On Colourability of Polygon Visibility Graphs*

Onur Çağırıcı[1], Petr Hliněný[2], and Bodhayan Roy[3]

1 **Faculty of Informatics, Masaryk University Brno, Czech Republic**
onur@mail.muni.cz
2 **Faculty of Informatics, Masaryk University Brno, Czech Republic**
hlineny@fi.muni.cz
3 **Faculty of Informatics, Masaryk University Brno, Czech Republic**
b.roy@fi.muni.cz

Abstract

We study the problem of colouring the visibility graphs of polygons. In particular, we provide a polynomial algorithm for 4-colouring of the polygon visibility graphs, and prove that the 6-colourability question is already NP-complete for them.

1998 ACM Subject Classification G.2.2 Graph algorithms, F.2.2 Geometrical problems and computations

Keywords and phrases polygon visibility graph, graph coloring, NP-completeness

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2017.21

1 Introduction

Visibility graphs are widely studied graph classes in computational geometry. Geometric sets such as sets of points or line segments, polygons, polygons with obstacles, etc., all can correspond to specific visibility graphs, and have uses in robotics, signal processing, security paradigms, decomposing shapes into clusters [1, 2, 7, 11, 15]. Here we study the visibility graphs of simple polygons in the Euclidean plane and henceforth in the paper, all polygons are simple unless stated otherwise.

Given an n-vertex polygon P (not necessarily convex) in the plane, two points p and q of P are said to be *mutually visible* if, and only if the line segment $\overline{pq}$ does not intersect the exterior of P. The n-vertex visibility graph $G(V, E)$ of P is defined as follows. The vertex set V of G contains a vertex v_i if, and only if, the polygon P contains the point p_i as its vertex. The edge set E of G contains an edge $\{v_i, v_j\}$ if, and only if, the points p_i and p_j are mutually visible. Given a polygon P in the plane, we can compute its visibility graph G in $\mathcal{O}(n^2)$ time using the polygon triangulation method [8, 17]. Hence, in this paper, we slightly abuse notation by not distinguishing between a polygon P and its visibility graph G and referring to a polygon vertex p_i as to the corresponding G-vertex v_i.

Visibility graphs of polygons have been studied with respect to various theoretical and practical computational problems. The complexities of several popular optimization problems have been determined for visibility graphs of polygons. A geometric variation of the dominating set problem, namely polygon guarding, is one of the most studied problems in computational geometry and is known as the Art Gallery Problem [15]. It has been studied extensively for both polygons with and without holes and has been found to be NP-hard in both cases [12, 16]. Besides, given a polygon, computing a maximum independent set is

* P. Hliněný and O. Çağırıcı are supported by the Czech Science Foundation, project no. 17-00837S.

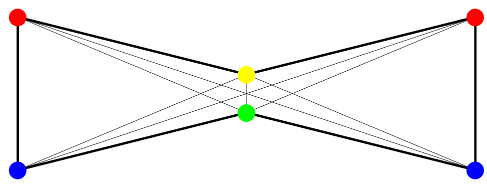

Figure 1 A visibility graph that is without a K_5, non planar but is 4-colourable.

known to be hard, due to Shermer [20], while computing a maximum clique has been shown to be in polynomial time by Ghosh et al. [19].

A *proper vertex colouring* of a graph is an assignment of labels or colours to the vertices of the graph so that no two adjacent vertices have the same colours. Henceforth, when we say colouring a graph, we refer to proper vertex colouring. The *chromatic number* of a graph is defined as the minimum number of colours used in any proper colouring of the graph. Visibility graph colouring has been studied for various types of visibility graphs. Babbitt et al. gave upper bounds for the chromatic numbers of k-visibility graphs of arcs and segments [3]. Kára et al. characterized 3-colourable visibility graphs of point sets and described a super-polynomial lower bound on the chromatic number with respect to the clique number of visibility graphs of point sets [10]. Pfender showed that, as for general graphs, the chromatic number of visibility graphs of point sets is also not upper-bounded by their clique numbers [18]. Diwan and Roy showed that for visibility graphs of point sets, the 5-colouring problem is NP-hard, but 4-colouring is solvable in polynomial time [5].

The problem of *colouring* the visibility graphs of given polygons has been studied in the special context where each internal point of the polygon is seen by a vertex, whose colour appears exactly once among the vertices visible to that point [4,6,9]. However, little is known on colouring visibility graphs of polygons without such constraints. Although 3-colouring is NP-hard for general graphs [14], in particular it is rather trivial to solve it for visibility graphs of polygons in polynomial time using a greedy approach. Already with 4 colours the same question has been open so far.

In this paper we settle (nearly in full) the complexity question of the general problem of colouring polygonal visibility graphs, which was declared open in 1995 by Lin and Skiena [13]. We provide a polynomial-time algorithm to find a 4-colouring of the visibility graph of a given polygon, if such a colouring exists. On the other hand, we provide a reduction showing that the question of k-colourability of the visibility graph of a given simple polygon is NP-complete for any $k \geq 6$. Only the problem of 5-colourability is left open.

2 4-Colouring visibility graphs

In this section, we study the algorithmic question of 4-colourability of the visibility graph of a given polygon. The full structure of 4-colourable visibility graphs is not yet known and it seems to be non-trivial. For instance, if a visibility graph is planar, it is obviously 4-colourable. Though, if such a graph contains K_5, then it is neither planar nor 4-colourable, but a visibility graph not containing any K_5 may be non-planar yet 4-colourable (Figure 1).

The related algorithmic problem of 3-colouring visibility graphs is rather easy to resolve as follows. Every simple polygon can be triangulated and, in such a triangulation, every non-boundary edge is contained in two triangles. One can then proceed greedily edge by edge: Suppose a triangle has already been coloured, and it shares an edge with a triangle that is not fully coloured. Then the two end vertices of the shared edge uniquely determine the colour of the third vertex of the uncoloured triangle.

Our algorithm essentially generalizes the 3-colouring method for 4-colouring. We first divide the polygon into *reduced polygons*. A polygon P is called a reduced polygon, if every chord of P is intersected by another chord of P. After the division, we find and colour a triangle (a K_3 subgraph) with three distinct colours in each reduced subpolygon. Subsequently, whenever we find an uncoloured vertex v adjacent to some three vertices coloured with three distinct colours (such as, to an already coloured triangle), we can uniquely colour also v, by the fourth colour. We will show that we can exhaust all vertices of a reduced subpolygon in this manner. Furthermore, we check for possible colouring conflicts – since the colouring process is unique, this suffices to solve 4-colourability.

Altogether, this will lead to the following theorem.

