On the Chromatic Art Gallery Problem

Sándor P. Fekete* Stephan Friedrichs* Michael Hemmer* Joseph B. M. Mitchell† Christiane Schmidt*

Abstract

For a polygonal region P with n vertices, a *guard cover* S is a set of points in P, such that any point in P can be seen from a point in S. In a *colored guard cover*, every element in a guard cover is assigned a color, such that no two guards with the same color have overlapping visibility regions. The Chromatic Art Gallery Problem (CAGP) asks for the minimum number of colors for which a colored guard cover exists.

We discuss the CAGP for the case of only two colors. We show that it is already *NP*-hard to decide whether two colors suffice for covering a polygon with holes, even when arbitrary guard positions are allowed. For simple polygons with a discrete set of possible guard locations, we give a polynomial-time algorithm for deciding whether a two-colorable guard set exists. This algorithm can be extended to optimize various additional objective functions for two-colorable guard sets, in particular minimizing the guard number, minimizing the maximum area of a visibility region, and minimizing or maximizing the overlap between visibility regions. We also show results for a larger number of colors: computing the minimum number of colors in simple polygons with arbitrary guard positions is *NP*-hard for $\Theta(n)$ colors, but allows an $O(\log(OPT))$ approximation for the number of colors.

1 Introduction

Consider a robot moving in a polygonal environment P. At any point p in P, the robot can navigate by referring to a beacon that is directly visible from p. In order to ensure unique orientation, each beacon has a "color"; the same color may be used for different beacons, if their visibility regions do not overlap. What is the minimum number $\chi_G(P)$ of colors for covering P, and where should the beacons be placed?

This is the Chromatic Art Gallery Problem (CAGP), which was first introduced by Erickson and LaValle [7]. Clearly, any feasible set of beacons for the CAGP must also be a feasible solution for the classical Art Gallery Problem (AGP). However, the number of guards and their positions for optimal AGP and CAGP solutions can be quite different, even in cases as simple as the one shown in Figure 1.

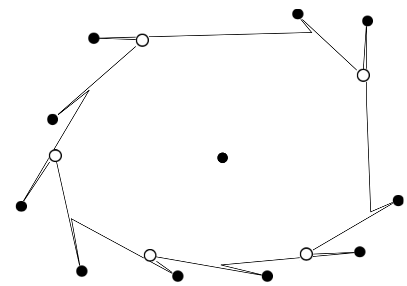


Figure 1: An example polygon with $n = 20$ vertices. A minimum-cardinality guard cover with $n/4$ guards (shown in white) requires $n/4$ colors, while a minimum-color guard cover (shown in black) has $n/2 + 1$ guards and requires only 3 colors.

Related Work. The closely related AGP is *NP*-hard [13], even for simple polygons. See [15, 16, 18] for three surveys with a wide variety of results. More recently, there has been work on developing practical optimization methods for computing optimal AGP solutions [12, 17, 10, 5].

The CAGP was first proposed by Erickson and LaValle, who presented a number of results, most notably upper and lower bounds on the number of colors for different classes of polygons [7, 8, 9]. In particular, they noted that the construction of Lee and Lin [13], which establishes *NP*-hardness of determining a minimum-cardinality guard cover for a simple polygon P, can be used to prove *NP*-hardness of computing $\chi_G(P)$, as long as all guards have to be picked from a specified candidate set [6, 9]. However, there is no straightforward way to extend this construction for showing *NP*-hardness of the CAGP with *arbitrary* guard positions. An upcoming paper by Zambon et al. [19] discusses worst-case bounds, as well as exact methods for computing optimal solutions.

Bärtschi and Suri introduced the Conflict-Free CAGP, in which they relax the chromatic requirements [2, 3]: Visibility regions of guards with the same color may overlap, as long as there is always one

*Department of Computer Science, TU Braunschweig, Germany. Email: s.fekete@tu-bs.de, stephan.friedrichs@tu-bs.de, mhsaar@gmail.com, c.schmidt@tu-bs.de.

