

Parameterized Hardness of Art Gallery Problems*

Édouard Bonnet¹ and Tillmann Miltzow²

- 1 Institute for Computer Science and Control, Hungarian Academy of Sciences (MTA SZTAKI), Budapest, Hungary
edouard.bonnet@lamsade.dauphine.fr
- 2 Institute for Computer Science and Control, Hungarian Academy of Sciences (MTA SZTAKI), Budapest, Hungary
t.miltzow@gmail.com

Abstract

Given a simple polygon \mathcal{P} on n vertices, two points x, y in \mathcal{P} are said to be visible to each other if the line segment between x and y is contained in \mathcal{P} . The POINT GUARD ART GALLERY problem asks for a minimum set S such that every point in \mathcal{P} is visible from a point in S . The VERTEX GUARD ART GALLERY problem asks for such a set S subset of the vertices of \mathcal{P} . A point in the set S is referred to as a guard. For both variants, we rule out a $f(k)n^{o(k/\log k)}$ algorithm, for any computable function f , where $k := |S|$ is the number of guards, unless the Exponential Time Hypothesis fails. These lower bounds almost match the $n^{O(k)}$ algorithms that exist for both problems.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems

Keywords and phrases art gallery problem, computational geometry, parameterized complexity, ETH-based lower bound, geometric set cover/hitting set

Digital Object Identifier 10.4230/LIPIcs.ESA.2016.19

1 Introduction

Given a simple polygon \mathcal{P} on n vertices, two points x, y in \mathcal{P} are said to be visible to each other if the line segment between x and y is contained in \mathcal{P} . The POINT GUARD ART GALLERY problem asks for a minimum set S such that every point in \mathcal{P} is visible from a point in S . The VERTEX GUARD ART GALLERY problem asks for such a set S subset of the vertices of \mathcal{P} . The set S is referred to as guards. In what follows, n refers to the number of vertices of \mathcal{P} and k to the size of an optimal set of guards.

The art gallery problem is arguably one of the most well-known problems in discrete and computational geometry. Since its introduction by Viktor Klee in 1976, three books [32, 34, 14] and two extensive surveys appeared [33, 8]. O’Rourke’s book from 1987 has over a thousand citations, and each year, top conferences publish new results on the topic. Many variants of the art gallery problem, based on different definitions of visibility, restricted classes of polygons, different shapes of guards, have been defined and analyzed. One of the first results is the elegant proof of Fisk that $\lfloor n/3 \rfloor$ guards are always sufficient and sometimes necessary for a polygon with n vertices [12].

NP-hardness and APX-hardness have been shown for many variants of the art gallery problem and other related problems [11, 23, 4, 26]. Due to those negative results, most

* Supported by the ERC grant PARAMTIGHT: “Parameterized complexity and the search for tight complexity results”, no. 280152.



papers concentrate on finding approximation algorithms and restrictions that are polynomially tractable [15, 25, 24, 30, 26]. However, considering the recent lack of progress in this direction, the study of other approaches becomes interesting. One such approach is finding heuristics to solve large instances of the art gallery problem [8]. The fundamental drawback of this approach is the lack of performance guarantee.

In the last twenty-five years, a fruitful paradigm, parameterized complexity, has been gaining some popularity. The underlying idea is to study algorithmic problems with dependence on a natural parameter. If the dependence on the parameter is practical and the parameter is small for real-life instances, one gets algorithms that give optimal solutions with reasonable running times. For a gentle introduction to parameterized complexity, we recommend Niedermeier's book [31]. For a thorough reading highlighting complexity classes, we suggest the book by Downey and Fellows [9]. For a recent book on the topic with an emphasis on algorithms, we advise to read the book by Cygan et al. [6]. An approach based on logic is given by Flum and Grohe [13]. Despite the recent successes of parameterized complexity, only very few results on the art gallery problem are known.

The first such result is the trivial algorithm for the vertex guard variant to check if a solution of size k exists in a polygon with n vertices. The algorithm runs in $O(n^{k+2})$ time, by checking all possible subsets of size k of the vertices. The second *not so well-known* result is the fact that one can find in time $n^{O(k)}$ a set of k guards for the point guard variant, if it exists [10], using tools from real algebraic geometry [2]. This was first observed by Sharir [10, Acknowledgment]. Despite the fact that the first algorithm is extremely basic and the second algorithm, even with remarkably sophisticated tools, uses almost no problem-specific insights, no better exact parameterized algorithms are known.

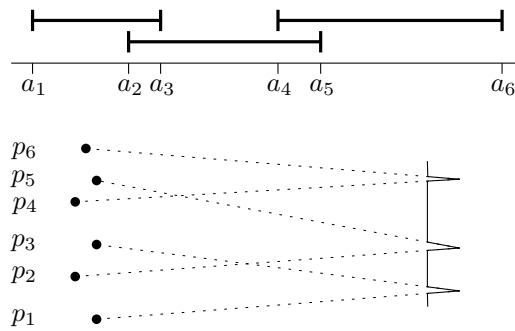
The Exponential Time Hypothesis (ETH) asserts that there is no $2^{o(N)}$ time algorithm for SAT on N variables. The ETH is used to attain more precise conditional lower bounds than the mere NP-hardness. A simple reduction from SET COVER by Eidenbenz et al. shows that there is no $n^{o(k)}$ algorithm for these problems, when we consider polygons with holes [11, Sec.4], unless the ETH fails. However, polygons with holes are very different from simple polygons. For instance, they have unbounded VC-dimension while simple polygons have bounded VC-dimension [35, 20, 22, 19]. Our contribution is to show that, even on simple polygons, one cannot expect a large improvement over the $n^{O(k)}$ algorithms. More precisely, we prove:

► **Theorem 1** (Parameterized hardness point guard). *Assuming the ETH, POINT GUARD ART GALLERY is not solvable in time $f(k)n^{o(k/\log k)}$, for any computable function f , even on simple polygons, where n is the number of vertices of the polygon and k is the number of guards allowed.*

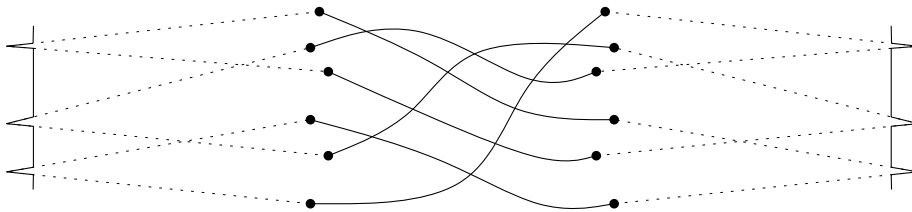
► **Theorem 2** (Parameterized hardness vertex guard). *Assuming the ETH, VERTEX GUARD ART GALLERY is not solvable in time $f(k)n^{o(k/\log k)}$, for any computable function f , even on simple polygons, where n is the number of vertices of the polygon and k is the number of guards allowed.*

Our reductions are from SUBGRAPH ISOMORPHISM. Therefore an algorithm solving the art gallery problem in time $f(k)n^{o(k/\log k)}$ would also improve current running times for SUBGRAPH ISOMORPHISM and for solving CSPs parameterized by treewidth, which are major open questions [28]. Our results imply, in particular, that both variants are $W[1]$ -hard parameterized by the number of guards.

Finally, let us mention a sample of works on the parameterized complexity (with an emphasis on hardness) of other geometric problems. The complexity of some fundamental



■ **Figure 1** Reduction from HITTING SET on interval graphs to a restricted version of the art gallery problem.



■ **Figure 2** Two instances of Hitting Set “magically” linked.

problems parameterized by the dimension d has been addressed [17]; it was shown that, assuming the ETH, algorithms running in time $n^{O(d)}$ are essentially optimal (with n being the size of the instance). Extracting from a finite set of points of \mathbb{R}^3 the largest subset in convex position and whose convex-hull interior is empty is W[1]-hard [16]. More results on geometric covering or packing problems include the following papers [5, 27, 1, 29, 7]. We refer the interested reader to the extensive survey of Giannopoulos et al. [18].

Proof ideas. In order to achieve these results, we slightly extend some known hardness results of geometric set cover/hitting set problems and combine them with problem-specific insights of the art gallery problem. One of the first problem-specific insights is the ability to encode HITTING SET on interval graphs. The reader can refer to Figure 1 for the following description. Assume that we have some fixed points p_1, \dots, p_n with increasing y -coordinates in the plane. We can build a pocket “far enough to the right” that can be seen only from $\{p_i, \dots, p_j\}$ for any $1 \leq i < j \leq n$.

Let I_1, \dots, I_n be n intervals with endpoints a_1, \dots, a_{2n} . Then, we construct $2n$ points p_1, \dots, p_{2n} representing a_1, \dots, a_{2n} . Further, we construct one pocket “far enough to the right” for each interval as described above. This way, we reduce HITTING SET on interval graphs to a restricted version of the art gallery problem. This observation is *not* so useful in itself since hitting set on interval graphs can be solved in polynomial time.

The situation changes rapidly if we consider HITTING SET on 2-track interval graphs, as described in Section 2. Unfortunately, we are not able to just “magically” link some specific pairs of points in the polygon of the art gallery instance. Therefore, we construct linker gadgets, which basically work as follows. We are given two set of points P and Q and a bijection σ between P and Q . The linker gadget is built in a way that it can be covered by two points (p, q) of $P \times Q$, if and only if $q = \sigma(p)$. The STRUCTURED 2-TRACK HITTING SET problem will be specifically designed so that the linker gadget is the main remaining ingredient to show hardness.

Organization of the paper. In Section 2, we introduce some notations, discuss the encoding of the polygon, and give some useful ETH-based lower bounds. We show a lower bound for STRUCTURED 2-TRACK HITTING SET based on the lower bound known for MULTICOLORED SUBGRAPH ISOMORPHISM. Due to space limitation, this proof is only included in the arxiv version of the paper [3]. Then, we reduce from the particularly convenient STRUCTURED 2-TRACK HITTING SET. In Section 3, we show the lower bound for the POINT GUARD ART GALLERY problem (Theorem 1). We design a linker gadget, show its correctness, and show how several linker gadgets can be combined consistently. In Section 4, we tackle the VERTEX GUARD ART GALLERY problem (Theorem 2). We have to design a very different linker gadget, that has to be combined with other gadgets and ideas.