▶ **Theorem 1.** *The 4-colourability problem is decidable in polynomial time for visibility graphs of simple polygons, and if a 4-colouring exists, then it can be computed in polynomial time.*

In the coming proof, consider a polygon P and its visibility graph $G(V, E)$, embedded on P. Hereafter we slightly abuse notation by equating P and G. Since we want to 4-colour P, we assume that G has no K_5 (or we answer 'no'). We denote the clockwise polygonal chain of P from a vertex u to a vertex v as $\Gamma(u, v)$.

One can easily see that it is enough to focus on reduced P in our proofs. Indeed, assume an edge uv of G which is a chord of P and not crossed by any other chord. We can partition P into subpolygons P_1 and P_2, where $P_1 = (u\,\Gamma(u, v)\,v)$ and $P_2 = (v\,\Gamma(v, u)\,u)$. Since no edge of G has one end in $P_1 \setminus P_2$ and the other in $P_2 \setminus P_1$, the polygons P_1 and P_2 can be 4-coloured separately and merged again (provided that P is 4-colourable).

Let u and v be two vertices of P. The *shortest path* between u and v is a (graph) path from u to v in G such that the sum of the Euclidean lengths of its edges is minimized. Such a shortest path between u and v is unique in P and is denoted as $\Pi(u, v)$. Observe that all non-terminal vertices of a shortest path are non-convex [7]. We will assume an implicit ordering of vertices on $\Pi(u, v)$ from u to v. When we say that some vertex w is the first (or last) vertex on $\Pi(u, v)$ with a certain property, we mean that w precedes (respectively, succeeds) all other vertices with that property on $\Pi(u, v)$.

For a proof of Theorem 1, we have got the following sequence of claims. Consider, in all of them, a K_5-free reduced polygon P and its three vertices t_1, t_2, t_3 forming a triangle $T \subseteq G$. Assume that T is already coloured (which is unique up to a permutation of the colours). Suppose that v_i is an uncoloured vertex, such that an edge incident to v_i intersects T. Then we have the following lemmas.

▶ **Lemma 2.** *Assume that two vertices $v_i \in \Gamma(t_1, t_2)$ and $v_j \in \Gamma(t_2, t_3)$ see each other, and the edge v_iv_j intersects t_1t_2 and t_2t_3. Then the colours of all vertices on the four paths $\Pi(t_1, v_i)$, $\Pi(t_2, v_i)$, $\Pi(t_2, v_j)$ and $\Pi(t_3, v_j)$, including v_i, v_j themselves, are uniquely determined by the colours of T.*

Proof. We prove the claim by induction on the four paths. As the base case, the first vertices of these paths are the vertices of T, which are already assigned different colours.

For the induction step, assume that $\Pi(t_1, v_i)$, $\Pi(t_2, v_i)$, $\Pi(t_2, v_j)$ and $\Pi(t_3, v_j)$ have been coloured till vertices v_a, v_b, v_c and v_d respectively. Also, their immediate uncoloured successors on $\Pi(t_1, v_i)$, $\Pi(t_2, v_i)$, $\Pi(t_2, v_j)$ and $\Pi(t_3, v_j)$ are v_p, v_q, v_r and v_s respectively. We aim to show that the colours of at least one of v_p, v_q, v_r and v_s is uniquely determined by the already coloured vertices.

If v_p does not see v_b and any predecessor of v_b on $\Pi(t_2, v_i)$, then v_q must see v_a or some predecessor of v_a on $\Pi(t_1, v_i)$. We have the following cases.

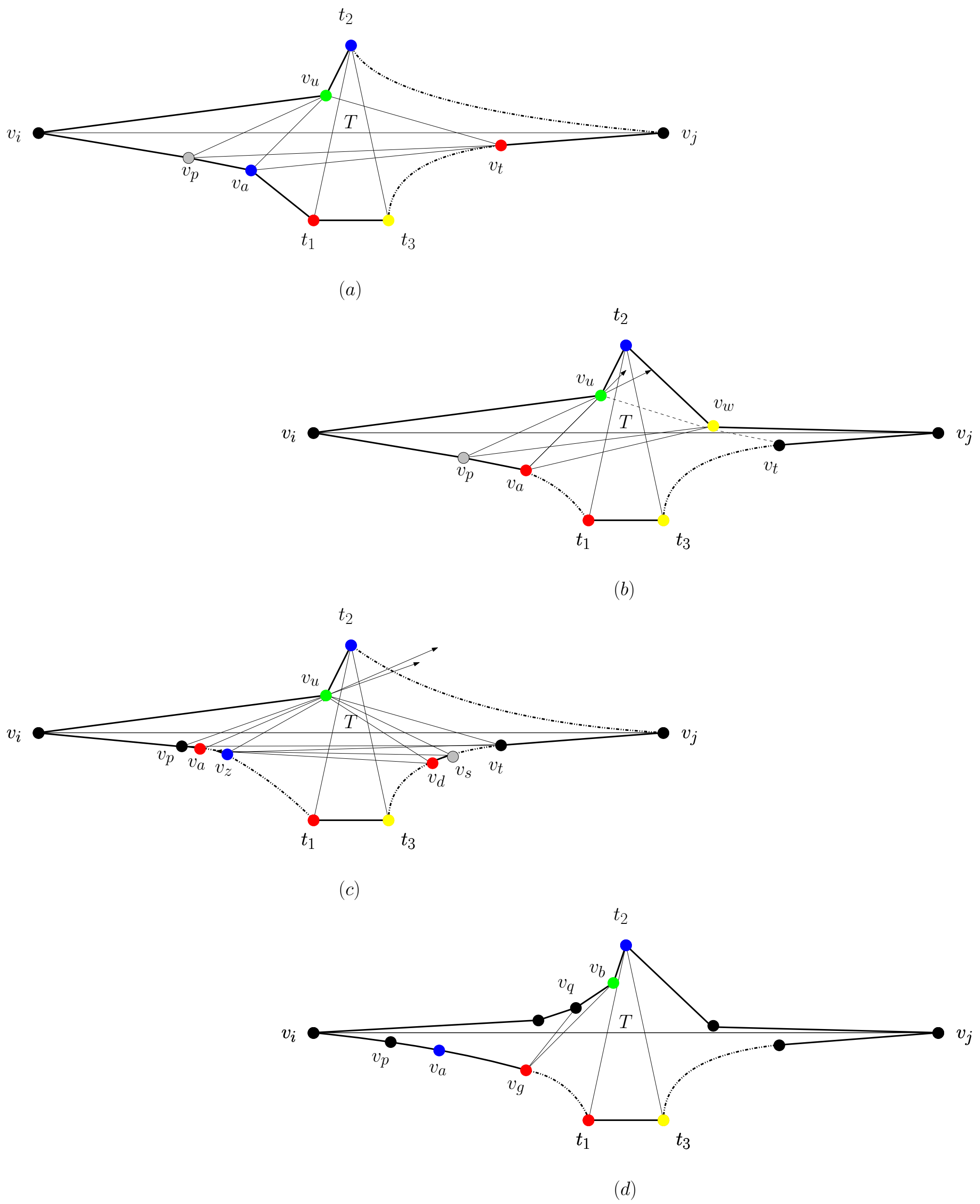

Figure 2 Illustration of the proof of Lemma 2: The vertices whose colours shall be uniquely determined next, are now drawn in gray. Polygonal boundaries containing multiple vertices not included in the figure are drawn with dashed lines. (a) v_p forms a K_4 with v_a, v_t and v_u. (b) v_p forms a K_4 with v_a, v_u and v_w. (c) v_s forms a K_4 with v_u, v_d and v_z. (d) v_g, v_q and v_b form a K_3.

