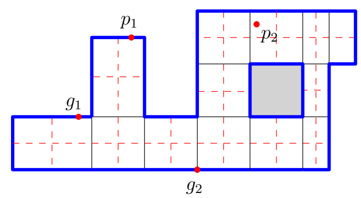

Figure 1: A polygon P with its standard pixelation (black, solid) and its 1-refinement (red, dashed). The gray area indicates a hole.

Motivation. One previously studied special case of the art gallery problem concerns *thin* polygons, defined to be orthogonal polygons for which every vertex of the standard pixelation lies on the boundary of the polygon. Thus a polygon is simple and thin if and only if the standard pixelation is an outer-planar graph. Tomás [17] showed that the (non-orthogonal) art gallery problem is NP-hard even for simple thin polygons if guards must be at vertices of the polygon. Naturally one wonders whether this NP-hardness can be transferred to orthogonal guarding models. This is not true, for example r-guarding (defined below) is polynomial on polygons whose standard pixelation is outer-planar, because it is polynomial on *any* simple polygon [18]. But what can be said about polygons that are "close" to being thin? Since outer-planar graphs have treewidth 2, this motivates the question of polygons where the standard pixelation has bounded treewidth.

The goal of this paper is to solve orthogonal guarding problems for polygons of bounded treewidth. There are many variants of what "orthogonal guarding" might mean; we list below the ones considered in this paper:

- Rectangle-guarding (r-guarding). A point guard g *r-guards* a point p if the minimum axis-aligned rectangle containing g and p is a subset of P.
- Staircase-guarding (s-guarding). A point guard g *s-guards* any point p that can be reached from g by a *staircase*, i.e., an orthogonal path inside P that is both x-monotone and y-monotone.
- Periscope-guarding. A *periscope guard* g can see all points p in which some orthogonal path inside P connects g to p and has at most one bend.

A natural generalization of periscope-guards are k-*periscope guards* in which a point guard g can see all points p that are connected via an orthogonal path inside P with at most k bends. (In contrast to s-guards, monotonicity of the path is not required.)

Another variant is to consider length rather than number of bends. Thus, an L_1-*distance guard* g (for some fixed distance-bound D) can see all points p for which some orthogonal path from g to p inside P has length at most D[1].

- Sliding cameras. Recently there has been much interest in *mobile guards*, where a guard can walk along a line segment inside polygon P, and can see all points that it can see from some point along the line segment. In an orthogonal setting, this type of guards becomes a *sliding camera*, i.e., an axis-aligned line segment s inside P that can see a point p if the perpendicular from p onto s lies inside P.

Related results. In the full version of the paper, we list (numerous) existing results about r-guarding, s-guarding, periscope guarding and sliding cameras. In a nutshell, most of these are NP-hard [4, 8, 10], and some of them can be solved in polynomial time if P has no holes [14, 18]. Of special relevance to the current paper is that r-guarding and guarding with sliding cameras can be solved in linear time for polygons with bounded treewidth [3, 4].

Our results. The main goal of this paper is to solve the s-guarding problem in polygons of bounded treewidth. The method used in [3, 4] does not work for this since s-guards can see along an infinite number of bends. Instead, we develop an entirely different approach. Note that the above guarding-models (except r-guarding) are defined as "there exists an orthogonal path from g to p that satisfies some property". One can argue (see Lemma 1) that we may assume the path to run along edges of the standard pixelation. The guarding problem then becomes the problem of reachability in a directed graph derived from the standard pixelation. This problem is polynomial in graphs of bounded treewidth, and we hence can solve the guarding problem for s-guards, k-periscope-guards, sliding cameras, and a special case of L_1-distance-guards, presuming the polygon has bounded treewidth.