2 Preliminaries

For any two integers $x \leq y$, we set $[x, y] := \{x, x + 1, \dots, y - 1, y\}$, and for any positive integer x , $[x] := [1, x]$. Given two points a, b in the plane, we define $\text{seg}(a, b)$ as the line segment with endpoints a, b . Given n points $v_1, \dots, v_n \in \mathbb{R}^2$, we define a polygonal closed curve c by $\text{seg}(v_1, v_2), \dots, \text{seg}(v_{n-1}, v_n), \text{seg}(v_n, v_1)$. If c is not self intersecting, it partitions the plane into a closed bounded area and an unbounded area. The closed bounded area is a *simple polygon* on the vertices v_1, \dots, v_n . Note that we do not consider the boundary as the polygon but rather all the points bounded by the curve c as described above. Given two points a, b in a simple polygon \mathcal{P} , we say that a *sees* b or a is *visible* from b if $\text{seg}(a, b)$ is contained in \mathcal{P} . By this definition, it is possible to “see through” vertices of the polygon. We say that S is a set of *point guards* of \mathcal{P} , if every point $p \in \mathcal{P}$ is visible from a point of S . We say that S is a set of *vertex guards* of \mathcal{P} , if additionally S is a subset of the vertices of \mathcal{P} . The POINT GUARD ART GALLERY problem and the VERTEX GUARD ART GALLERY problem are formally defined as follows.

POINT GUARD ART GALLERY

Input: The vertices of a simple polygon \mathcal{P} in the plane and a natural number k .

Question: Does there exist a set of k point guards for \mathcal{P} ?

VERTEX GUARD ART GALLERY

Input: A simple polygon \mathcal{P} on n vertices in the plane and a natural number k .

Question: Does there exist a set of k vertex guards for \mathcal{P} ?

For any two distinct points v and w in the plane we denote by $\text{ray}(v, w)$ the ray starting at v and passing through w , and by $\ell(v, w)$ the supporting line passing through v and w . For any point x in a polygon \mathcal{P} , $V_{\mathcal{P}}(x)$, or simply $V(x)$, denotes the *visibility region* of x within \mathcal{P} , that is the set of all the points $y \in \mathcal{P}$ seen by x . We say that two vertices v and w of a polygon \mathcal{P} are *neighbors* or *consecutive* if vw is an edge of \mathcal{P} . A *sub-polygon* \mathcal{P}' of a simple polygon \mathcal{P} is defined by any l distinct consecutive vertices v_1, v_2, \dots, v_l of \mathcal{P} (that is, for every $i \in [l - 1]$, v_i and v_{i+1} are neighbors in \mathcal{P}) such that v_1v_l does not cross any edge of \mathcal{P} . In particular, \mathcal{P}' is a simple polygon.

We assume that the vertices of the polygon are either given by integers or by rational numbers. We also assume that the output is given either by integers or by rational numbers. The instances we generate as a result of Theorem 1 and Theorem 2 have rational coordinates. We can represent them by specifying the nominator and denominator. The number of bits is bounded by $O(\log n)$ in both cases. We can transform the coordinates to integers by multiplying every coordinate with the least common multiple of all denominators. However, this leads to integers using $O(n \log n)$ bits.

ETH-based lower bounds. The *Exponential Time Hypothesis* (ETH) is a conjecture by Impagliazzo et al. [21] asserting that there is no $2^{o(n)}$ -time algorithm for 3-SAT on instances with n variables.

The MULTICOLORED SUBGRAPH ISOMORPHISM problem can be defined in the following equivalent way. One is given a graph with n vertices partitioned into l color classes V_1, \dots, V_l such that only k of the $\binom{l}{2}$ sets $E_{ij} = E(V_i, V_j)$ are non empty. The goal is to pick one vertex in each color class so that the selected vertices induce k edges. Observe that l corresponds to the number of vertices of the pattern graph. The technique of color coding and a result of Marx imply that:

► **Theorem 3** ([28]). *Unless the ETH fails, MULTICOLORED SUBGRAPH ISOMORPHISM cannot be solved in time $f(k) n^{o(k/\log k)}$ where k is the number of edges of the solution and f any computable function.*

Naturally, this result still holds when restricted to connected input graphs. In that case, $k \geq l - 1$.

In the 2-TRACK HITTING SET problem, the input consists of an integer k , two totally ordered ground sets A and B of the same cardinality, and two sets \mathcal{S}_A of A -intervals, and \mathcal{S}_B of B -intervals. In addition, the elements of A and B are in one-to-one correspondence $\phi : A \rightarrow B$ and each pair $(a, \phi(a))$ is called a *2-element*. The goal is to find, if possible, a set S of k 2-elements such that the first projection of S is a hitting set of \mathcal{S}_A , and the second projection of S is a hitting set of \mathcal{S}_B .

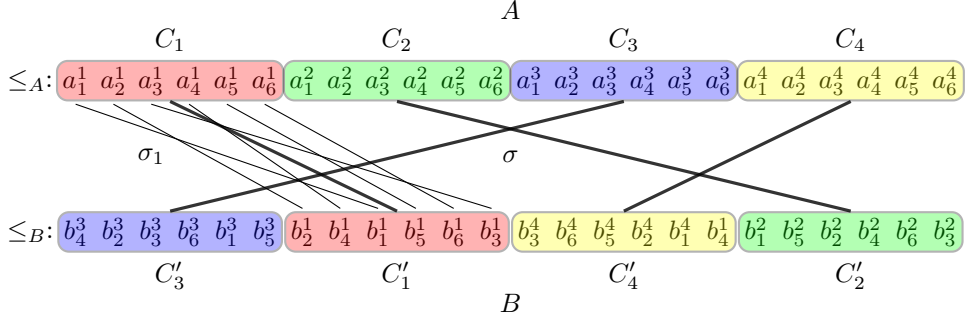
STRUCTURED 2-TRACK HITTING SET is the same problem with color classes over the 2-elements, and a restriction on the one-to-one mapping ϕ . Given two integers k and t , A is partitioned into (C_1, C_2, \dots, C_k) where $C_j = \{a_1^j, a_2^j, \dots, a_t^j\}$ for each $j \in [k]$. A is ordered: $a_1^1, a_2^1, \dots, a_1^t, a_2^t, \dots, a_t^t, \dots, a_1^k, a_2^k, \dots, a_t^k$. We define $C'_j := \phi(C_j)$ and $b_i^j := \phi(a_i^j)$ for all $i \in [t]$ and $j \in [k]$. We now impose that ϕ is such that, for each $j \in [k]$, the set C'_j is a B -interval. That is, B is ordered: $C'_{\sigma(1)}, C'_{\sigma(2)}, \dots, C'_{\sigma(k)}$ for some permutation on $[k]$, $\sigma \in \mathfrak{S}_k$. For each $j \in [k]$, the order of the elements within C'_j can be described by a permutation $\sigma_j \in \mathfrak{S}_t$ such that the ordering of C'_j is: $b_{\sigma_j(1)}^j, b_{\sigma_j(2)}^j, \dots, b_{\sigma_j(t)}^j$. In what follows, it will be convenient to see an instance of STRUCTURED 2-TRACK HITTING SET as a tuple $\mathcal{I} = (k \in \mathbb{N}, t \in \mathbb{N}, \sigma \in \mathfrak{S}_k, \sigma_1 \in \mathfrak{S}_t, \dots, \sigma_k \in \mathfrak{S}_t, \mathcal{S}_A, \mathcal{S}_B)$, where we recall that \mathcal{S}_A is a set of A -intervals and \mathcal{S}_B is a set of B -intervals. We denote by $[a_i^j, a_{i'}^{j'}]$ (resp. $[b_i^j, b_{i'}^{j'}]$) all the elements $a \in A$ (resp. $b \in B$) such that $a_i^j \leq_A a \leq_A a_{i'}^{j'}$ (resp. $b_i^j \leq_B b \leq_B b_{i'}^{j'}$).

Taking inspiration from previous results, we show hardness of STRUCTURED 2-TRACK HITTING SET by a reduction from MULTICOLORED SUBGRAPH ISOMORPHISM. Due to lack of space, we do not include the proof of the following theorem. The interested reader can find this proof in the arxiv version of the paper [3].

► **Theorem 4.** *STRUCTURED 2-TRACK HITTING SET is $W[1]$ -hard, and not solvable in time $f(k) |\mathcal{I}|^{o(k/\log k)}$ for any computable function f , unless the ETH fails.*

3 Parameterized hardness of the point guard variant

As exposed in the introduction, we give a reduction from the STRUCTURED 2-TRACK HITTING SET problem. The main challenge is to design a *linker* gadget that groups together specific pairs of points in the polygon. The following introductory lemma inspires the *linker* gadgets for both POINT GUARD ART GALLERY and VERTEX GUARD ART GALLERY.



■ **Figure 3** An illustration of the $k + 1$ permutations $\sigma \in \mathfrak{S}_k, \sigma_1 \in \mathfrak{S}_t, \dots, \sigma_k \in \mathfrak{S}_t$ of an instance of STRUCTURED 2-TRACK HITTING SET, with $k = 4$ and $t = 6$.