Case 1: v_p sees v_b or some predecessor of $v_b$on $\Pi(t_2, v_i)$.

By definition, v_p is the immediate successor of v_a on $\Pi(t_1, v_i)$, so v_p must see v_a. The right tangent of v_a to $\Pi(t_2, v_i)$ lies to the right of the right tangent of v_p to $\Pi(t_2, v_i)$. So, if the right tangent of v_p to $\Pi(t_2, v_i)$ touches $\Pi(t_2, v_i)$ at a point v_u, then v_a sees v_u. Note that either $v_u = v_b$ or v_u precedes v_b on $\Pi(t_2, v_i)$. In any case, v_u is already coloured. Since v_p,

v_a and $\Pi(t_3, v_j)$ lie on the same side of v_iv_j, and v_p is nearer to v_iv_j than v_a is, v_p and v_a see a vertex v_t of $\Pi(t_3, v_j)$. If v_u also sees v_t, and v_t is already coloured, then the claim is proved (Figure 2(a)). So we consider the other two cases, namely, that either v_u does not see v_t, or v_t is not yet coloured.

Subcase 1.a: v_u does not see v_t. Since v_t and v_u lie on different sides of v_iv_j, some vertex of $\Pi(t_2, v_j)$ must be blocking v_u and v_t. Let v_w be the first vertex of $\Pi(t_2, v_j)$ blocking v_u and v_t. Then v_u sees v_w. The vertex v_w is closer to v_iv_j than v_u is. Also, v_w lies to the right of $\overrightarrow{v_av_u}$ and $\overrightarrow{v_pv_u}$, and to the left of $\overrightarrow{v_av_t}$ and $\overrightarrow{v_pv_t}$. Then the only possible blockers between v_w and v_p or v_a can be from $\Pi(t_2, v_i)$. But all the vertices on $\Pi(t_2, v_i)$ preceding v_u are further from v_iv_j than v_u is. So, there can be no such blocker, and v_w must be visible from both v_a and v_p. If v_w is already coloured, then the claim is proved (Figure 2(b)). Suppose that v_w is not already coloured. Then consider v_r, which precedes v_w on $\Pi(t_2, v_j)$. The vertices v_r and v_c are consecutive on $\Pi(t_2, v_j)$ and hence see each other. Since $\Pi(t_2, v_j)$ and $\Pi(t_1, v_i)$ are on opposite sides of v_iv_j, the vertices v_c and v_r either see v_a or a vertex preceding v_a on $\Pi(t_1, v_i)$. Let v_x be the last coloured vertex of $\Pi(t_1, v_i)$ seen by both v_c and v_r. If $v_x \neq v_a$ then let v_y be the last vertex of $\Pi(t_2, v_i)$ that blocks v_c from the successor of v_y on $\Pi(t_1, v_i)$. Then v_y must be visible from v_x, v_r and v_c. Since v_x precedes v_a on $\Pi(t_1, v_i)$, and v_y precedes v_b on $\Pi(t_2, v_i)$, both v_x and v_y must be already coloured. So, T uniquely determines the colour of v_r. If $v_x = v_a$ then since v_u is on the right tangent of v_a to $\Pi(t_2, v_i)$, both v_c and v_r see v_u. Hence, T uniquely determines the colour of v_r. Now we move to the second subcase.

Subcase 1.b: v_u sees v_t, but v_t is not yet coloured. Since v_u sees v_t, $\Pi(t_2, v_j)$ is a concave chain and the edge t_1t_3 exists in P, v_u must see every predecessor of v_t on $\Pi(t_2, v_j)$. This means that both v_d and v_s see v_u. The vertex v_s must see the vertex (say, v_y) of $\Pi(t_1, v_i)$ where the right tangent from v_d touches $\Pi(t_1, v_i)$, because the last vertices v_i and v_j of concave chains $\Pi(t_1, v_i)$ and $\Pi(t_3, v_j)$ see each other. Also, the left tangent of v_u to $\Pi(t_1, v_i)$ must touch $\Pi(t_1, v_i)$ at a vertex equal to or preceding v_y. Thus, all three of v_s, v_d and v_u see a common vertex v_z on $\Pi(t_1, v_i)$ which precedes v_a, since v_u and v_t see v_a. Thus, v_z is already coloured, and v_u, v_d and v_z form a K_4 with v_s and uniquely determine the colour of v_s (Figure 2(c)).

Case 2: v_p does not see v_b or some predecessor of v_b on $\Pi(t_2, v_i)$.

Here, we have the opposite situation to Case 1. Then v_q sees v_a or some predecessor of v_a on $\Pi(t_1, v_i)$. Let the left tangents from v_q and v_b touch $\Pi(t_2, v_i)$ at v_e and v_f respectively. Either v_b sees v_e or v_q must be a blocker between v_b and v_e. In this case, v_b and v_q see the last vertex v_g of $\Pi(t_1, v_i)$ that is not blocked from v_b by v_q. Since v_b does not see v_p, the vertex v_g must be already coloured (Figure 2(d)). Then, again, v_b and v_q see a vertex v_t on $\Pi(t_2, v_j)$. Now, some already coloured vertex v_u in $\Pi(t_2, v_i)$, adjacent to v_b and v_q might also see v_t, which may or may not be coloured. Or else, v_t might be blocked from such a vertex v_u by a vertex v_w of $\Pi(t_3, v_j)$. It can be seen that each of these arguments can be augmented similar to the subcases of Case 1, a K_4 can be found and the colour of one of the vertices v_p, v_q, v_r and v_s can be uniquely determined. ◀

▶ **Corollary 3.** *If any vertex v_i of P sees a vertex of T and their visibility edge crosses one of the edges of T, then the colour of v_i is uniquely determined by the colours of T.*

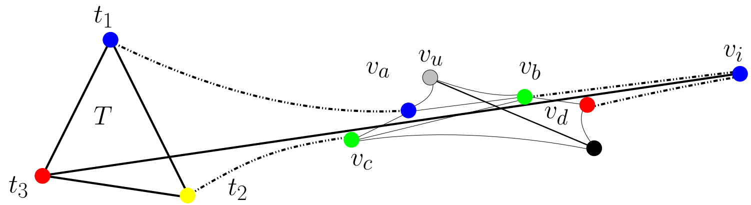

Figure 3 The vertex v_a has an edge incident to one of the vertices of v_a, v_b, v_c, where v_a, v_b and v_c lie on the already coloured shortest paths from t_1 and t_3 to v_i.