†Department of Applied Mathematics and Statistics, Stony Brook University, USA. Email: jsbm@ams.sunysb.edu.

uniquely colored guard visible. They showed an upper bound of $O(\log^2 n)$ on the worst-case number of guards, which was lowered to $O(\log n)$ for simple polygons by Bärtschi et al. [1]. This is significantly smaller than the lower bound of $\Theta(n)$ for $\chi_G(P)$ established by Erickson and LaValle [8].

Another loosely related line of research is by Biro et al. [4], who consider beacons of a different kind: in order to get to a new location, a robot aims for a sequence of destinations for shortest geodesic paths, each from a finite set of beacons. Their results include a tight worst-case bound of $\lfloor \frac{n}{2} \rfloor - 1$ for the number of beacons and a proof of *NP*-hardness for finding a smallest set of beacons in a simple polygon.

Our Results. We provide a number of positive and negative results on the CAGP. In particular, we show the following:

- For a polygon P with holes, it is *NP*-hard to decide whether there is a k-colorable guard cover of P, even if $k = 2$ and arbitrary guard positions in P may be used. We also provide a proof for $k \geq 3$ for lack of space. Because $k = 1$ requires a single guard, i.e., P must be a star-shaped polygon, we get a complete complexity analysis.

- For a simple polygon P and a given discrete set L of c "candidate" guard locations, it can be decided in polynomial time whether there is a 2-colorable guard set $S \subset L$.

- For a simple polygon P that has a 2-colorable guard set $S \subset C$ chosen from a given discrete set L of ℓ guard positions, we can find a smallest 2-colorable guard set in polynomial time. We can also compute 2-colorable subsets that optimize any from a variety of other objective functions, including minimizing or maximizing the largest visibility region, or minimizing or maximizing the overlap between regions.

- For a simple polygon P, it is *NP*-hard to compute a guard cover with the smallest number of colors, even when arbitrary guard positions may be used. The proof establishes this for $\chi_G(P) \in \Theta(n)$.

- For a simple polygon with a given discrete set L of ℓ possible guard locations, there is a polynomial-time $O(\log(\chi_G(P))$-approximation for the number $\chi_G(P)$ of colors.

2 Preliminaries

Let P be a polygon. P is *simple* if its boundary is connected (and not self-intersecting). For $p \in P$, $\mathcal{V}(p) \subseteq P$ denotes the *visibility polygon* of p, i.e., all points p' that can be connected to p using the line segment $\overline{pp'} \subset P$. For any $S \subseteq P$, we denote by $\mathcal{V}(S) = \bigcup_{g \in S} \mathcal{V}(g)$. A finite $S \subset P$ with $\mathcal{V}(S) = P$ is called a *guard cover* of P; $g \in S$ is a *guard*. We say that g *covers* all points in $\mathcal{V}(g)$.

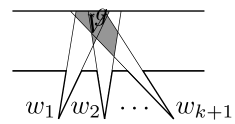


Figure 2: Forcing one guard into the region V (gray).

Let S be a guard cover of P. We call $G(S)$ the intersection graph of visbility polygons of guards $g \in S$. For a coloring $c : S \to \{1, \ldots, k\}$ of the guards, (S, c) is a k-*coloring* of P, if it induces a k-coloring of $G(S)$, i.e., no point in P sees two guards of the same color. $\chi_G(P)$ is the *chromatic (guard) number* of P, i.e., the minimal k, such that there is a k-coloring of P. (The index G in $\chi_G(P)$ as introduced in [8, 9] does not refer to a specific guard set, nor to a graph G.)