One crucial ingredient (similarly used in [3, 4]) is that we can usually reduce the (infinite) set of possible guards to a finite set of "candidate guards", and the (infinite) set of points that need to be guarded to a finite set of "watch points" while maintaining an equivalent problem. This is not trivial (and in fact, false for some guarding-types), and may be of independent interest since it does not require the polygon to have bounded treewidth. We discuss this in Section 2.

To explain the construction for s-guarding, we first solve (in Section 3.1) a subproblem in which an s-guard can only see along a staircase in north-eastern direction. We then combine four of the obtained constructions to solve s-guarding (Section 3.2). In Section 4, we modify the construction to solve several other orthogonal guarding variants. We conclude in Section 5.

2 Preliminaries

Throughout the paper, let P denote an orthogonal polygon (possibly with holes) with n vertices. We already defined α-guards (for $\alpha = r, s$, periscope, etc.). The α-*guarding problem* consists of finding the minimum set of α-guards that can see all points in P. We solve a more general problem that allows to restrict the set of guards and points to be guarded. Thus, the (Γ, X)-α-*guarding problem*, for some (possibly infinite) sets $\Gamma \subseteq P$ and $X \subseteq P$, consists of finding a minimum subset S of Γ such that all points in X are α-guarded by some point in S, or reporting that no such set exists. Note that with this, we can for example restrict guards to be only at polygon-vertices or at the polygon-boundary, if so desired. The standard α-guarding problem is the same as the (P, P)-α-guarding problem.

Recall that the *standard pixelation* of P is obtained by extending a horizontal and a vertical ray inwards from any reflex vertex until they hit the boundary. For the rest of this paper, we refer to the standard pixelation simply as the pixelation of P. The *1-refinement* of the pixelation of P is the result of partitioning every pixel into four equal-sized rectangles. See Figure 1.

The pixelation of P can be seen as planar straight-line graph, with vertices at pixel-corners and edges along pixel-sides. For ease of notation we do not distinguish between the geometric construct (pixel/pixel-corner/pixel-side) and its equivalent in the graph (face/vertex/edge). To solve guarding problems, it usually suffices to study this graph due to the following lemma whose proof is given in the full version of the paper:

Lemma 1 *Let P be a polygon with the pixelation Ψ. Let π be an orthogonal path inside P that connects two vertices g, p of Ψ. Then there exists a path π' from g to p along edges of Ψ that satisfies*

- *π' is monotone if π was,*
- *π' has no more bends than π,*
- *π' is no longer than π.*

2.1 Tree decompositions

A *tree decomposition* of a graph G is a tree I and an assignment $\mathcal{X} : I \to 2^{V(G)}$ of *bags* to the nodes of I

[1] We use "L_1" to emphasize that this path *must* be orthogonal; the concept would make sense for non-orthogonal paths but we do not have any results for them.

such that (i) for any vertex v of G, the bags containing v form a connected subtree of I and (ii) for any edge (v, w) of G, some bag contains both v and w. The width of such a decomposition is $\max_{X \in \mathcal{X}} |X| - 1$, and the *treewidth* $tw(G)$ of G is the minimum width over all tree decompositions of G.

We aim to prove results for polygons where the pixelation has bounded treewidth. Because we sometimes use the 1-refinement of P instead, we need the following observation, which holds since every pixel-corner in a bag can be replaced by the 9 pixel-corners of the 1-refinement that it shares a pixel with.

Observation 2.1 *Let P be a polygon with the pixelation Ψ of treewidth t. Then the 1-refinement of Ψ has treewidth $O(t)$.*

The pixelation of an n-vertex polygon may well have $\Omega(n^2)$ vertices in general, but not for polygons of bounded treewidth. The following lemma is proved in the full version of the paper.

Lemma 2 *Let P be a polygon with n vertices and treewidth t. Then, the pixelation Ψ of P has $O(3^t n)$ vertices.*

The 1-refinement has asymptotically the same number of vertices as the pixelation, hence it also has $O(3^t n)$ vertices.