Proof. Without loss of generality, suppose that v_i sees t_1, and v_it_1 crosses t_2t_3. Then $v_j = t_1$, $\Pi(t_2, v_j) = t_2t_1$ and $\Pi(t_1, v_j) = t_1$, and Lemma 2 proves the claim. ◀

▶ **Lemma 4.** *If a reduced polygon is 4-colourable, then it has a unique 4-colouring (up to permutation of colours).*

Proof. Consider a triangle T in a reduced polygon P. If P is not just T, then at least one edge of T is not a boundary edge of P. Without loss of generality, let t_1t_2 be such an edge. Since P is reduced, there must be a vertex on $\Gamma(t_1, t_2)$ such that an edge incident to v_i crosses t_1t_2. By Lemma 2 and Corollary 3, if P is 4-colourable, then all vertices on the paths $\Pi(t_1, v_i)$ and $\Pi(t_2, v_i)$, including v_i have a 4-colouring uniquely determined by T. In case t_2t_3 or t_3t_1 are not boundary edges of P, we can similarly find v_j on $\Gamma(t_2, t_3)$ and v_k on $\Gamma(t_3, t_1)$ and uniquely 4-colour $\Pi(t_2, v_j)$, $\Pi(t_3, v_j)$, $\Pi(t_3, v_k)$ and $\Pi(t_1, v_k)$. Now, all the remaining uncoloured vertices of P are on polygonal chains of the form $\Gamma(v_a, v_b)$, where v_a and v_b are two consecutive vertices in one of the six paths mentioned above. Furthermore, no vertex in the polygonal chain $\Gamma(v_a, v_b)$, other than v_a and v_b, is coloured. Without loss of generality, let v_a and v_b be two consecutive vertices on $\Pi(t_1, t_2)$. If v_av_b is not a boundary edge of P, then since P is reduced, there must be an uncoloured vertex v_u in $\Gamma(v_a, v_b)$ such that an edge incident to v_u crosses v_av_b. This edge is either incident to a vertex of $\Pi(t_2, v_i)$, or crosses an edge of $\Pi(t_2, v_i)$. Consider the case where such an edge to a vertex of $\Pi(t_2, v_i)$ exists. Then consider a vertex v_w that is closest to v_av_b among all the vertices of $\Pi(t_2, v_i)$ that see a vertex (say, v_z) of $\Gamma(v_a, v_b)$. Since the edge v_wv_z exists, v_w cannot be blocked by any vertex of $\Pi(t_1, v_i)$. Due to the choice of v_w, no vertex of $\Pi(t_2, v_i)$ can block v_w from v_a or v_b. So, v_w sees both v_a and v_b. Now consider the case where no vertex of $\Gamma(v_a, v_b)$ sees any vertex of $\Pi(t_2, v_i)$, but some vertex of $\Gamma(v_a, v_b)$ sees some vertex of $\Gamma(v_c, v_d)$, where v_c and v_d are consecutive points on $\Pi(t_2, v_i)$. Without loss of generality, assume that v_c precedes v_d in $\Pi(t_2, v_i)$. Then v_c must see both v_a and v_b (Figure 3), for otherwise a vertex of $\Gamma(v_a, v_b)$ must have an edge with some vertex of $\Pi(t_2, v_i)$ acting as a blocker for v_c, contrary to our assumption. Then, in the above two cases, based on the triangle $v_av_bv_w$ and $v_av_bv_c$, respectively, again Lemma 2 and Corollary 3 can be used to uniquely determine a 4-colouring for $\Pi(v_a, v_u)$ and $\Pi(v_b, v_u)$.

Now we generalize the above procedure. Let $T_0 = \{T\}$, and $S_0 = \{\Pi(t_1, v_i), \Pi(t_2, v_i), \Pi(t_2, v_j), \Pi(t_3, v_j), \Pi(t_3, v_k), \Pi(t_1, v_k)\}$. Note that we have assumed that none of the edges of T are boundary edges. If some edges of T are boundary edges then S_0 will have less elements. By the above procedure, we can uniquely 4-colour all the vertices of all elements of S_0. Now, all the uncoloured vertices lie on $\Gamma(v_a, v_b)$, where v_a and v_b are consecutive vertices of some element of S_0. For each such v_av_b, we find a new triangle $v_av_bv_c$ or $v_av_bv_d$, and two new shortest paths of the form $\Pi(v_a, v_u)$ and $\Pi(v_b, v_u)$. Let T_1 denote the set of all such new triangles, and S_1 denote the set of all new shortest paths obtained from T_0 and S_0. Now, the remaining uncoloured vertices must line on polygonal chains of the form $\Gamma(v_e, v_f)$ where v_e and v_f are two consecutive vertices of some element of S_1. In general, following the same method we can always construct T_{i+1} and S_{i+1} from T_i and S_i, until all vertices of P are coloured. Since in each step, the colours of vertices are uniquely determined, it follows that if P has a 4-colouring, then it must be unique. ◀

Our algorithm to decide 4-colourablity of visibility graphs of polygons is given next.

Algorithm 1: Algorithm to decide 4-colourablity of visibility graphs of polygons

```
Input: A simple polygon P with the visibility edges
Output: If P is 4-colourable or not. If so, then proper 4-colouring of P.
Decompose P into reduced subpolygons P_1, ..., P_k;
foreach reduced subpolygon P_i do
    Locate a triangle;
    repeat
        Compute a 4-colouring for vertices on the polygonal chain of each
          non-boundary edge of the triangle;
        /* Using the method of Lemma 2 and Corollary 3                    */
        Continue the process using the method of Lemma 4;
    until Each vertex in P_i is coloured;
end
if two adjacent vertices receive the same colour then
    Output 'non-4-colourable';
    Terminate;
end
Rejoin the reduced subpolygons such that each pair P_i, P_j of subpolygons having a
  common edge have exactly two vertices in common;
Permute the colours of the vertices so that there is no conflict.
```

Now, in light of the above Algorithm 1, we prove Theorem 1.

Proof of Theorem 1: Lemma 2 and Corollary 3 colour the shortest path from a triangle to a vertex uniquely with 4 colours. Lemma 4 repeats the process exhausting all vertices. Since the colour of each vertex is uniquely determined by some three previously coloured vertices, the resulting 4-colouring is unique, if P is 4-colourable. Consequently, if a conflict is found, it follows that P is not 4-colourable. So, the algorithm is correct.