Definition 2.1 (Chromatic Art Gallery Problem) *For $k \in \mathbb{N}$, the* k-Chromatic Art Gallery Problem (k-CAGP) *is the following decision problem: Given a polygon P, decide whether $\chi_G(P) \leq k$.*

3 Polygons with Holes

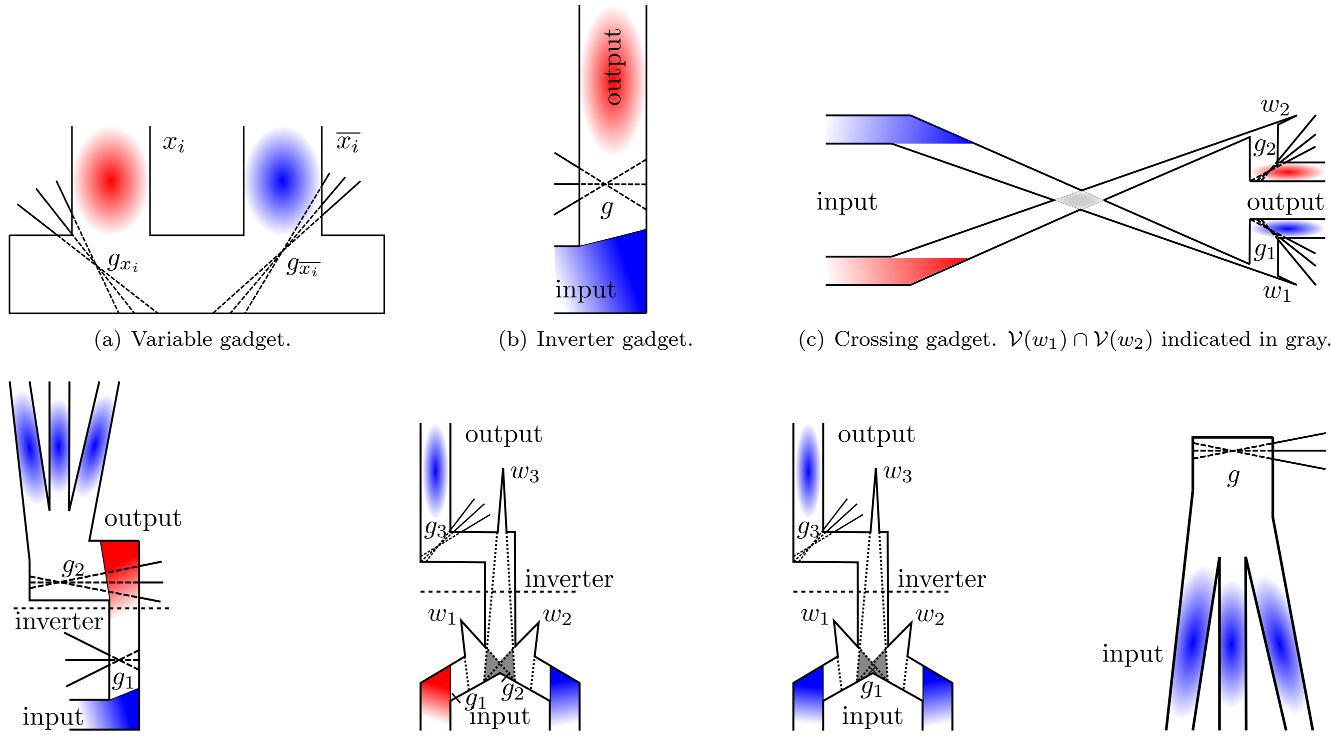
In this section, we show that the k-CAGP is NP-hard for any fixed $k \geq 3$, even when guards may be chosen arbitrarily in P.

3.1 2-Colorability

Lemma 3.1 (Needle lemma) *Consider $k+1$ "needles" with end points $W = \{w_1, \ldots, w_{k+1}\}$ (tips of the "needle" spikes), such that there is some $g \in \bigcap_{i=1}^{k+1} \mathcal{V}(w_i)$ with $\mathcal{V}(W) \subseteq \mathcal{V}(g)$, as in Figure 2. Then a k-coloring must place a guard in $V = \{p \in P \mid |\mathcal{V}(p) \cap W| \geq 2\}$, e.g. at g.*