2.2 Reducing the problem size

In the standard guarding problem, guards can be at an infinite number of points inside P, and we must guard the infinite number of all points inside P. To reduce the guarding problem to a graph problem, we must argue that it suffices to consider a finite set of guards (we call them *candidate guards*) and to check that a finite set of points is guarded (we call them *watch points*). Such reductions are known for r-guarding [4] and sliding cameras [3]. Rather than re-proving it for each guarding type individually, we give here a general condition under which such a reduction is possible.

We need some notation. First, all our guarding models (with the exception of sliding cameras) use *point guards*, i.e., guards are points that belong to P. Also, all guarding models are *symmetric*, i.e., point g guards point p if and only if p guards g. We say that two guarding problems (Γ, X) and (Γ', X') are *equivalent* if given the solution of one of them, we can obtain the solution of the other one in linear time. We prove the following lemma in the full version due to space constraints.

Lemma 3 *Let P be an orthogonal polygon with the pixelation Ψ. Consider a guarding-model α that uses point guards, is symmetric, and satisfies the following:*

(a) For any pixel ψ and any point $g \in P$, if g α-guards one point p in the interior of ψ, then it α-guards all points in ψ.

(b) For any edge e of a pixel and any point $g \in P$, if g α-guards one point p in the interior of e, then it α-guards all points on e.

Then for any (possibly infinite) sets $X, \Gamma \subseteq P$ there exist (finite) sets X', Γ' such that (Γ, X)-α-guarding and (Γ', X')-α-guarding are equivalent. Moreover, X' and Γ' consist of vertices of the 1-refinement of Ψ.

It is easy to see that the conditions of Lemma 3 are satisfied for r-guarding, s-guarding and k-periscope guarding (for any k). We leave the details to the reader.

3 Algorithm for (Γ, X)-s-guarding

In this section, we give a linear-time algorithm for the (Γ, X)-s-guarding problem on any orthogonal polygon P with bounded treewidth. By Lemma 3, we may assume that Γ and X consist of vertices of the 1-refinement of the pixelation. As argued earlier, the 1-refinement also has bounded treewidth. Thus, it suffices to solve the (Γ, X)-s-guarding where Γ and X are vertices of the pixelation Ψ that has bounded treewidth.

3.1 (Γ, X)-NE-Guarding

For ease of explanation, we first solve a special case where guards can look in only two of the four directions and then show how to generalize it to s-guarding. We say that a point g ***NE**-guards* a point p if there exists an orthogonal path π inside P from g to p that goes alternately *north* and *east*; we call π a ***NE**-path*. Define NW-, SE- and SW-guarding analogously.

Note that NE-guarding does not satisfy the conditions of Lemma 3 because it is not symmetric; see e.g. Figure 2, where all crosses are needed to NE-guard all circles. So, we cannot solve the NE-guarding problem in general, but we can solve (X, Γ)-NE-guarding since we already know that X and Γ are vertices of the pixelation.

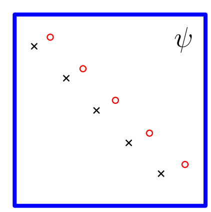

Figure 2: One pixel needs many guards.

Constructing an auxiliary graph H. Define graph H to be the graph of the pixelation of P and direct each edge of H toward north or east; see Figure 3 for an example. By assumption, $X \subseteq V(H)$ and $\Gamma \subseteq V(H)$.

By Lemma 1, there exists an NE-path from guard $g \in \Gamma$ to point $p \in X$ if and only if there exists one along the pixelation-edges. With our choice of edge-directions for H, hence there exists such a NE-path if and only if there exists a directed path from g to p in H.