Let the number of vertices and edges in G be n and m respectively. The chords that do not cross any other chord, can be found in $O(m^2)$ time. Thus, the decomposition of P into reduced subpolygons takes $O(m^2)$ time. Shortest paths from a triangle to a vertex can be found in $O(n)$ time. While computing the colouring on the shortest paths, a pointer can be kept on each of the shortest paths, and the colouring takes $O(n)$ time. The colouring step can be iterated at most once for each vertex, so the complexity for all vertices is $O(n^2)$. Checking for conflict takes $O(m)$ time. Finally, rejoining the reduced subpolygons takes $O(n)$ time. Thus, the complexity of the algorithm is $O(m^2)$. ◀

3 Hardness of 6-colourability

In this section we prove that the problem of deciding whether the visibility graph G of a given simple polygon P can be properly coloured with 6 colours, is NP-complete.

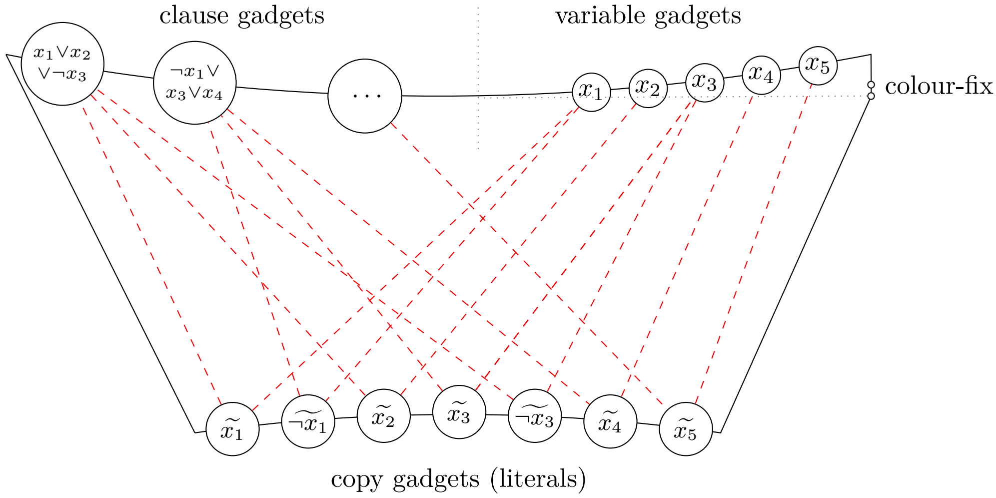

Figure 4 A scheme of the polygon P constructed from a 3-SAT formula in Section 3. Note that the top and bottom part are placed on slightly concave arcs, which block undesired visibilities between gadgets. The colour-fix gadget is placed so that it can see none of the clause gadgets. The red dashed lines show "visibility communication" between related variable and copy gadgets (the literals), and between related copy and clause gadgets.

Membership of our problem in NP is trivial (since G can be efficiently computed from P and then a colouring checked on G). We are going to present a polynomial reduction from the NP-hard problem of *Not-all-equal 3-SAT*: Given is a formula Φ in the conjunctive normal form, such that every clause of Φ contains exactly 3 literals, and the task is to find a (satisfying) assignment to the variables of Φ such that every clause contains at least one true and at least one false literal. For that we will construct a polygon P such that its proper 6-colourings correspond to satisfying assignments of Φ. We start with a rough informal outline of the construction.

- Our polygon P will consist of one *colour-fixing* gadget, a series of *variable* gadgets (one per each variable of Φ), a series of *copy* gadgets (one per each literal occurring in Φ), and a series of *clause* gadgets (one per each clause of Φ). Visibility edges will allow "communication" between variable gadgets and their corresponding copies representing the literals, and between the literals and their clause gadgets. Apart from that, there will be no other visibility relation between "internal" vertices of our gadgets. See Figure 4.
- Assume that the visibility graph G of P can be properly 6-coloured. The role of the colour-fixing gadget is to fix these six colours so that precisely two of them, named here as *red* and *blue*, can be used to colour the vertices representing the variables of Φ. The remaining four colours play an auxiliary role; they are used to colour those vertices which "separate" the gadgets from each other, or to "moderate" clause gadgets. More specifically, *yellow* and *orange* colour separating vertices at the variable and clause side ("top"), and *light* and *dark green* colour separating vertices of the copy gadgets ("bottom").
- For each variable x_i of Φ, there will be a variable gadget $R(x_i)$ which, in particular, contains two mutually visible internal vertices named x_i and $\neg x_i$. They must hence be coloured red and blue, or blue and red, encoding the logical value of x_i in Φ. There is no direct influence between colouring decisions of distinct variable gadgets.

- For each literal ℓ occurring in Φ (such as $\ell = x_j$ or $\ell = \neg x_j$ for some variable x_j), there will be a copy gadget $P(\ell)$ which, in particular, contains an internal vertex named $\widetilde{\ell}$. Visibility between the gadgets $R(x_j)$ and $P(\ell)$ are adjusted so that $\widetilde{\ell}$ must receive, in any 6-colouring of G, the same colour as that of ℓ in $P(\ell)$. Furthermore, the point $\widetilde{\ell}$ is positioned so that it is visible only from selected vertices of the corresponding clause gadget of ℓ, as specified later (note that different literals of x_j have separate copy gadgets).
- For each clause $c = \ell_1 \vee \ell_2 \vee \ell_3$ of Φ, there will be a clause gadget $S(c)$ whose vertices can selectively see, among all internal vertices of all the copy gadgets, exactly the points $\widetilde{\ell}_1$, $\widetilde{\ell}_2$, $\widetilde{\ell}_3$. This selected visibility is such that, locally, the clause gadget $S(c)$ can be properly coloured iff not all three points $\widetilde{\ell}_1$, $\widetilde{\ell}_2$, $\widetilde{\ell}_3$ have the same colour. Furthermore, for any satisfying assignment of Φ, proper colourings of all the clause gadgets can be properly combined together.

Note also that, within the presented reduction scheme, 6 colours is a necessary minimum. We need two colours for the separating vertices of the top part, another two such at the bottom part, and then two more colours are required to encode logical values of the variables.

Altogether, this will lead to the following:

▶ **Theorem 5.** *The problem – given a simple polygon P in the plane, to decide whether the visibility graph of P is properly k-colourable – is NP-complete for every $k \geq 6$.*

Proof. As mentioned, the problem is in NP since one can construct the visibility graph G of P in polynomial time [8,17] and then verify a colouring. In the opposite direction, we reduce from the NP-complete Not-all-equal 3-SAT problem. Given a 3-SAT formula Φ, we efficiently construct a polygon P such that the visibility graph G of P is k-colourable if, and only if, Φ is not-all-equal satisfiable. In the proof, we refer to the previous construction outline.