Proof. Suppose there is no guard in V. Then there must be a guard in each $\mathcal{V}(w_i)$, requiring a total of at least $k + 1$ guards; two of them share the same color. Because $\mathcal{V}(W) \subseteq \mathcal{V}(g)$, g sees two guards with the same color, which is impossible in a k-coloring. $\square$

We use a reduction from 3SAT for showing *NP*-hardness of deciding whether $\chi_G(P) = 2$. Throughout this section, we associate colors with Boolean values; w. l. o. g., blue corresponds to *true* and red to *false*. The output of every gadget is (according to Lemma 3.1) a guard at a specific position, colored in red or blue which entirely covers all output tunnels serving as input for the next gadgets.



(a) Variable gadget. (b) Inverter gadget. (c) Crossing gadget. $\mathcal{V}(w_1) \cap \mathcal{V}(w_2)$ indicated in gray.

(d) Multiplexer gadget. (e) Or gadget, mixed input. (f) Or gadget, uniform input. (g) And gadget.

Figure 3: 3SAT to 2-CAGP reduction gadgets.

Variable Gadget. This gadget uses the construction in Figure 3(a) to encode one decision variable x_i. The needles enforce locating two guards at the indicated positions. The color used for the left guard is interpreted as the value of x_i: Blue means *true*, red means *false*.

Inverter Gadget. The gadget in Figure 3(b) inverts colors. Its input area is illuminated by one color; the guard forced to position g must have the other, or a point in the lower right corner can observe two guards of the same color.

Crossing Gadget. Crossings of channels propagating colors is achieved by the gadget in Figure 3(c). For any guard g that sees w_1, $\mathcal{V}(w_1) \subseteq \mathcal{V}(g)$. As $\mathcal{V}(w_1)$ intersects both the input area and $\mathcal{V}(g_1)$, g_1 must have the same color as the corresponding input; the same holds for w_2. If both input areas are covered by guards of the same color, then one guard of the opposite color is placed in $\mathcal{V}(w_1) \cap \mathcal{V}(w_2)$. Otherwise, we place two guards of different color outside of $\mathcal{V}(w_1) \cap \mathcal{V}(w_2)$, e.g., at w_1 and w_2.

Multiplexer Gadget. Multiplexing is achieved with the gadget in Figure 3(d). It uses an inverter gadget, and forces a guard to position g_2 which covers all output tunnels. The gadget is easily generalized to an arbitrary number of output tunnels.

Or Gadget. The gadget in Figures 3(e)–3(f) is a binary *or*, allowing construction of a ternary one. We argue that there is a guard cover that colors the output area blue, if and only if at least one input area is blue.

If two different input colors are applied, a guard cover with blue output exists: g_1 blue, g_2 red, and g_3 blue in Figure 3(e). The same is true for blue/blue input: g_1 red and g_3 blue in Figure 3(f).

If the input is red/red, the output cannot be blue: By Lemma 3.1, the gray area must contain a guard g, which can only be blue. $\mathcal{V}(g) \cap \mathcal{V}(g_3) \neq \emptyset$, so g_3 is red.

And Gadget. This gadget is similar to the multiplexer gadget, see Figure 3(g). The guard forced to position g can be colored if and only if all input regions have the same color. Note that this gadget forces all inputs to be identical, either true or false.

The 3SAT Reduction. Any 3SAT instance $S = C_1 \wedge \cdots \wedge C_m$ with variables $x_1, \ldots, x_n$ can be encoded as a 2-CAGP instance using the gadgets and the overall layout depicted in Figures 3 and 4. If S is satisfiable, setting every true variable output to blue results in a valid 2-coloring. If there is a 2-coloring of the polygon, the final and gadget's input must be uniformly colored, w. l. o. g. blue. Then the blue guards in the variable gadgets encode which variable is true and which is false in a valid truth assignment for S.