► **Lemma 5.** *The only minimum hitting sets of the set-system $\mathcal{S} = \{S_i = \{1, 2, \dots, i, \bar{i} + 1, \bar{i} + 2, \dots, \bar{n}\} \mid i \in [n]\} \cup \{\bar{S}_i = \{\bar{1}, \bar{2}, \dots, \bar{i}, i + 1, i + 2, \dots, n\} \mid i \in [n]\}$ are $\{i, \bar{i}\}$, for each $i \in [n]$.*

Proof. First, for each $i \in [n]$, one may easily observe that $\{i, \bar{i}\}$ is a hitting set of \mathcal{S} . Now, because of the sets S_n and \bar{S}_n one should pick one element i and one element \bar{j} for some $i, j \in [n]$. If $i < j$, then set \bar{S}_i is not hit, and if $i > j$, then S_j is not hit. Therefore, i should be equal to j . ◀

► **Theorem 1** (Parameterized hardness point guard). *Assuming the ETH, POINT GUARD ART GALLERY is not solvable in time $f(k) n^{o(k/\log k)}$, for any computable function f , even on simple polygons, where n is the number of vertices of the polygon and k is the number of guards allowed.*

Proof. Given an instance $\mathcal{I} = (k \in \mathbb{N}, t \in \mathbb{N}, \sigma \in \mathfrak{S}_k, \sigma_1 \in \mathfrak{S}_t, \dots, \sigma_k \in \mathfrak{S}_t, \mathcal{S}_A, \mathcal{S}_B)$ of STRUCTURED 2-TRACK HITTING SET, we build a simple polygon \mathcal{P} with $O(kt + |\mathcal{S}_A| + |\mathcal{S}_B|)$ vertices, such that \mathcal{I} is a YES-instance iff \mathcal{P} can be guarded by $3k$ points.

Outline. We recall that A 's order is: $a_1^1, \dots, a_t^1, \dots, a_1^k, \dots, a_t^k$ and B 's order is determined by σ and the σ_j 's (see Figure 3). Let us focus on one color class $j \in [k]$ together with a permutation $\sigma_j : A \rightarrow B$. The global strategy of the reduction is to *allocate*, $2t$ special points for this polygon. The points a_1^j, \dots, a_t^j on track A are represented by $\alpha_1^j, \dots, \alpha_t^j$ points in \mathcal{P} , and the points $\sigma_j(a_1^j), \dots, \sigma_j(a_t^j)$ on track B are represented by $\beta_1^j, \dots, \beta_t^j$ in the polygon. Placing a guard in α_i^j and β_i^j shall correspond to picking the 2-element $(a_i^j, \sigma_j(b_i^j))$. The points α_i^j 's and β_i^j 's ordered by increasing y -coordinates will match the order of the a_i^j 's along the order \leq_A and then of the b_i^j 's along \leq_B . Then, far in the horizontal direction, we will place pockets to encode each A -interval of \mathcal{S}_A , and each B -interval of \mathcal{S}_B .

The first critical issue will be to *link* point α_i^j to point β_i^j . Indeed, in the STRUCTURED 2-TRACK HITTING SET problem, one selects 2-elements (one per color class), so we should prevent one from placing two guards in α_i^j and $\beta_{i'}^j$ with $i \neq i'$. The so-called *point linker* gadget will realize the intervals as described in Lemma 5.

The second critical issue is to enforce these positions. For this purpose, we will need to introduce a *copy* $\bar{\alpha}_i^j$ of each α_i^j . In each part of the gallery encoding a color class $j \in [k]$, the only way of guarding all the pockets with only three guards will be to place them in $\alpha_i^j, \bar{\alpha}_i^j$, and β_i^j for some $i \in [t]$ (see Figure 5). Hence, $3k$ guards will be necessary and sufficient to

guard the whole \mathcal{P} iff there is a solution to the instance of STRUCTURED 2-TRACK HITTING SET.

We now get into the details of the reduction. We will introduce several characteristic lengths and compare them; when $l_1 \ll l_2$ means that l_1 should be thought as really small compared to l_2 , and $l_1 \approx l_2$ means that l_1 and l_2 are roughly of the same order. The motivation is to guide the intuition of the reader without bothering her/him too much about the details. At the end of the construction, we will specify more concretely how those lengths are chosen.

Construction. We start with an explicit specification of the coordinates. The description will be dependent on some parameters x, y, L, D, F that we will specify later. The value x represents the offset between elements with respect to the x -coordinate and likewise the value y represents the offset between elements with respect to the y -coordinate. D represents the vertical distance between different color classes and L represents the horizontal distance between all the α 's and the β 's, see also Figure 6. The value F will become relevant later and describes the distance of the points to the pockets to the far right. The crucial point of the construction is that the order of the α 's corresponds exactly to the order of the a 's along track A and the same relation holds between the β 's and b 's.

We recall that we want the points α_i^j 's and β_i^j 's ordered by increasing y -coordinates, to match the order of the a_i^j 's and b_i^j 's along \leq_A and \leq_B , with first all the elements of A and then all the elements of B . Starting from some y -coordinate y_1 (which is the one given to point α_1^1), the y -coordinates of the α_i^j 's are regularly spaced out by an offset y ; that is, the y -coordinate of α_i^j is $y_1 + (i + (j - 1)t)y$. Between the y -coordinate of the last element in A (i.e., α_t^k whose y -coordinate is $y_1 + (kt - 1)y$) and the first element in B , there is a large offset L , such that the y -coordinate of β_i^j is $y_1 + (kt - 1)y + L + (\text{ord}(b_i^j) - 1)y$ (for any $j \in [k]$ and $i \in [t]$) where $\text{ord}(b_i^j)$ is the rank of b_i^j along the order \leq_B .

For each color class $j \in [k]$, let $x_j := x_1 + (j - 1)D$ for some x -coordinate x_1 and value D , and $y_j := y_1 + (j - 1)ty$. The allocated points $\alpha_1^j, \alpha_2^j, \alpha_3^j, \dots, \alpha_t^j$ are on a line at coordinates: $(x_j, y_j), (x_j + x, y_j + y), (x_j + 2x, y_j + 2y), \dots, (x_j + (t - 1)x, y_j + (t - 1)y)$, for some value x . We place, to the left of those points, a rectangular pocket $\mathcal{P}_{j,r}$ of width, say, y and length, say¹, tx such that the uppermost longer side of the rectangular pocket lies on the line $\ell(\alpha_1^j, \alpha_t^j)$ (see Figure 4). The y -coordinates of $\beta_1^j, \beta_2^j, \beta_3^j, \dots, \beta_t^j$ have already been defined. We set, for each $i \in [t]$, the x -coordinate of β_i^j to $x_j + (i - 1)x$, so that β_i^j and α_i^j share the same x -coordinate. One can check that it is consistent with the previous paragraph. We also observe that, by the choice of the y -coordinate for the β_i^j 's, we have both encoded the permutations σ_j 's and permutation σ (see Figure 6 or Figure 4). This finishes the description of the coordinates.

Now, we will give a description how, we can encode intervals by on track A and B by small pockets and, we describe, where to place them. From hereon, for a vertex v and two points p and p' , we informally call *triangular pocket rooted at vertex v and supported by $\text{ray}(v, p)$ and $\text{ray}(v, p')$* a sub-polygon w, v, w' (a triangle) such that $\text{ray}(v, w)$ passes through p , $\text{ray}(v, w')$ passes through p' , while w and w' are close to v (sufficiently close not to interfere with the rest of the construction). We say that v is the *root* of the triangular pocket, that we often denote by $\mathcal{P}(v)$. We also say that the pocket $\mathcal{P}(v)$ *points* towards p and p' . It is easy to see that each point that sees v also sees the entire triangular pocket $\mathcal{P}(v)$.

¹ The exact width and length of this pocket are not relevant; the reader may just think of $\mathcal{P}_{j,r}$ as a thin pocket which forces to place a guard on a thin strip whose uppermost boundary is $\ell(\alpha_1^j, \alpha_t^j)$

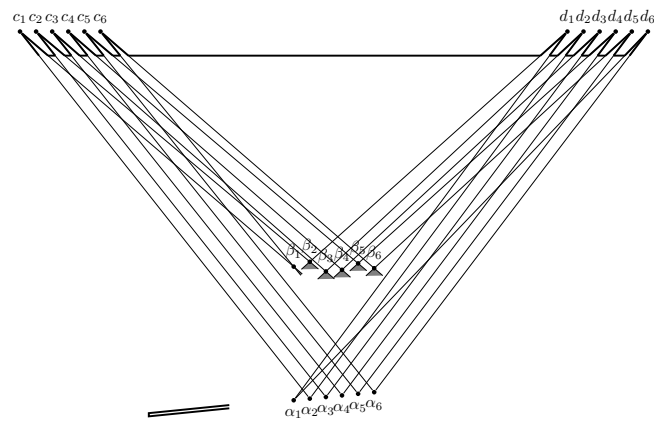
For each A -interval $I_q = [a_i^j, a_{i'}^{j'}] \in \mathcal{S}_A$ we construct one triangular pocket $\mathcal{P}(z_{A,q})$ rooted at vertex $z_{A,q}$ and supported by $\text{ray}(z_{A,q}, \alpha_i^j)$ and $\text{ray}(z_{A,q}, \alpha_{i'}^{j'})$. The placement of this triangular pocket is very far to the right. The x -coordinate of $z_{A,q}$ equals $x_k + (t-1)x + F$, for some large value F to be specified later. The y -coordinate shall be between y_1 and $y_k + (kt-1)y$. We place those $|\mathcal{S}_A|$ pockets along the y -axis, and space them out by some small distance s . To guarantee that we have enough room to place all those pockets, s will be chosen sufficiently small ($s \ll y$).

We will show later, for appropriate values $y \ll x \ll D \ll F$, the only $\alpha_{i''}^{j''}$ seeing vertex $z_{A,q}$ should be the points such that $a_i^j \leq_A \alpha_{i''}^{j''} \leq_A a_{i'}^{j'}$ (see Figure 6).