We construct only the least case $k = 6$ since for higher k the construction can be easily adjusted (simply saying, we can add more shades of green to the copy gadgets), as detailed at the end. As for our terminology, a *gadget* is a consecutive part of the polygonal chain of P. The vertices of each gadget are divided to *internal* and *external* ones (except clause gadgets which have no external vertices). The internal vertices define the function of each gadget, while the external ones serve as separators from the neighbouring gadgets. Two consecutive gadgets may share their external vertices.

The unique *colour-fix gadget* A is a (convex) chain of 6 vertices $a_1, \ldots, a_6$ in this clockwise order (cf. Figure 4) which see each other. Without loss of generality, in every 6-colouring of G the colours of external vertices a_1, a_2 are *yellow and orange*, the colours of internal a_3, a_4 are *red and blue* and the colours of external a_5, a_6 are *light and dark green.*

For each variable x_i of Φ, there is one *variable gadget* $R(x_i)$ formed as a convex chain of 4 vertices $r^i_1, r^i_2, r^i_3, r^i_4$, hence seeing each other (Figure 5 top). Furthermore, the external vertices r^i_1, r^i_4 of $R(x_i)$ are visible from all four a_3, a_4, a_5, a_6 of the colour-fix gadget A, while the internal vertices r^i_2, r^i_3 of $R(x_i)$ are visible from a_5, a_6 of A. Consequently, r^i_1, r^i_4 can only be coloured yellow and orange, and r^i_2, r^i_3 can only receive colours red and blue. The internal vertices r^i_2 and r^i_3 are nicknamed x_i and $\neg x_i$, respectively, and their colours will represent their value 'true' (red) and 'false' (blue). Together, the variable gadgets $R(x_i)$, $i = 1, 2, \ldots, n$, are chained together (the order does not really matter) such that r^i_4 of $R(x_i)$ is identified with r^{i+1}_1 of $R(x_{i+1})$, and the rightmost r^n_4 is identified with a_1 of the colour-fix gadget A. Globally, the variable gadgets are arranged in a nearly-straight concave position (cf. Figure 4), so that they do not see each other (except consecutive ones at the shared external vertices). The points a_1, a_2 of A are also part of this concave arrangement.

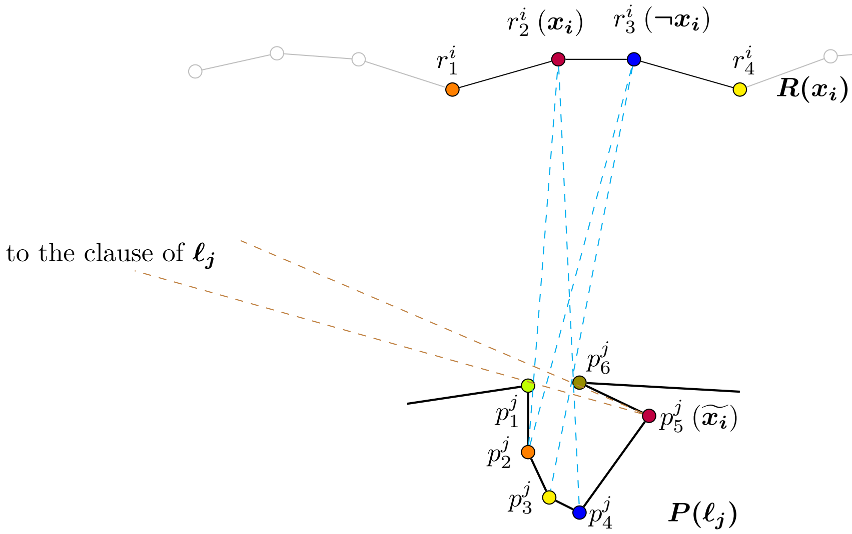

Figure 5 Detailed arrangement of the copy gadget $P(\ell_j)$ of a literal $\ell = x_i$, where the cyan dashed lines show important visibility relations between $P(\ell_j)$ and the variable gadget $R(x_i)$. Specifically, all 6 available colours have to be used on $P(\ell_j)$ and, thanks to p_4^j not seeing r_3^i, the only viable choice is to colour p_4^j same as r_3^i (blue) and then p_5^j to get the same colour as r_2^i (red). Brown dashed lines show the visibility angle of p_5^j which will be used by the corresponding clause gadget.

For each literal ℓ_j occurring in Φ (such as $\ell_j = x_i$ or $\ell_j = \neg x_i$ for some variable x_i), there is one separate *copy gadget* $P(\ell_j)$ formed as a convex chain of 6 vertices $p_1^j, \ldots, p_6^j$ (shaped as a "cavity"). Among them, p_1^j and p_6^j are the external ones, visible in particular from all four a_1, a_2, a_3, a_4 of the colour-fix gadget A. The remaining internal vertices $p_2^j, p_3^j, p_4^j, p_5^j$ are visible only from selected vertices of the variable gadget $R(x_i)$ and of the clause gadget corresponding to the literal ℓ_j. Specifically, the arrangement is as depicted in Figure 5 (for the case $\ell_j = x_i$): r_2^i sees all points of $P(\ell_j)$ except p_5^j and r_3^i sees all except p_4^j, p_5^j, while r_1^i cannot see p_2^j and r_4^i cannot see p_3^j. Besides this, r_1^i may possibly see p_4^j and r_4^i may see p_2^j, but this does not matter. The special point p_5^j (previously named $\widetilde{\ell_j}$) will see, except $P(\ell_j)$, only a later specified part of a clause gadget to which ℓ_j belongs to. The purpose of this arrangement is to force p_5^j to the same colour as r_2^i has (Claim 6), while keeping full flexibility of selecting the visibility angle of p_5^j.

If, on the other hand, the considered literal is $\ell_j = \neg x_i$, we only slightly shift the points in Figure 5, such that r_2^i could not see p_2^j and r_3^i would see p_4^j. (Alternatively, we could avoid this case by considering Φ without negations, in which case not-all-equal satisfiability remains hard.) Globally, all the copy gadgets $P(\ell_j)$ are chained together (the order does not matter) again in a nearly-straight concave shape as in Figure 4, but this time without identification of their external vertices. In particular, p_6^j is a neighbour of p_1^{j+1} on the polygonal chain of P but, importantly, p_6^j cannot see p_6^{j+1}. The points a_5, a_6 of the colour-fix gadget A are also part of this concave arrangement.