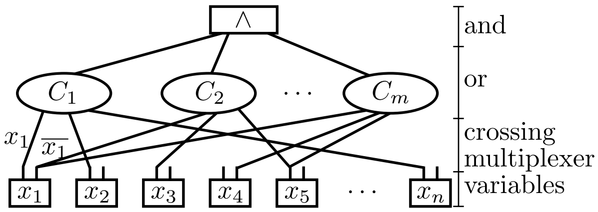


Figure 4: 3SAT reduction gadget usage.

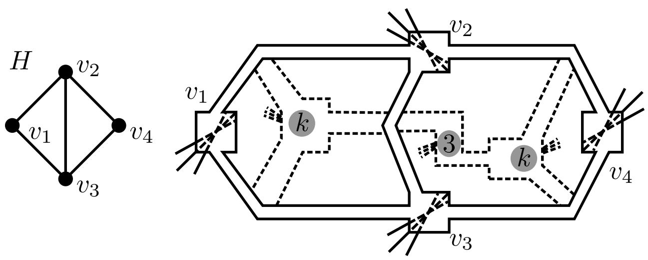


Figure 5: 3- and $(3 + k)$-colorability. 3-colorability is shown in continuous lines, $(3+k)$-colorability is achieved by adding the structures drawn in dashed lines. The gray areas contain k or 3 forced guards.

Theorem 3.2 *2-CAGP, i.e., deciding if a polygon is 2-colorable, is NP-hard.*

3.2 3- and $(3 + k)$-Colorability

The *NP*-hardness of 3-CAGP follows from the *NP*-hardness of deciding whether a planar graph H is 3-colorable-complete [11]. The idea is shown in continuous lines in Figure 5: Each node of the graph H is turned into a convex region of the polygon P, and needles force exactly one guard into each of them by Lemma 3.1. Those guards cover P; it is easy to see that P is 3-colorable iff H is 3-colorable. For $1 \leq k \in \mathbb{N}$, the construction can be generalized to $(3+k)$-CAGP by adding the structures drawn in dashed lines to Figure 5. Details are straightforward.

4 Simple Polygons

4.1 Two Colors

In the following, we consider the CAGP for a simple polygon P and a given discrete set L of ℓ candidate locations, from which a 2-colorable guard set $S \subset L$ must be chosen. Overall, we will prove the following main result.

Theorem 4.1 *For a simple polygon P with n vertices and ℓ discrete guard locations L, it can be decided in polynomial time whether there is a 2-colorable guard set.*

The proof is based on a number of structural lemmas, then proceeds by dynamic programming.

Basic Lemmas. We state a number of topological properties of 2-colorable simple polygons. Note that these hold even if we do not have a discrete candidate set for guard locations.

Lemma 4.2 *Let P be a simple polygon. A 2-colorable guard set S for P cannot contain three guards $g_1, g_2, g_3 \in S$, such that $\mathcal{V}(g_1) \cap \mathcal{V}(g_2) \cap \mathcal{V}(g_3) \neq \emptyset$.*

Proof. Trivial. □

Lemma 4.3 *Let P be a simple polygon, and let S be a 2-colorable guard set of P. Let $g_1, g_2 \in S$ be two different guards. Then any point $p \in \partial\mathcal{V}(g_1) \cap \partial\mathcal{V}(g_2)$ must be in ∂P, i.e., boundaries of visibility regions intersect only on the boundary of the polygon.*

Proof. Suppose p is in the interior of P and consider an infinitesimal neighborhood of p. This contains points that are in neither visibility region, so they must be in a third visibility region, say, of g_3. Because visibility regions are closed sets, this must also intersect p, so the claim follows from Lemma 4.2 □

Lemma 4.4 *Let P be a simple polygon, and let S be a 2-colorable guard set. Then $G(S)$ must be a tree.*

Proof. Consider a cycle in $G(S)$, which corresponds to a sequence of visibility regions. By Lemma 4.3, all of their boundary intersections must be on the boundary of P. Furthermore, these intersections must be pairwise disjoint by Lemma 4.2, so the intersection between overlapping visibility regions is non-degenerate. Thus, there is a closed path through the interior of all involved visibility regions (and thus through the interior of P) that separates two of these boundary points. Therefore, the boundary of P cannot be connected, a contradiction. □