Thus, (Γ, X)-NE-guarding reduces to the following problem which we call *reachability-cover*: Given a di-

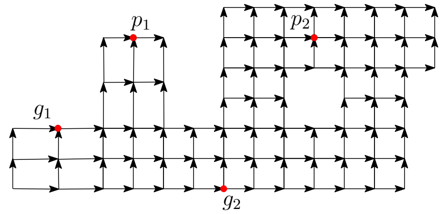

Figure 3: The graph H corresponding to NE-guarding the polygon of Figure 1. Guards and points have been shifted to pixelation-vertices.

rected graph G and vertex sets A and B, find a minimum set $S \subseteq A$ such that for any $t \in B$ there exists an $s \in S$ with a directed path from s to t. (X, Γ)-NE-guarding is equivalent to reachability-cover in graph H using $A := \Gamma$ and $B := X$.

Reachability-cover is NP-hard because set cover can easily be expressed in it. We now argue that reachability-cover can be solved in graphs of bounded treewidth, by appealing to monadic second order logic or MSOL (see [7] for an overview). Briefly, this means that the desired graph property can be expressed as a logical formula that may have quantifications, but only on variables and sets. Courcelle's theorem states that any problem expressible in MSOL can be solved in linear time on graphs of bounded treewidth [6]. (Courcelle's original result was only for decision problems, but it can easily be generalized to minimization problems.) Define $\texttt{Reachability}(u, v, G)$ to be the property that there exists a directed path from u to v in a directed graph G. This can be expressed in MSOL [7]. Consequently, the (Γ, X)-NE-guarding problem can be expressed in MSOL as follows:

$$\exists S \subseteq \Gamma : \forall p \in X : \exists g \in S : \texttt{Reachability}(g, p, H).$$

So we can solve the (Γ, X)-NE-guarding problem if Γ and X are vertices of a given pixelation that has bounded treewidth.

3.2 (Γ, X)-s-guarding

Solving the (Γ, X)-s-guarding problem now becomes very simple, by exploiting that a guard g s-guards a point p if only if g β-guards p for some $\beta \in$\{NE, NW, SE, SW\}. We can solve the (Γ, X)-β-guarding problem for $\beta \neq$ NE similarly as in the previous section, by directing the auxiliary graph H according to the directions we wish to take. Let $H_{\texttt{NE}}, H_{\texttt{NW}}, H_{\texttt{SE}}, H_{\texttt{SW}}$ be the four copies of graph H (directed in four different ways) that we get. Define a new auxiliary graph H^* as follows (see also Figure 4): Initially, let $H^* := H_{\texttt{NE}} \cup H_{\texttt{NW}} \cup H_{\texttt{SE}} \cup H_{\texttt{SW}}$. For each $g \in \Gamma$, add to H^* a new vertex $v^\Gamma(g)$ and the directed edges $(v^\Gamma(g), v_\beta(g))$ where $v_\beta(g)$ (for $\beta \in$\{NE, NW, SE, SW\}) is the vertex in H_β) corresponding to g. Similarly, for each $p \in X$, add to H^* a new vertex $v^X(p)$ and the directed edges $(v_\beta(p), v^X(p))$ for $\beta \in$\{NE, NW, SE, SW\}).

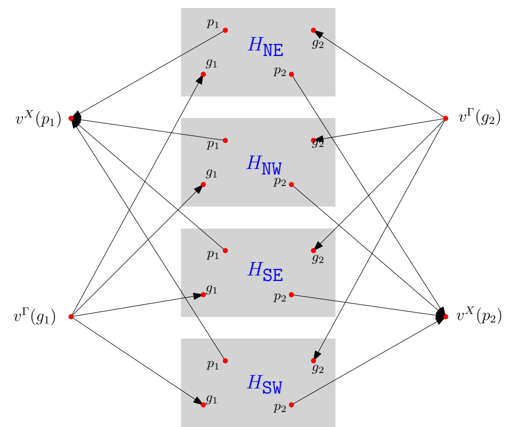

Figure 4: The construction of graph H^*.