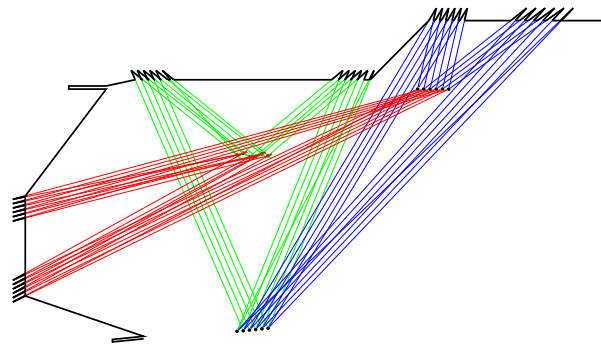
Similarly, we represent each interval $I_q \in \mathcal{S}_B$ by a triangular pocket rooted at $z_{B,q}$. These pockets are placed at the x -coordinate $x_k + (t-1)x + F$ and spaced out by distance s along the y -axis between y -coordinates $y_1 + (kt-1)y + L$ and $y_1 + 2(kt-1)y + L$. The B -interval $I_q = [b_i^j, b_{i'}^{j'}]$ is represented by the triangular pocket $\mathcal{P}(z_{B,q})$ rooted at vertex $z_{B,q}$ supported by $\text{ray}(z_{B,q}, \sigma_j(a_i^j))$ and $\text{ray}(z_{B,q}, \sigma_{j'}(a_{i'}^{j'}))$. Note that $\sigma_j(a_i^j)$ is the point on track B that corresponds to β_i^j . The different values (s, x, y, D, L , and F) introduced so far compare in the following way: $s \ll y \ll x \ll D \ll F$, and $x \ll L \ll F$, see Figure 6.

Now, we describe how we *link* each point α_i^j to its associate β_i^j . For each $j \in [k]$, let us mentally draw $\text{ray}(\alpha_t^j, \beta_1^j)$ and consider points slightly to the left of this ray at a distance, say, L' from point α_t^j . Let us call $\mathcal{R}_{\text{left}}^j$ that informal region of points. Any point in $\mathcal{R}_{\text{left}}^j$ sees, from right to left, in the order α_1^j, α_2^j up to α_t^j , and then, β_1^j, β_2^j up to β_t^j . This observation relies on the fact that $y \ll x \ll L$. So, from the distance, the points $\beta_1^j, \dots, \beta_t^j$ look almost *flat*. It makes the following construction possible. In $\mathcal{R}_{\text{left}}^j$, for each $i \in [t-1]$, we place a triangular pocket $\mathcal{P}(c_i^j)$ rooted at vertex c_i^j and supported by $\text{ray}(c_i^j, \alpha_{i+1}^j)$ and $\text{ray}(c_i^j, \beta_i^j)$. We place also a triangular pocket $\mathcal{P}(c_t^j)$ rooted at c_t^j supported by $\text{ray}(c_t^j, \beta_1^j)$ and $\text{ray}(c_t^j, \beta_t^j)$. We place vertices c_i^j and c_{i+1}^j at the same y -coordinate and spaced out by distance x along the x -axis (see Figure 4). Similarly, let us informally refer to the region slightly to the right of $\text{ray}(\alpha_1^j, \beta_t^j)$ at a distance L' from point α_1^j , as $\mathcal{R}_{\text{right}}^j$. Any point $\mathcal{R}_{\text{right}}^j$ sees, from right to left, in this order β_1^j, β_2^j up to β_t^j , and then, α_1^j, α_2^j up to α_t^j . Therefore, one can place in $\mathcal{R}_{\text{left}}^j$, for each $i \in [t-1]$, a triangular pocket $\mathcal{P}(d_i^j)$ rooted at d_i^j supported by $\text{ray}(d_i^j, \beta_{i+1}^j)$ and $\text{ray}(d_i^j, \alpha_i^j)$. We place also a triangular pocket $\mathcal{P}(d_t^j)$ rooted at d_t^j supported by $\text{ray}(d_t^j, \alpha_1^j)$ and $\text{ray}(d_t^j, \alpha_t^j)$. Again, those t pockets are placed at the same y -coordinate and spaced out horizontally by x (see Figure 4). We denote by $\mathcal{P}_{j,\alpha,\beta}$ the set of pockets $\{\mathcal{P}(c_1^j), \dots, \mathcal{P}(c_t^j), \mathcal{P}(d_1^j), \dots, \mathcal{P}(d_t^j)\}$ and informally call it the *weak point linker* (or simply, *weak linker*) of $\alpha_1^j, \dots, \alpha_t^j$ and $\beta_1^j, \dots, \beta_t^j$. We may call the pockets of $\mathcal{R}_{\text{left}}^j$ (resp. $\mathcal{R}_{\text{right}}^j$) *left* pockets (resp. *right* pockets).

As we will show later, if one wants to guard with only two points all the pockets of $\mathcal{P}_{j,\alpha,\beta} = \{\mathcal{P}(c_1^j), \dots, \mathcal{P}(c_t^j), \mathcal{P}(d_1^j), \dots, \mathcal{P}(d_t^j)\}$ and one first decides to put a guard on point α_i^j (for some $i \in [t]$), then one is not forced to put the other guard on point β_i^j but only on an area whose uppermost point is β_i^j (see the shaded areas below the b_i^j 's in Figure 4). Now, if the points $\beta_1^j, \dots, \beta_t^j$ would all lie on a common line ℓ , we could shrink the shaded area of each β_i^j (Figure 4) down to the single point β_i^j by adding a thin rectangular pocket on ℓ (similarly to what we have for $\alpha_1^j, \dots, \alpha_t^j$). Naturally, we need that $\beta_1^j, \dots, \beta_t^j$ are *not* on a common line to be able to encode the permutation σ_j . The remedy we pursue is the following. For each $j \in [k]$, we allocate t points $\bar{\alpha}_1^j, \bar{\alpha}_2^j, \dots, \bar{\alpha}_t^j$ on a horizontal line, spaced out by distance x , say, $\approx \frac{D}{2}$ to the right and $\approx L$ above of β_t^j . We place a thin horizontal rectangular pocket $\mathcal{P}_{j,\bar{\alpha}}$ of the same dimension as $\mathcal{P}_{j,r}$ such that the lowermost longer side of $\mathcal{P}_{j,\bar{\alpha}}$ is on the line $\ell(\bar{\alpha}_1^j, \bar{\alpha}_t^j)$. We add the $2t$ pockets corresponding to a weak



■ **Figure 4** Weak point linker gadget.

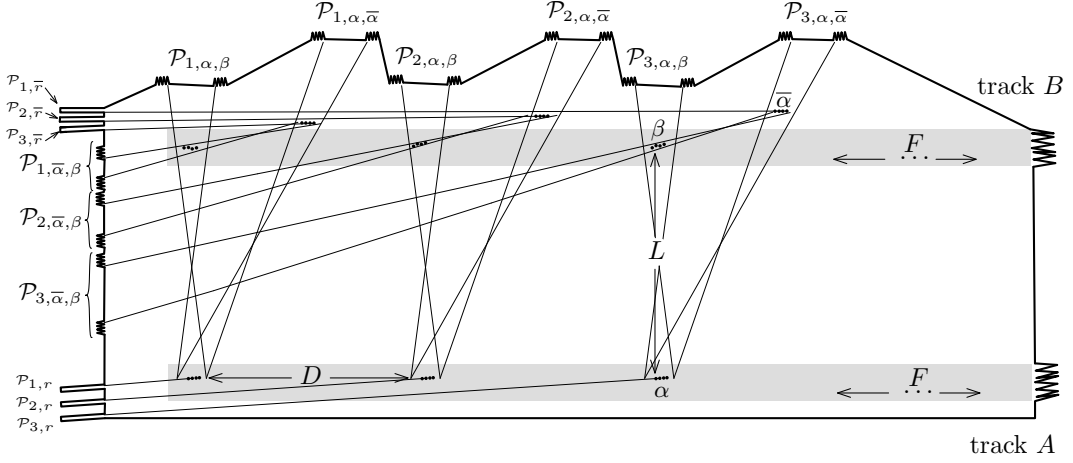


■ **Figure 5** Point linker gadget: a triangle of (three) weak point linkers.

linker $\mathcal{P}_{j,\alpha,\bar{\alpha}}$ between $\alpha_1^j, \dots, \alpha_t^j$ and $\bar{\alpha}_1^j, \dots, \bar{\alpha}_t^j$ as well as the $2t$ pockets of a weak linker $\mathcal{P}_{j,\bar{\alpha},\beta}$ between $\bar{\alpha}_1^j, \dots, \bar{\alpha}_t^j$ and $\beta_1^j, \dots, \beta_t^j$ as pictured in Figure 5. We denote by \mathcal{P}_j the union $\mathcal{P}_{j,r} \cup \mathcal{P}_{j,\bar{r}} \cup \mathcal{P}_{j,\alpha,\beta} \cup \mathcal{P}_{j,\alpha,\bar{\alpha}} \cup \mathcal{P}_{j,\bar{\alpha},\beta}$ of all the pockets involved in the encoding of color class j . Now, say, one wants to guard all the pockets of \mathcal{P}_j with only three points, and chooses to put a guard on α_i^j (for some $i \in [t]$). Because of the pockets of $\mathcal{P}_{j,\alpha,\bar{\alpha}} \cup \mathcal{P}_{j,\bar{r}}$, one is forced to place a second guard precisely on $\bar{\alpha}_i^j$. Now, because of the weak linker $\mathcal{P}_{j,\alpha,\beta}$ the third guard should be on a region whose uppermost point is β_i^j , while, because of $\mathcal{P}_{j,\bar{\alpha},\beta}$ the third guard should be on a region whose lowermost point is β_i^j . The conclusion is that the third guard should be put precisely on β_i^j . This *triangle* of weak linkers is called the *linker* of color class j . The k linkers are placed accordingly to Figure 6. This ends the construction.

Specification of the distances. We can specify the coordinates of positions of all the vertices by fractions of integers. These integers are polynomially bounded in n . If we want to get integer coordinates, we can transform the rational coordinates to integer coordinates by multiplying all of them with the least common multiple of all the denominators, which is not polynomially bounded anymore. The length of the integers in binary is still polynomially bounded.