Then, for each clause $c_m = (\ell_{m_1} \vee \ell_{m_2} \vee \ell_{m_3})$ of Φ, there is one *clause gadget* $S(c_m)$ formed as a nearly-straight convex chain of 4 vertices $s_1^m, s_2^m, s_3^m, s_4^m$. All points of $S(c_m)$ are visible from a_5, a_6 of the colour-fix gadget A, and so all four remaining colours (including red and blue) have to be used on $S(c_m)$. Furthermore, the point $p_5^{m_1}$ of the copy gadget $P(\ell_{m_1})$

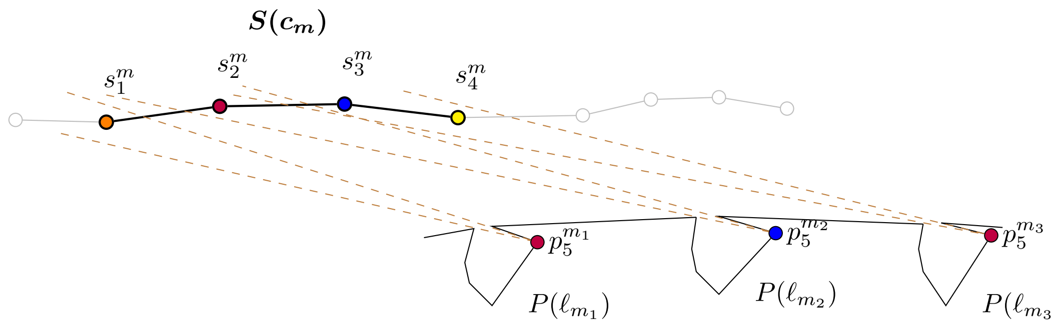

Figure 6 Detailed arrangement of the gadget $S(c_m)$ of a clause $c_m = (\ell_{m_1} \vee \ell_{m_2} \vee \ell_{m_3})$, where dashed brown lines delimit the visible angles of the points $p_5^{m_1}, p_5^{m_2}, p_5^{m_3}$. Since all clause gadgets see also colours light and dark green (e.g., of A), $S(c_m)$ can be properly coloured by itself iff not all of $p_5^{m_1}, p_5^{m_2}, p_5^{m_3}$ come with the same colour (which mimics not-all-equal satisfiability). The copy gadgets of one clause do not have to be consecutive, even though the picture shows them such.

sees exactly the point s_1^m, and likewise $p_5^{m_2}$ of $P(\ell_{m_2})$ sees exactly s_2^m. The point $p_5^{m_3}$ of $P(\ell_{m_3})$ sees both s_3^m, s_4^m. See Figure 6. The aim of this arrangement is that there would not be enough colours for whole $S(c_m)$ if all three $p_5^{m_1}, p_5^{m_2}, p_5^{m_3}$ came with the same colour. All the clause gadgets $S(c_m)$ are globally chained together, without vertex identification, in the same nearly-straight concave arrangement with the variable gadgets (cf. Figure 4). Although, no point of clause gadgets is visible from a_1, a_2, a_3, a_4 of the colour-fix gadget A.

Finally, one extra vertex (the bottom-left corner in Figure 4) is used to close the polygon P between the clause and copy sections. Validity of Theorem 5 is established from the following sequence of simple claims.

First assume that the visibility graph G of P is properly 6-coloured.

▶ **Claim 6.** *For every variable x_i of Φ, the vertices r_2^i and r_3^i of $R(x_i)$ receive colours blue and red, in either order. For every literal ℓ_j of Φ such that $\ell_j = x_i$ ($\ell_j = \neg x_i$, respectively), the vertex p_5^j of $P(\ell_j)$ receives the same colour as r_2^i (as r_3^i).*

Since (mutually visible) points r_1^i and r_4^i of $R(x_i)$ are visible from all four a_3, a_4, a_5, a_6 of the colour-fix gadget A, they must receive the colours of a_1, a_2 (yellow and orange). Then, since r_2^i and r_3^i are visible from a_5, a_6 and also from r_1^i, r_4^i, they must be coloured the same as a_3, a_4, which is red and blue.

For $\ell_j = x_i$, the points of $P(\ell_j)$ must be coloured as follows: p_1^j, p_6^j see a_1, a_2, a_3, a_4 of A, and so they have the same colours as a_5, a_6 (light and dark green). Furthermore, p_2^j, p_3^j are visible from r_2^i, r_3^i, and so they can be neither red nor blue. Consequently, p_4^j and p_5^j are red and blue (as r_2^i, r_3^i), and since r_2^i sees p_4^j, the only proper choice is to have p_5^j coloured the same as r_2^i. For $\ell_j = \neg x_i$, identical arguments lead to p_5^j being coloured the same as r_3^i. Claim 6 is finished.

▶ **Claim 7.** *Interpreting the colours of vertices p_5^j of literal ℓ_j as logical 'true' (red) and 'false' (blue), every clause of Φ receives at least one true and one false literal. Consequently, Φ is not-all-equal satisfiable.*

This claim is trivial; consider a clause $c_m = (\ell_{m_1} \vee \ell_{m_2} \vee \ell_{m_3})$. All points of $S(c_m)$ see a_5, a_6 of A, and so only the remaining four colours (yellow, orange, red, blue) are available for the four mutually visible vertices of the clause gadget $S(c_m)$. If all three points $p_5^{m_1}, p_5^{m_2}, p_5^{m_3}$ had the same colour (either red or blue), then the remaining three colours would not be enough for $S(c_m)$, and so both colours occur among $p_5^{m_1}, p_5^{m_2}, p_5^{m_3}$, as desired.

In the remaining direction of Theorem 5, we need to argue as follows.

▶ **Claim 8.** *The construction of P can be realized in a grid of polynomial size in $|\Phi|$. Consequently, the construction is a polynomial reduction.*

We refer to the sketch of P in Figure 4. Both the top and bottom concave chains can be realized as "fat" parabolas, requiring only rough resolution of $\mathcal{O}\big(|\Phi|^2\big)$. We place all the gadgets (roughly) equally spaced along, with their external vertices on these parabolas. Positioning of all the vertices of the variable and clause gadgets, and of the colour-fix gadget, is natural and easy, requiring no finer resolution. The only delicate part is to precisely place the points of the copy gadgets. The external vertices p_1^j, p_6^j get placed very close to each other on the bottom parabola, and the internal ones are then fine-positioned so that they have the required visible angles (with respect to the upper parabola). This, for each copy gadget, is done independently of all other copy gadgets, and only an additional polynomial (cubic) sub-resolution is needed for the whole copy section. This finishes Claim 8.