If some guard g s-guards a point p, then there exists a β-path from g to p inside P for some $\beta \in$\{NE, NW, SE, SW\}. We can turn this path into a β-path along pixelation-edges by Lemma 1, and therefore find a path from $v^\Gamma(g)$ to $v^X(p)$ by going to H_β and following the path within it. Vice versa, any directed path from $v^\Gamma(g)$ to $v^X(p)$ must stay inside H_β for some $\beta \in$\{NE, NW, SE, SW\} since $v^\Gamma(g)$ is a source and $v^X(p)$ is a sink. Therefore (Γ, X)-s-guarding is the same as reachability-cover in H^* with respect to the sets $V(\Gamma) := \{v^\Gamma(g) : g \in \Gamma\}$ and $V(X) := \{v^X(p) : p \in X\}$.

It is easy to see that H^* has bounded treewidth if Ψ does, by replacing each vertex p of Ψ in a bag by its up to six copies $v_\beta(p)$, $v^X(p)$, $v^\Gamma(p)$. Now we put it all together. Assume P has bounded treewidth, hence its (standard) pixelation has bounded treewidth and $O(n)$ edges, and so does its 1-refinement. This is in fact the partition of P that we use to obtain H^*, therefore H^* also has bounded treewidth and $O(n)$ edges. We can apply Courcelle's theorem to solve reachability-cover in H^* and obtain:

Theorem 4 *Let P be an orthogonal polygon with bounded treewidth. Then, there exists a linear-time algorithm for the (Γ, X)-s-guarding problem on P.*

4 Other Guarding Types

In this section, we show how similar methods apply to other types of orthogonal guarding. The main difference is that we need edge-weights on the auxiliary graph. To solve the guarding problem, we hence use a version of reachability-cover defined as follows. The

(G, A, B, W)-*bounded-reachability-cover problem* has as input an edge-weighted directed graph G, two vertex sets A and B, and a length-bound W. The objective is to find a minimum-cardinality set $S \subseteq A$ such that for any $t \in B$ there exists an $s \in S$ with a directed path from s to t that has length at most W. We need to argue that this problem is solvable if G has bounded treewidth, at least if W is sufficiently small. Recall that reachability-cover can be expressed in monadic second-order logic. Arnborg et al. [1] introduced the class of *extended* monadic second-order problems which allow integer weights on the input. They showed that problems expressible in extended monadic second-order logic can be solved on graphs of bounded treewidth, with a run-time that is polynomial in the graph-size and the maximum weight.

4.1 L_1-distance guarding

We first study the L_1-distance guarding problem. We have not been able to solve the L_1-distance guarding problem for all polygons of bounded treewidth. The main problem is that the bounded-reachability-cover problem is solved in run-time that depends on the maximum weight. For this to be polynomial, we must assume that all edges of the input-polygon have integer length that is polynomial in n.

Let Γ and X be subsets of the vertices of the pixelation Ψ of P. Let H_{dist} be the auxiliary graph obtained from the pixelation graph by making all edges bi-directional. Set the weight of each edge to be its length. If a guard $g \in \Gamma$ sees a point $p \in X$ in the L_1-distance guarding model (with distance-bound D), then there exists a path π from g to p that has length at most D. By Lemma 1, we may assume that π runs along pixel-edges. Hence π gives rise to a directed path in H_{dist} of length less than D. Vice versa, any such path in H_{dist} means that g can L_1-distance-guard p. In consequence, the (Γ, X)-L_1-distance guarding problem is the same as the $(\Gamma, X, H_{\text{dist}}, D)$-bounded-reachability-cover problem. This can be solved in polynomial time, presuming the pixelation has bounded treewidth and $O(n)$ edges and the lengths of all edges of P are integers that are polynomial in n.