We can safely set s to one, as it is the smallest length, we specified. We will put $|\mathcal{S}_a|$ pockets on track A and $|\mathcal{S}_b|$ pockets on track B . It is sufficient to have an opening space of one between them. Thus, the space on the right side of \mathcal{P} , for all pockets of track A is bounded by $2|\mathcal{S}_a|$. Thus setting y to $|\mathcal{S}_a| + |\mathcal{S}_b|$ secures us that we have plenty of space to



■ **Figure 6** The overall picture of the reduction with $k = 3$.

place all the pockets. We specify $F = (|\mathcal{S}_a| + |\mathcal{S}_b|)Dk = yDk$. We have to show that this is large enough to guarantee that the pockets on track A distinguish the picked points only by the y -coordinate. Let p and q be two points among the α_i^j . Their vertical distance is upper bounded by Dk and their horizontal distance is lower bounded by y . Thus the slope of $\ell = \ell(p, q)$ is at least $\frac{y}{Dk}$. At the right side of \mathcal{P} the line ℓ will be at least $F \frac{y}{Dk}$ above the pockets of track A. Note $F \frac{y}{Dk} = yDk \frac{y}{Dk} > y^2 > |\mathcal{S}_a|^2 > 2|\mathcal{S}_a|$. The same argument shows that F is sufficiently large for track B.

The remaining lengths x, L, L' , and D can be specified in a similar fashion. For the construction of the pockets, let $s \in \mathcal{S}_a$ be an A -interval with endpoints a and b , represented by some points p and q and assume the opening vertices v and w of the triangular pocket are already specified. Then the two lines $\ell(p, v)$ and $\ell(q, w)$ will meet at some point x to the right of v and w . It is easy to see that x has rational coordinates and the integers to represent them can be expressed by the coordinates of p, q, v , and w . This way, all the pockets can be explicitly constructed using rational coordinates as claimed above.

Soundness. We now show that the reduction is correct. The following lemma is the main argument for the easier implication: *if \mathcal{I} is a YES-instance, then the gallery that we build can be guarded with $3k$ points.*

► **Lemma 6.** $\forall j \in [k], \forall i \in [t]$, the three associate points $\alpha_i^j, \bar{\alpha}_i^j, \beta_i^j$ guard entirely \mathcal{P}_j .

Proof. The rectangular pockets $\mathcal{P}_{j,r}$ and $\mathcal{P}_{j,\bar{r}}$ are entirely seen by respectively α_i^j and $\bar{\alpha}_i^j$. The pockets $\mathcal{P}(c_1^j), \mathcal{P}(c_2^j), \dots, \mathcal{P}(c_{i-1}^j)$ and $\mathcal{P}(d_i^j), \mathcal{P}(d_{i+1}^j), \dots, \mathcal{P}(d_t^j)$ are all entirely seen by α_i^j , while the pockets $\mathcal{P}(c_i^j), \mathcal{P}(c_{i+1}^j), \dots, \mathcal{P}(c_t^j)$ and $\mathcal{P}(d_1^j), \mathcal{P}(d_2^j), \dots, \mathcal{P}(d_{i-1}^j)$ are all entirely seen by β_i^j . This means that α_i^j and β_i^j jointly see all the pockets of $\mathcal{P}_{j,\alpha,\beta}$. Similarly, α_i^j and $\bar{\alpha}_i^j$ jointly see all the pockets of $\mathcal{P}_{j,\alpha,\bar{\alpha}}$, and $\bar{\alpha}_i^j$ and β_i^j jointly see all the pockets of $\mathcal{P}_{j,\bar{\alpha},\beta}$. Therefore, $\alpha_i^j, \bar{\alpha}_i^j, \beta_i^j$ jointly see all the pockets of \mathcal{P}_j . ◀

Assume that \mathcal{I} is a YES-instance and let $\{(a_{s_1}^1, b_{s_1}^1), \dots, (a_{s_k}^k, b_{s_k}^k)\}$ be a solution. We claim that $G = \{\alpha_{s_1}^1, \bar{\alpha}_{s_1}^1, \beta_{s_1}^1, \dots, \alpha_{s_k}^k, \bar{\alpha}_{s_k}^k, \beta_{s_k}^k\}$ guard the whole polygon \mathcal{P} . By Lemma 6, $\forall j \in [k]$, \mathcal{P}_j is guarded. For each A -interval (resp. B -interval) in \mathcal{S}_A (resp. \mathcal{S}_B) there is at least one 2-element $(a_{s_j}^j, b_{s_j}^j)$ such that $a_{s_j}^j \in \mathcal{S}_A$ (resp. $b_{s_j}^j \in \mathcal{S}_B$). Thus, the corresponding pocket is guarded by $\alpha_{s_j}^j$ (resp. $\beta_{s_j}^j$). The rest of the polygon \mathcal{P} (which is not part of pockets)

is guarded by, for instance, $\{\bar{\alpha}_{s_1}^1, \dots, \bar{\alpha}_{s_k}^k\}$. So, G is indeed a solution and it contains $3k$ points.

Assume now that there is no solution to the instance \mathcal{I} of STRUCTURED 2-TRACK HITTING SET. We show that there is no set of $3k$ points guarding \mathcal{P} . We observe that no point of \mathcal{P} sees inside two triangular pockets one being in $\mathcal{P}_{j,\alpha,\gamma}$ and the other in $\mathcal{P}_{j',\alpha,\gamma'}$ with $j \neq j'$ and $\gamma, \gamma' \in \{\beta, \bar{\alpha}\}$. Further, $V(r(\mathcal{P}_{j,\alpha,\beta} \cup \mathcal{P}_{j,\alpha,\bar{\alpha}})) \cap V(r(\mathcal{P}_{j',\alpha,\beta} \cup \mathcal{P}_{j',\alpha,\bar{\alpha}})) = \emptyset$ when $j \neq j'$, where r maps a set of triangular pockets to the set of their root. Also, for each $j \in [k]$, seeing entirely $\mathcal{P}_{j,\alpha,\beta}$ and $\mathcal{P}_{j,\alpha,\bar{\alpha}}$ requires at least 3 points. This means that for each $j \in [k]$, one should place three guards in $V(r(\mathcal{P}_{j,\alpha,\beta} \cup \mathcal{P}_{j,\alpha,\bar{\alpha}}))$. Furthermore, one can observe among those three points one should guard a triangular pocket $\mathcal{P}_{j',r}$ and another should guard $\mathcal{P}_{j'',\bar{r}}$. Let us try to guard entirely \mathcal{P}_1 and two rectangular pockets $\mathcal{P}_{j',r}$ and $\mathcal{P}_{j'',\bar{r}}$, with only three guards. Let call ℓ_1 (resp. ℓ'_1) the line corresponding to the extension of the uppermost (resp. lowermost) longer side of $\mathcal{P}_{1,r}$ (resp. $\mathcal{P}_{1,\bar{r}}$). The only points of \mathcal{P} that can see a rectangular pocket $\mathcal{P}_{j',r}$ and at least t pockets of $\mathcal{P}_{1,\alpha,\bar{\alpha}}$ are on ℓ_1 : more specifically, they are the points $\alpha_1^1, \dots, \alpha_t^1$. The only points that can see a rectangular pocket $\mathcal{P}_{j'',\bar{r}}$ and at least t pockets of $\mathcal{P}_{1,\alpha,\bar{\alpha}}$ are on ℓ'_1 : they are the points $\bar{\alpha}_1^1, \dots, \bar{\alpha}_t^1$. As $\mathcal{P}_{1,\alpha,\bar{\alpha}}$ has $2t$ pockets, one has to take a point α_i^1 and $\bar{\alpha}_{i'}^1$. By the same argument as in Lemma 5, i should be equal to i' (otherwise, $i < i'$ and the left pocket pointing towards $\bar{\alpha}_{i'-1}^1$ and $\alpha_{i'}^1$ is not seen, or $i > i'$ and the right pocket pointing towards α_{i+1}^1 and $\bar{\alpha}_i^1$ is not seen). We now denote by s_1 this shared value. Now, to see the left pocket $\mathcal{P}(c_{s_1}^1)$ and the right pocket $\mathcal{P}(d_{s_1-1}^1)$ (that should still be seen), the third guard should be to the left of $\ell(c_{s_1}^1, \beta_{s_1}^1)$ and to the right of $\ell(d_{s_1-1}^1, \beta_{s_1}^1)$ (see shaded area of Figure 4). That is, the third guard should be on a region in which $\beta_{s_1}^1$ is the uppermost point. The same argument with the pockets of $\mathcal{P}_{1,\bar{\alpha},\beta}$ implies that the third guard should also be on a region in which $\beta_{s_1}^1$ is the lowermost point. Thus, the position of the third guard has to be point $\beta_{s_1}^1$. Therefore, one should put guards on points $\alpha_{s_1}^1, \bar{\alpha}_{s_1}^1$, and $\beta_{s_1}^1$, for some $\alpha_1 \in [t]$.