▶ **Claim 9.** *If Φ is not-all-equal satisfiable, then G can be properly 6-coloured.*

We describe a desired proper 6-colouring of G of P. First, we colour the gadget A and the external vertices of the variable and copy gadgets. We give vertices $a_1, a_2, a_3, a_4, a_5, a_6$ of A colours yellow, orange, red, blue, dark green, light green in this order. In every copy gadget $P(\ell_j)$, we colour p_6^j dark green and p_1^j light green (as in Figure 5). The extra vertex of P added to the left of the copy section, gets colour dark green. Thanks to concave arrangement of the copy section, this is so far a proper partial colouring of G.

For variable gadgets $R(x_i)$, we alternate colouring of the external vertices – while r_1^i may be orange and r_4^i yellow, for the next one it is $r_1^{i+1} = r_4^i$ yellow and r_4^{i+1} orange, and so on, until the last $r_4^n = a_1$ is yellow. Again, thanks to concave arrangement of the variable section, this is so far a proper partial colouring of G.

Next, we assume a not-all-equal satisfying assignment of Φ. For a variable x_i, we colour r_2^i red and r_3^i blue if x_i is 'true', and we colour r_2^i blue and r_3^i red if x_i is 'false'. Then we correspondingly colour each copy gadget featuring x_i, as in Figure 5 – this is always possible since p_2^j may inherit the colour of r_1^i and p_3^j that of r_4^i and p_4^j that of r_3^i. So, p_5^j has he same colour as r_2^i (as r_3^i, respectively, if the literal of x_i is negated).

Finally, it only remains to colour the clause section. Consider, independently of others, a clause $c_m = (\ell_{m_1} \vee \ell_{m_2} \vee \ell_{m_3})$. By the assumption of a not-all-equal satisfying assignment of Φ, the points $p_5^{m_1}, p_5^{m_2}, p_5^{m_3}$ are not all red and not all blue. Up to symmetry, $p_5^{m_3}$ is red, and so we colour s_3^m blue and s_4^m yellow. Then one of $p_5^{m_1}, p_5^{m_2}$ is not red, and so we may use remaining red and orange to colour s_1^m, s_2^m in a suitable order. Using the same rules for all clause gadgets, we do not get any "global" conflict since s_4^m would always be yellow and hence different from s_4^{m+1}, etc. Lastly, if the rightmost end (yellow) of the clause section conflicts with the leftmost end r_1^1 of the variable section, then we exchange yellow with orange in the whole clause section. This is a proper colouring of G, proving Claim 9.

The last step is to adjust the proof for $k > 6$. This is straightforward, and so we only sketch the small change: We expand the colour-fix gadget with additional $k - 6$ vertices

$a_7, \ldots, a_k$, to be coloured by more shades of the green colour. We analogously add $k-6$ new vertices to each copy gadget between p_1^j and p_2^j. All the arguments then remain the same. ◀

4 Conclusions

In this paper we have showed that the problem of deciding 6-colourability for visibility graphs of simple polygons, is NP-hard. We have also showed that the 4-colouring problem can be solved for visibility graphs of simple polygons, in polynomial time. However, the 5-colouring problem still remains open. Also, we would like to point out to the reader that the 4-colouring and 5-colouring problems on polygons with holes require further study.

References

1 Oswin Aichholzer, Greg Aloupis, Erik D. Demaine, Martin L. Demaine, Vida Dujmovic, Ferran Hurtado, and Günter Rote, André Schulz, Diane L. Souvaine, and Andrew Winslow Anna Lubiw. Convexifying polygons without losing visibilities. In *CCCG*, 2011.

2 Martin Aigner and M. Fromme. A game of cops and robbers. *Discrete Applied Mathematics*, 8(1):1–12, 1984. doi:10.1016/0166-218X(84)90073-8.

3 Matthew Babbitt, Jesse Geneson, and Tanya Khovanova. On k-visibility graphs. *Journal of Graph Algorithms and Applications*, 19(1):345–360, 2015.

4 Andreas Bärtschi, Subir Kumar Ghosh, Matúš Mihalák, Thomas Tschager, and Peter Widmayer. Improved bounds for the conflict-free chromatic art gallery problem. In *Proceedings of the Thirtieth Annual Symposium on Computational Geometry*, SOCG'14, pages 144:144–144:153, New York, NY, USA, 2014.

5 A. A. Diwan and B. Roy. On Colouring Point Visibility Graphs. *Proceedings of the 3rd Conference on Algorithms and Discrete Applied Mathematics*, pages 156–165, 2017 (Manuscript updated later - *arXiv:1610.00952*.

6 Sándor P. Fekete, Stephan Friedrichs, Michael Hemmer, Joseph B. M. Mitchell, and Christiane Schmidt. On the chromatic art gallery problem. In *Proceedings of the 26th Canadian Conference on Computational Geometry, CCCG 2014, Halifax, Nova Scotia, Canada, 2014*, 2014.

7 Subir Ghosh. *Visibility Algorithms in the Plane*. Cambridge University Press, New York, NY, USA, 2007.

8 J. Hershberger. Finding the visibility graph of a polygon in time proportional to its size. *Algorithmica*, 4:141–155, 1989.

9 Frank Hoffmann, Klaus Kriegel, Subhash Suri, Kevin Verbeek, and Max Willert. Tight Bounds for Conflict-Free Chromatic Guarding of Orthogonal Art Galleries. In Lars Arge and János Pach, editors, *31st International Symposium on Computational Geometry (SoCG 2015)*, volume 34 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 421–435, 2015.

10 J. Kára, A. Pór, and D. R. Wood. On the Chromatic Number of the Visibility Graph of a Set of Points in the Plane. *Discrete and Computational Geometry*, 34(3):497–506, 2005.

11 Jean-Claude Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Norwell, MA, USA, 1991.

12 D. T. Lee and A. K. Lin. Computational Complexity of Art Gallery Problems. *IEEE Transactions on Information Theory*, 32(2):276–282, 1986.

13 Yaw-Ling Lin and Steven Skiena. Complexity aspects of visibility graphs. *International Journal of Computational Geometry and Applications*, 5:289–312, 1995.

14 M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.

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

16 J. O'Rourke and K. Supowit. Some NP-hard polygon decomposition problems. *IEEE Transactions on Information Theory*, 29:181–190, 1983.

17 Joseph O'Rourke and Ileana Streinu. The vertex-edge visibility graph of a polygon. *Computational Geometry*, 10(2):105–120, 1998.

18 F. Pfender. Visibility Graphs of Point Sets in the Plane. *Discrete and Computational Geometry*, 39(1):455–459, 2008.

19 S. K. Ghosh and T. Shermer and B. K. Bhattacharya and P. P. Goswami. Computing the maximum clique in the visibility graph of a simple polygon. *Journal of Discrete Algorithms*, 5:524–532, 2007.

20 T. Shermer. Hiding people in polygons. *Computing*, 42:109–131, 1989.