4.2 k-periscope guarding

For k-periscope guarding, we define an auxiliary graph H_{peri} based on the graph of the pixelation, but modify it near each vertex and add weights to encode the number of bends, rather than the length, of a path. If u is a vertex of the pixelation, then replace it with a K_4 as shown in Figure 5. We denote this copy of K_4 by K_4^u, and let its four vertices be u_N, u_S, u_W and u_E according to compass directions. For a vertex u on the boundary of P we omit those vertices in K_4^u that would fall outside P. We connect copies K_4^u and K_4^v of a pixel-edge (u, v) in the natural way, e.g. if (u, v) was vertical with u below v, then we connect u_N to v_S. All edges are bidirectional.

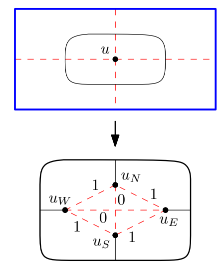

Figure 5: Adding K_4.

For any $g \in \Gamma$, define a new vertex $v^\Gamma(g)$ and add edges from it to all of g_N, g_S, g_E, g_W that exist in the graph. For any $p \in X$, define a new vertex $v^X(p)$ and add edges from all of p_N, p_S, p_E, p_W to $v^X(p)$. Set all edge weights to 0, except for the "diagonal" edges between consecutive vertices of a K_4, which have weight 1 as shown in Figure 5.

Clearly, $g \in \Gamma$ can see $p \in X$ (in the k-periscope guarding model) if and only if there is a directed path from $v^\Gamma(g)$ to $v^X(p)$ in the constructed graph that uses at most k diagonal edges, i.e., that has length at most k. Thus the k-periscope guarding model reduces to bounded-reachability-cover. Since k-periscope-guarding satisfies the conditions for Lemma 3, we can hence solve the k-periscope guarding problem in polynomial time in any polygon of bounded treewidth. Note that the run-time depends polynomially on k, so k need not be a constant.

4.3 Sliding cameras

It was already known that the sliding camera problem is polynomial in polygons of bounded treewidth [3]. However, using much the same auxiliary graph as in the previous subsection we can get a second (and in our opinion, easier) method of obtaining this result.

We solve the (Γ, X)-sliding camera guarding problem, for some set of sliding cameras Γ (which are segments inside P) and watch points X. It was argued in [3] that we may assume Γ to be a finite set of maximal segments that lie along the pixelation; in particular the endpoints of candidate guards are pixel-vertices. As for X, we cannot apply Lemma 3 directly, since sliding cameras are not point guards and hence not symmetric. But sliding cameras do satisfy conditions (a) and (b) of Lemma 3. As one can easily verify by following the proof, we may therefore assume X to consist of pixel-vertices of the 1-refinement. (A similar result was also argued in [3].)

We build an auxiliary graph H_{slide} almost exactly as in the previous subsection. Thus, start with the graph of the 1-refinement of the pixelation. Replace every vertex by a K_4, weighted as before. (All other edges receive weight 0.) For each $p \in X$, define a new vertex $v^X(p)$ and connect it as in the previous subsection, i.e., add edges from p_N, p_E, p_W, p_S to $v^X(p)$. For any sliding camera $\gamma \in \Gamma$, add a new vertex $v^\Gamma(\gamma)$. The only new thing is how these vertices get connected. If γ is horizontal, then add an edge from $v^\Gamma(\gamma)$ to g_W, where g_W is the left endpoint of γ. If γ is vertical, then add an edge from $v^\Gamma(\gamma)$ to g_N, where g_N is the top endpoint of γ.

It is not hard to verify that a sliding camera γ can see a point p if and only if there exists a directed path from $v^\Gamma(\gamma)$ to $v^X(p)$ in H_{slide} that has length at most 1. Due to space constraints, we prove this formally in the full version of the paper. Therefore the sliding camera problem reduces to a bounded-reachability-cover problem where all weights are at most 1; this can be solved in polynomial (in fact, linear) time if the polygon has bounded treewidth.