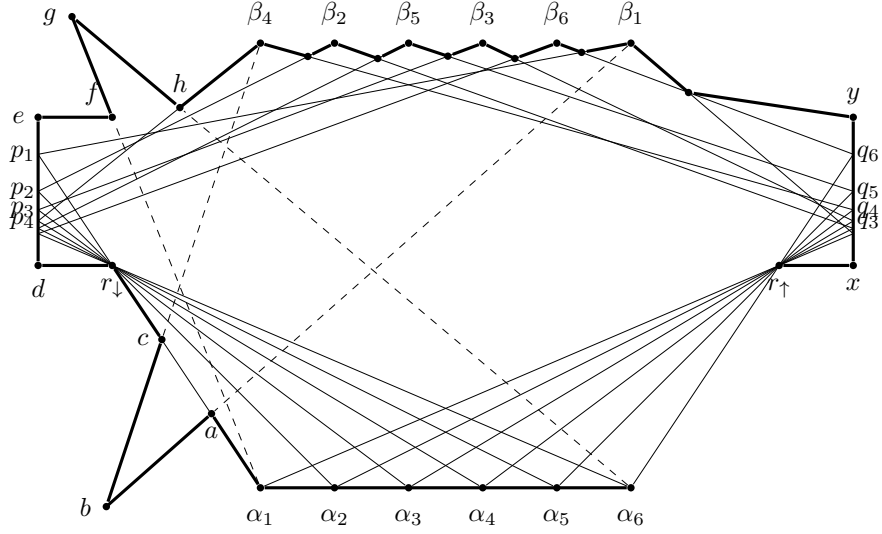
As none of those three points see any pocket $\mathcal{P}_{j,\bar{\alpha},\beta}$ with $j > 1$ (we already mentioned that no pocket of $\mathcal{P}_{j,\alpha,\beta}$ and $\mathcal{P}_{j,\alpha,\bar{\alpha}}$ with $j > 1$ can be seen by those points), we can repeat the argument for the second color class; and so forth up to color class k . Thus, a potential solution with $3k$ guards should be of the form $\{\alpha_{s_1}^1, \bar{\alpha}_{s_1}^1, \beta_{s_1}^1, \dots, \alpha_{s_k}^k, \bar{\alpha}_{s_k}^k, \beta_{s_k}^k\}$. As there is no solution to \mathcal{I} , there should be a set in $\mathcal{S}_A \cup \mathcal{S}_B$ that is not hit by $\{(a_{s_1}^1, b_{s_1}^1), \dots, (a_{s_k}^k, b_{s_k}^k)\}$. By construction, the pocket associated to this set is not entirely seen. ◀

4 Parameterized hardness of the vertex guard variant

We now turn to the vertex guard variant and show the same hardness result. Again, we reduce from STRUCTURED 2-TRACK HITTING SET and our main task is to design a *linker gadget*. Though, *linking* pairs of vertices turns out to be very different from *linking* pairs of points. Therefore, we have to come up with fresh ideas to carry out the reduction. In a nutshell, the principal ingredient is to *link* pairs of convex vertices by introducing reflex vertices at strategic places. As placing guards on those reflex vertices is not supposed to happen in the STRUCTURED 2-TRACK HITTING SET instance, we design a so-called *filter gadget* to prevent any solution from doing so.

► **Theorem 2 (Parameterized hardness vertex guard).** *Assuming the ETH, VERTEX GUARD ART GALLERY is not solvable in time $f(k)n^{o(k/\log k)}$, for any computable function f , even on simple polygons, where n is the number of vertices of the polygon and k is the number of guards allowed.*

Proof. From an instance $\mathcal{I} = (k \in \mathbb{N}, t \in \mathbb{N}, \sigma \in \mathfrak{S}_k, \sigma_1 \in \mathfrak{S}_t, \dots, \sigma_k \in \mathfrak{S}_t, \mathcal{S}_A, \mathcal{S}_B)$, we build



■ **Figure 7** Vertex linker gadget. We omitted the superscript j in all the labels. Here, $\sigma_j(1) = 4$, $\sigma_j(2) = 2$, $\sigma_j(3) = 5$, $\sigma_j(4) = 3$, $\sigma_j(5) = 6$, $\sigma_j(6) = 1$.

a simple polygon \mathcal{P} with $O(kt + |\mathcal{S}_A| + |\mathcal{S}_B|)$ vertices, such that \mathcal{I} is a YES-instance iff \mathcal{P} can be guarded by $3k$ vertices.

Linker gadget. For each $j \in [k]$, permutation σ_j will be encoded by a sub-polygon \mathcal{P}_j that we call *vertex linker*, or simply *linker* (see Figure 7). We regularly set t consecutive vertices $\alpha_1^j, \alpha_2^j, \dots, \alpha_t^j$ in this order, along the x -axis. Opposite to this *segment*, we place t vertices $\beta_{\sigma_j(1)}^j, \beta_{\sigma_j(2)}^j, \dots, \beta_{\sigma_j(t)}^j$ in this order, along the x -axis, too. The $\beta_{\sigma_j(1)}^j, \dots, \beta_{\sigma_j(t)}^j$, contrary to $\alpha_1^j, \dots, \alpha_t^j$, are *not* consecutive; we will later add some reflex vertices between them. At mid-distance between α_1^j and $\beta_{\sigma_j(1)}^j$, to the left, we put a reflex vertex r_{\downarrow}^j . Behind this reflex vertex, we place a vertical wall $d^j e^j$ (r_{\downarrow}^j, d^j , and e^j are three consecutive vertices of \mathcal{P}), so that $\text{ray}(\alpha_1^j, r_{\downarrow}^j)$ and $\text{ray}(\alpha_t^j, r_{\downarrow}^j)$ both intersect $\text{seg}(d^j, e^j)$. That implies that for each $i \in [t]$, $\text{ray}(\alpha_i^j, r_{\downarrow}^j)$ intersects $\text{seg}(d^j, e^j)$. We denote by p_i^j this intersection. The greater i , the closer p_i^j is to d^j . Similarly, at mid-distance between α_t^j and $\beta_{\sigma_j(t)}^j$, to the right, we put a reflex vertex r_{\uparrow}^j and place a vertical wall $x^j y^j$ (r_{\uparrow}^j, x^j , and y^j are consecutive), so that $\text{ray}(\alpha_1^j, r_{\uparrow}^j)$ and $\text{ray}(\alpha_t^j, r_{\uparrow}^j)$ both intersect $\text{seg}(x^j, y^j)$. For each $i \in [t]$, we denote by q_i^j the intersection between $\text{ray}(\alpha_i^j, r_{\uparrow}^j)$ and $\text{seg}(x^j, y^j)$. The smaller i , the closer q_i^j is to x^j .

For each $i \in [t]$, we put around β_i^j two reflex vertices, one in $\text{ray}(\beta_i^j, p_i^j)$ and one in $\text{ray}(\beta_i^j, q_i^j)$. In Figure 7, we merged some reflex vertices but the essential part is that $V(\beta_i^j) \cap \text{seg}(d^j, e^j) = \text{seg}(d^j, p_i^j)$ and $V(\beta_i^j) \cap \text{seg}(x^j, y^j) = \text{seg}(x^j, q_i^j)$. Finally, we add a triangular pocket rooted at g^j and supported by $\text{ray}(g^j, \alpha_1^j)$ and $\text{ray}(g^j, \alpha_t^j)$, as well as a triangular pocket rooted at b^j and supported by $\text{ray}(g^j, \beta_{\sigma_j(1)}^j)$ and $\text{ray}(g^j, \beta_{\sigma_j(t)}^j)$. This ends the description of the vertex linker (see Figure 7).

The following lemma formalizes how exactly the vertices α_i^j and β_i^j are linked: say, one chooses to put a guard on a vertex α_i^j , then the only way to see entirely \mathcal{P}_j by putting a second guard on a vertex of $\{\beta_1^j, \dots, \beta_t^j\}$ is to place it on the vertex β_i^j .

► **Lemma 7.** For any $j \in [k]$, the sub-polygon \mathcal{P}_j is seen entirely by $\{\alpha_v^j, \beta_w^j\}$ iff $v = w$.

Proof. The regions of \mathcal{P}_j not seen by α_v^j (i.e., $\mathcal{P}_j \setminus V(\alpha_v^j)$) consist of the triangles $d^j r_{\downarrow}^j p_v^j$, $x^j r_{\uparrow}^j q_v^j$ and partially the triangle $a^j b^j c^j$. The triangle $a^j b^j c^j$ is anyway entirely seen by the vertex β_i^j , for any $i \in [t]$. It remains to prove that $d^j r_{\downarrow}^j p_v^j \cup x^j r_{\uparrow}^j q_v^j \subseteq V(\beta_w^j)$ iff $v = w$.

It holds that $d^j r_{\downarrow}^j p_v^j \cup x^j r_{\uparrow}^j q_v^j \subseteq V(\beta_v^j)$ since, by construction, the two reflex vertices neighboring β_v^j are such that β_v^j sees $\text{seg}(d^j, p_{\alpha}^j)$ (hence, the whole triangle $d^j r_{\downarrow}^j p_v^j$) and $\text{seg}(x^j, q_{\alpha}^j)$ (hence, the whole triangle $x^j r_{\uparrow}^j q_v^j$). Now, let us assume that $v \neq w$. If $v < w$, the interior of the segment $\text{seg}(p_v, p_w)$ is not seen by $\{\alpha_v^j, \beta_w^j\}$, and if $v > w$, the interior of the segment $\text{seg}(q_v, q_w)$ is not seen by $\{\alpha_v^j, \beta_w^j\}$. ◀

The issue we now have is that one could decide to place a guard on a vertex α_i^j and a second guard on a reflex vertex between $\beta_{\sigma_j(w)}^j$ and $\beta_{\sigma_j(w+1)}^j$ (for some $w \in [t-1]$). This is indeed another way to guard the whole \mathcal{P}_j . We will now describe a sub-polygon \mathcal{F}_j (for each $j \in [k]$) called *filter gadget* (see Figure 8) satisfying the property that all its (triangular) pockets can be guarded by adding only one guard on a vertex of \mathcal{F}_j iff there is already a guard on a vertex β_i^j of \mathcal{P}_j . Therefore, the filter gadget will prevent one from placing a guard on a reflex vertex of \mathcal{P}_j . The functioning of the gadget is again based on Lemma 5.

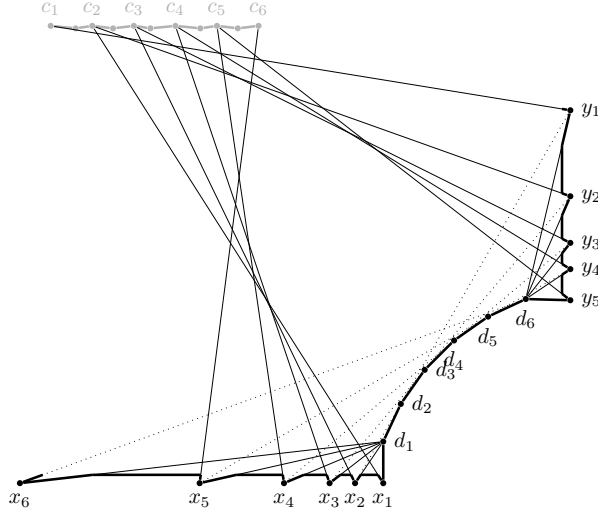
Filter gadget. Let d_1^j, \dots, d_t^j be t consecutive vertices of a regular, say, $20t$ -gon, so that the angle made by $\text{ray}(d_1^j, d_2^j)$ and the x -axis is a bit below 45° , while the angle made by $\text{ray}(d_{t-1}^j, d_t^j)$ and the x -axis is a bit above 45° . The vertices d_1^j, \dots, d_t^j can therefore be seen as the discretization of an arc \mathcal{C} . We now mentally draw two lines ℓ_h and ℓ_v ; ℓ_h is a horizontal line a bit below d_1^j , while ℓ_v is a vertical line a bit to the right of d_t^j . We put, for each $i \in [t]$, a vertex x_i^j at the intersection of ℓ_h and the tangent to \mathcal{C} passing through d_i^j . Then, for each $i \in [t-1]$, we set a triangular pocket $\mathcal{P}(x_i^j)$ rooted at x_i^j and supported by $\text{ray}(x_i^j, d_1^j)$ and $\text{ray}(x_i^j, \beta_{\sigma_j(i+1)}^j)$. For convenience, each point $\beta_{\sigma_j(i)}^j$ is denoted by c_i^j on Figure 8. We also set a triangular pocket $\mathcal{P}(x_t^j)$ rooted at x_t^j and supported by $\text{ray}(x_t^j, d_t^j)$ and $\text{ray}(x_t^j, d_1^j)$. Similarly, we place, for each $i \in [t-1]$, a vertex y_i^j at the intersection of ℓ_v and the tangent to \mathcal{C} passing through d_{i+1}^j . Finally, we set a triangular pocket $\mathcal{P}(y_i^j)$ rooted at y_i^j and supported by $\text{ray}(y_i^j, \beta_{\sigma_j(i)}^j)$ and $\text{ray}(y_i^j, d_t^j)$, for each $i \in [t-1]$ (see Figure 8). We denote by $\mathcal{P}(\mathcal{F}_j)$ the $2t-1$ triangular pockets of \mathcal{F}_j .

► **Lemma 8.** *For each $j \in [k]$, the only ways to see entirely $\mathcal{P}(\mathcal{F}_j)$ and the triangle $a^j b^j c^j$ with only two guards on vertices of $\mathcal{P}_j \cup \mathcal{F}_j$ is to place them on vertices c_i^j and d_i^j (for any $i \in [t]$).*

Proof. Proving this lemma will, in particular, entail that it is not possible to see entirely $\mathcal{P}(\mathcal{F}_j)$ with only two vertices if one of them is a reflex vertex between c_i^j and c_{i+1}^j . Let us call such a vertex an intermediate reflex vertex (in color class j). Because of the pocket $a^j b^j c^j$, one should put a guard on a c_i^j (for some $i \in [t]$) or on an intermediate reflex vertex in class j . As vertices a^j , b^j , and c^j do not see anything of $\mathcal{P}(\mathcal{F}_j)$, placing the first guard at one of those three vertices cannot work as a consequence of what follows.

Say, the first guard is placed at c_i^j ($= \beta_{\sigma_j(i)}^j$). The pockets $\mathcal{P}(x_1^j), \mathcal{P}(x_2^j), \dots, \mathcal{P}(x_{i-1}^j)$ and $\mathcal{P}(y_i^j), \mathcal{P}(y_{i+1}^j), \dots, \mathcal{P}(x_{t-1}^j)$ are entirely seen, while the vertices $x_i^j, x_{i+1}^j, \dots, x_t^j$ and $y_1^j, y_2^j, \dots, y_{i-1}^j$ are not. The only vertex that sees simultaneously all those vertices is d_i^j . The vertex d_i^j even sees the *whole* pockets $\mathcal{P}(x_i^j), \mathcal{P}(x_{i+1}^j), \dots, \mathcal{P}(x_t^j)$ and $\mathcal{P}(y_1^j), \mathcal{P}(y_2^j), \dots, \mathcal{P}(y_{i-1}^j)$. Therefore, all the pockets $\mathcal{P}(\mathcal{F}_j)$ are fully seen.

Now, say, the first guard is put on an intermediate reflex vertex r between c_i^j and c_{i+1}^j (for some $i \in [t-1]$). Both vertices x_i^j and y_i^j , as well as x_t^j , are not seen by r and should



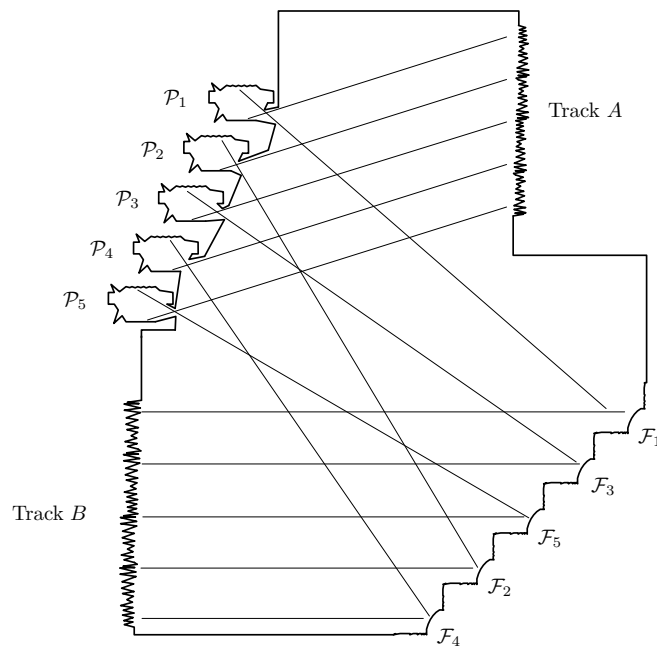
■ **Figure 8** The filter gadget \mathcal{F}_j . Again, we omit the superscript j on the labels. Vertices c_1, c_2, \dots, c_t are not part of \mathcal{F}_j and are in fact the vertices $\beta_{\sigma_j(1)}^j, \beta_{\sigma_j(2)}^j, \dots, \beta_{\sigma_j(t)}^j$ and the vertices in between the c_i 's are the reflex vertices that we have to *filter out*.

therefore be seen by the second guard. However, no vertex simultaneously sees those three vertices. ◀

Putting the pieces together. The permutation σ is encoded the following way. We position the vertex linkers $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_k$ such that \mathcal{P}_{i+1} is below and slightly to the left of \mathcal{P}_i . Far below and to the right of the \mathcal{P}_i 's, we place the \mathcal{F}_i 's such that the uppermost vertex of $\mathcal{F}_{\sigma(i)}$ is close and connected to the leftmost vertex of $\mathcal{F}_{\sigma(i+1)}$, for all $i \in [t-1]$. We add a constant number of vertices in the vicinity of each \mathcal{P}_j , so that the only filter gadget that vertices $\beta_1^j, \dots, \beta_t^j$ can see is $\mathcal{F}_{\sigma(j)}$ (see Figure 9). Similarly to the point guard version, we place vertically and far from the α_i^j 's, one triangular pocket $\mathcal{P}(z_{A,q})$ rooted at vertex $z_{A,q}$ and supported by $\text{ray}(z_{A,q}, \alpha_i^j)$ and $\text{ray}(z_{A,q}, \alpha_{i'}^j)$, for each A -interval $I_q = [a_i^j, a_{i'}^j] \in \mathcal{S}_A$ (Track A). Finally, we place vertically and far from the d_i^j 's, one triangular pocket $\mathcal{P}(z_{B,q})$ rooted at vertex $z_{B,q}$ and supported by $\text{ray}(z_{B,q}, d_i^j)$ and $\text{ray}(z_{B,q}, d_{i'}^j)$, for each B -interval $I_q = [b_{\sigma_j(i)}^j, b_{\sigma_j(i')}^j] \in \mathcal{S}_B$ (Track B). This ends the construction (see Figure 9).

Soundness. We now prove the correctness of the reduction. Assume that \mathcal{I} is a YES-instance and let $\{(a_{s_1}^1, b_{s_1}^1), \dots, (a_{s_k}^k, b_{s_k}^k)\}$ be a solution. We claim that the set of vertices $G = \{\alpha_{s_1}^1, \beta_{s_1}^1, d_{\sigma_1^{-1}(s_1)}^1, \dots, \alpha_{s_k}^k, \beta_{s_k}^k, d_{\sigma_k^{-1}(s_k)}^k\}$ guards the whole polygon \mathcal{P} . Let $z^j := d_{\sigma_j^{-1}(s_j)}^j$ for notational convenience. By Lemma 7, for each $j \in [k]$, the sub-polygon \mathcal{P}_j is entirely seen, since there are guards on $\alpha_{s_j}^j$ and $\beta_{s_j}^j$. By Lemma 8, for each $j \in [k]$, all the pockets of \mathcal{F}_j are entirely seen, since there are guards on $\beta_{s_j}^j = c_{\sigma_j^{-1}(s_j)}^j$ and $d_{\sigma_j^{-1}(s_j)}^j = z^j$. For each A -interval (resp. B -interval) in \mathcal{S}_A (resp. \mathcal{S}_B) there is at least one 2-element $(a_{s_j}^j, b_{s_j}^j)$ such that $a_{s_j}^j \in \mathcal{S}_A$ (resp. $b_{s_j}^j \in \mathcal{S}_B$). Thus, the corresponding pocket is guarded by $\alpha_{s_j}^j$ (resp. $\beta_{s_j}^j$). The rest of the polygon is seen by, for instance, $z^{\sigma(1)}$ and $z^{\sigma(k)}$.