5 Conclusion

In this paper, we gave algorithms for guarding orthogonal polygons of bounded treewidth. We considered various models of orthogonal guarding, and solved the guarding problem on such polygons for s-guards, k-periscope guards, and sliding cameras, and some other related guarding types.

As for open problems, the main question is whether these results could be used to obtain better approximation algorithms. Baker's method [2] yields a PTAS for many problems in planar graphs by splitting the graph into graphs of bounded treewidth and combining solutions suitably. However, this requires the problems to be "local" in some sense, and the guarding problems considered here are not local in that a guard may see a point whose distance in the graph of the pixelation is very far, which seems to make Baker's approach infeasible. Are these guarding problems APX-hard in polygons with holes?

References

[1] S. Arnborg, J. Lagergren, and D. Seese. Easy problems for tree-decomposable graphs. *J. Algorithms*, 12(2):308–340, 1991. 5

[2] B. Baker. Approximation algorithms for NP-complete problems on planar graphs. *J. ACM*, 41(1):153–180, 1994. 6

[3] T. Biedl, T. M. Chan, S. Lee, S. Mehrabi, F. Montecchiani, and H. Vosoughpour. On guarding orthogonal polygons with sliding cameras. In *WALCOM 2017*, volume 10167 of *LNCS*, pages 54–65. Springer, 2017. 2, 3, 5

[4] T. Biedl and S. Mehrabi. On r-guarding thin orthogonal polygons. In *ISAAC 2016*, volume 64 of *LIPIcs*, pages 17:1–17:13, 2016. 2, 3

[5] V. Chvátal. A combinatorial theorem in plane geometry. *Journal of Combinatorial Theory, Series B*, 18:39–41, 1975. 1

[6] B. Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Inf. Comput.*, 85(1):12–75, 1990. 4

[7] B. Courcelle. On the expression of graph properties in some fragments of monadic second-order logic. In *DIAMCS Workshop on Descriptive Complexity and Finite Models*, pages 33–57. American Mathematical Society, 1997. 4

[8] S. Durocher and S. Mehrabi. Guarding orthogonal art galleries using sliding cameras: algorithmic and hardness results. In *MFCS 2013*, volume 8087 of *LNCS*, pages 314–324, 2013. 2

[9] S. Eidenbenz, C. Stamm, and P. Widmayer. Inapproximability results for guarding polygons and terrains. *Algorithmica*, 31(1):79–113, 2001. 1

[10] L. Gewali and S. C. Ntafos. Covering grids and orthogonal polygons with periscope guards. *Comput. Geom.*, 2:309–334, 1992. 2

[11] S. K. Ghosh. Approximation algorithms for art gallery problems in polygons. *Disc. App. Math.*, 158(6):718–722, 2010. 1

[12] E. Krohn and B. J. Nilsson. Approximate guarding of monotone and rectilinear polygons. *Algorithmica*, 66(3):564–594, 2013. 1

[13] D. T. Lee and A. K. Lin. Computational complexity of art gallery problems. *IEEE Transactions on Information Theory*, 32(2):276–282, 1986. 1

[14] R. Motwani, A. Raghunathan, and H. Saran. Covering orthogonal polygons with star polygons: The perfect graph approach. In *SoCG 1988*, pages 211–223, 1988. 2

[15] J. O'Rourke. *Art Gallery Theorems and Algorithms*. Oxford University Press, New York, NY, 1987. 1

[16] D. Schuchardt and H.-D. Hecker. Two NP-hard art-gallery problems for ortho-polygons. *Mathematical Logic Quarterly*, 41(2):261–267, 1995. 1

[17] A. P. Tomás. Guarding thin orthogonal polygons is hard. In *Fundamentals of Computation Theory (FCT 2013)*, volume 8070 of *LNCS*, pages 305–316, 2013. 1

[18] C. Worman and J. M. Keil. Polygon decomposition and the orthogonal art gallery problem. *Int. J. Comput. Geometry Appl.*, 17(2):105–138, 2007. 1, 2