Assume now that there is no solution to the instance \mathcal{I} of STRUCTURED 2-TRACK HITTING SET, and, for the sake of contradiction, that there is a set G of $3k$ vertices guarding \mathcal{P} . For each $j \in [k]$, vertices b^j, g^j , and x_t^j are seen by three disjoint set of vertices. The first



■ **Figure 9** Overall picture of the reduction with $k = 5$.

two sets are contained in the vertices of sub-polygon \mathcal{P}_j and the third one is contained in the vertices of \mathcal{F}_j . Therefore, to see entirely $\mathcal{P}_j \cup \mathcal{P}(\mathcal{F}_j)$, three vertices are necessary. Summing that over the k color classes, this corresponds already to $3k$ vertices which is the size of the supposed set G . Thus, there should *exactly* 3 guards placed among the vertices of $\mathcal{P}_j \cup \mathcal{F}_j$. Therefore, by Lemma 8, there should be an $s_j \in [t]$ such that both $d_{s_j}^j$ and $c_{s_j}^j = \beta_{\sigma_j(s_j)}^j$ are in G . Then, by Lemma 7, a guard should be placed at vertex $\alpha_{\sigma_j(s_j)}^j$. Indeed, the only vertices seeing g^j are f^j, g^j, h^j and a_1^j, \dots, a_t^j ; but, if the third guard is placed at vertex f^j, g^j , or h^j , then vertices β_w^j (with $w \neq \sigma_j(i)$) are not seen. So far, we showed that G should be of the form $\{\alpha_{\sigma_1(s_1)}^1, \beta_{\sigma_1(s_1)}^1, d_{s_1}^1, \dots, \alpha_{\sigma_j(s_j)}^j, \beta_{\sigma_j(s_j)}^j, d_{s_j}^j, \dots, \alpha_{\sigma_k(s_k)}^k, \beta_{\sigma_k(s_k)}^k, d_{s_k}^k\}$. Though, as there is no solution to \mathcal{L} , there should be a set in $\mathcal{S}_A \cup \mathcal{S}_B$ that is not hit by $\{(a_{\sigma_1(s_1)}^1, b_{\sigma_1(s_1)}^1), \dots, (a_{\sigma_k(s_k)}^k, b_{\sigma_k(s_k)}^k)\}$. By construction, the pocket associated to this set is not entirely seen; a contradiction.

Let us bound the number of vertices of \mathcal{P} . Each sub-polygon \mathcal{P}_j or \mathcal{F}_j contains $O(t)$ vertices. *Track A* contains $3|\mathcal{S}_A|$ vertices and *Track B* contains $3|\mathcal{S}_B|$ vertices. Linking everything together requires $O(k)$ additional vertices. So, in total, there are $O(kt + |\mathcal{S}_A| + |\mathcal{S}_B|)$ vertices. Thus, this reduction together with Theorem 4 implies that VERTEX GUARD ART GALLERY is $W[1]$ -hard and cannot be solved in time $f(k)n^{o(k/\log k)}$ for any computable function f , where n is the number of vertices of the polygon and k the number of guards, unless the ETH fails. ◀

Acknowledgment. We thank Saeed Mehrabi and Meirav Zehavi for fruitful discussions.

References

- 1 Jochen Alber and Jirí Fiala. Geometric separation and exact solutions for the parameterized independent set problem on disk graphs. *J. Algorithms*, 52(2):134–151, 2004. doi:10.1016/j.jalgor.2003.10.001.

- 2 Saugata Basu, Richard Pollack, and Marie-Francoise Roy. Algorithms in real algebraic geometry. *AMC*, 10:12, 2011.
- 3 Édouard Bonnet and Tillmann Miltzow. The parameterized hardness of the art gallery problem. *CoRR*, abs/1603.08116, 2016. URL: <http://arxiv.org/abs/1603.08116>.
- 4 Björn Brodén, Mikael Hammar, and Bengt J. Nilsson. Guarding lines and 2-link polygons is apx-hard. In *Proceedings of the 13th Canadian Conference on Computational Geometry, University of Waterloo, Ontario, Canada, August 13-15, 2001*, pages 45–48, 2001. URL: <https://dSPACE.mah.se/handle/2043/6645>.
- 5 Sergio Cabello, Panos Giannopoulos, Christian Knauer, Dániel Marx, and Günter Rote. Geometric clustering: Fixed-parameter tractability and lower bounds with respect to the dimension. *ACM Trans. Algorithms*, 7(4):43, 2011. doi:10.1145/2000807.2000811.
- 6 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.
- 7 Mark de Berg, Hans Bodlaender, and Sándor Kisfaludi-Bak. Connected dominating set in unit-disk graphs is $w[1]$ -hard. In *EuroCG 2016*, 2016.
- 8 Pedro Jussieu de Rezende, Cid C. de Souza, Stephan Friedrichs, Michael Hemmer, Alexander Kröller, and Davi C. Tozoni. Engineering art galleries. *CoRR*, abs/1410.8720, 2014. URL: <http://arxiv.org/abs/1410.8720>.
- 9 Rodney G. Downey and Michael R. Fellows. *Parameterized complexity*. Springer Science & Business Media, 2012.
- 10 Alon Efrat and Sarel Har-Peled. Guarding galleries and terrains. *Inf. Process. Lett.*, 100(6):238–245, 2006. doi:10.1016/j.ipl.2006.05.014.
- 11 Stephan Eidenbenz, Christoph Stamm, and Peter Widmayer. Inapproximability results for guarding polygons and terrains. *Algorithmica*, 31(1):79–113, 2001.
- 12 Steve Fisk. A short proof of Chvátal’s watchman theorem. *J. Comb. Theory, Ser. B*, 24(3):374, 1978. doi:10.1016/0095-8956(78)90059-X.
- 13 Jörg Flum and Martin Grohe. *Parameterized complexity theory, volume xiv of texts in theoretical computer science. an EATCS series*, 2006.
- 14 Subir K. Ghosh. *Visibility algorithms in the plane*. Cambridge University Press, 2007.
- 15 Subir K. Ghosh. Approximation algorithms for art gallery problems in polygons. *Discrete Applied Mathematics*, 158(6):718–722, 2010.
- 16 Panos Giannopoulos and Christian Knauer. Finding a largest empty convex subset in space is $w[1]$ -hard. *CoRR*, abs/1304.0247, 2013. URL: <http://arxiv.org/abs/1304.0247>.
- 17 Panos Giannopoulos, Christian Knauer, and Günter Rote. The parameterized complexity of some geometric problems in unbounded dimension. In *Parameterized and Exact Computation, 4th International Workshop, IWPEC 2009, Copenhagen, Denmark, September 10-11, 2009, Revised Selected Papers*, pages 198–209, 2009. doi:10.1007/978-3-642-11269-0_16.
- 18 Panos Giannopoulos, Christian Knauer, and Sue Whitesides. Parameterized complexity of geometric problems. *Comput. J.*, 51(3):372–384, 2008. doi:10.1093/comjnl/bxm053.
- 19 Matt Gibson, Erik Krohn, and Qing Wang. The VC-dimension of visibility on the boundary of a simple polygon. In *Algorithms and Computation*, pages 541–551. Springer, 2015.
- 20 Alexander Gilbers and Rolf Klein. A new upper bound for the VC-dimension of visibility regions. *Computational Geometry*, 47(1):61–74, 2014.
- 21 Russell Impagliazzo and Ramamohan Paturi. Complexity of k-sat. In *Computational Complexity, 1999. Proceedings. Fourteenth Annual IEEE Conference on*, pages 237–240. IEEE, 1999.
- 22 Gil Kalai and Jiří Matoušek. Guarding galleries where every point sees a large area. *Israel Journal of Mathematics*, 101(1):125–139, 1997.

- 23 Matthew J. Katz and Gabriel S. Roisman. On guarding the vertices of rectilinear domains. *Computational Geometry*, 39(3):219–228, 2008.
- 24 James King. Fast vertex guarding for polygons with and without holes. *Comput. Geom.*, 46(3):219–231, 2013. doi:10.1016/j.comgeo.2012.07.004.
- 25 David G. Kirkpatrick. An $O(lglgOPT)$ -approximation algorithm for multi-guarding galleries. *Discrete & Computational Geometry*, 53(2):327–343, 2015. doi:10.1007/s00454-014-9656-8.
- 26 Erik Krohn and Bengt J. Nilsson. Approximate guarding of monotone and rectilinear polygons. *Algorithmica*, 66(3):564–594, 2013. doi:10.1007/s00453-012-9653-3.
- 27 Dániel Marx. Parameterized complexity of independence and domination on geometric graphs. In *Parameterized and Exact Computation, Second International Workshop, IWPEC 2006, Zürich, Switzerland, September 13-15, 2006, Proceedings*, pages 154–165, 2006. doi:10.1007/11847250_14.
- 28 Dániel Marx. Can you beat treewidth? *Theory of Computing*, 6(1):85–112, 2010. doi:10.4086/toc.2010.v006a005.
- 29 Dániel Marx and Michal Pilipczuk. Optimal parameterized algorithms for planar facility location problems using Voronoi diagrams. In *ESA 2015*, pages 865–877, 2015. doi:10.1007/978-3-662-48350-3_72.
- 30 Rajeev Motwani, Arvind Raghunathan, and Huzur Saran. Covering orthogonal polygons with star polygons: The perfect graph approach. *J. Comput. Syst. Sci.*, 40(1):19–48, 1990. doi:10.1016/0022-0000(90)90017-F.
- 31 Rolf Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.
- 32 Joseph O’Rourke. *Art gallery theorems and algorithms*, volume 57. Oxford University Press Oxford, 1987.
- 33 Thomas C. Shermer. Recent results in art galleries [geometry]. *Proceedings of the IEEE*, 80(9):1384–1399, 1992.
- 34 Jorge Urrutia et al. Art gallery and illumination problems. *Handbook of computational geometry*, 1(1):973–1027, 2000.
- 35 Pavel Valtr. Guarding galleries where no point sees a small area. *Israel Journal of Mathematics*, 104(1):1–16, 1998.