Lemma 4.5 *Let P be a simple polygon and let S be a 2-colorable guard set. Consider a triangulation $\mathcal{T}$ of the overlay of visibility regions $\mathcal{V}(g), g \in S$. Then the dual graph of $\mathcal{T}$ is a tree.*

Proof. By Lemma 4.3, boundaries of visibility regions only intersect on the boundary of P, so all vertices of cells in the overlay lie on the boundary. It follows that any chord in a cell separates P into two pieces, proving the claim. □

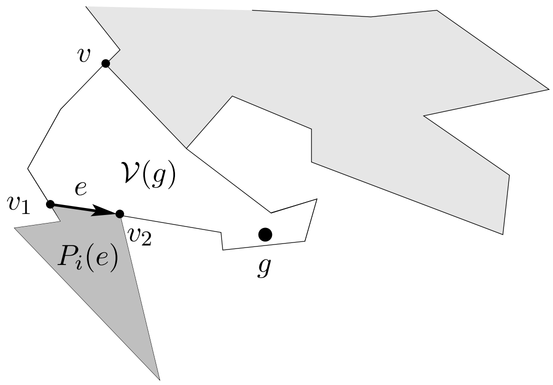


Figure 6: Illustrating Definition 4.6.

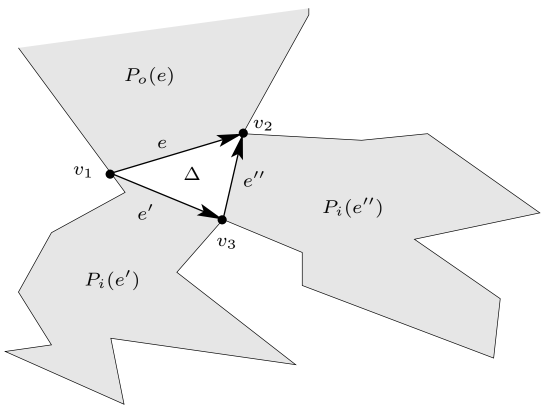


Figure 7: Solving a subproblem (e, Γ) by combining solutions for subproblems (e', Γ') and (e'', Γ'').

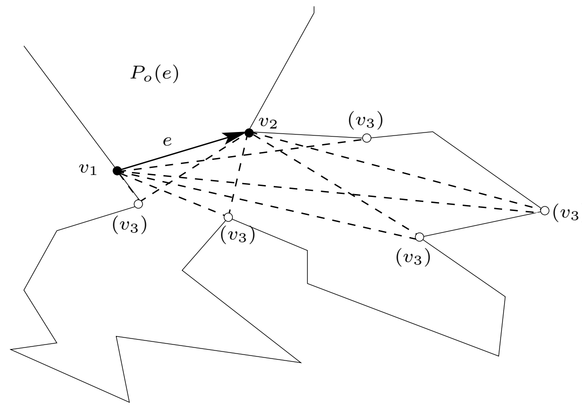


Figure 8: Different possible positions for v_3 for combining a solution for subproblem (e, Γ) from solutions for subproblems $((v_1, v_3), \Gamma')$ and $((v_3, v_2), \Gamma'')$. It suffices if *one* of these combinations yields a feasible combination for (e, Γ).

Shadow Vertices and Labeled Edges. Next we consider some basic structures for our algorithm. By Lemma 4.3, we know that the vertices of subregions in the overlay of visibility regions of active guards must lie on the boundary. This allows us to compute a 2-colorable guard set by building a triangulation of the overlay of the visibility regions of *active* guards in the polygon, one triangle at a time, based on considering the limited set of possible chords between overlay vertices. Refer to Figure 6 for the following definitions.

Definition 4.6 *Let P be a simple polygon, and let L be a discrete set of ℓ possible guard locations. Then we define the following.*

(a) For a (not necessarily active) guard position $g \in L$, a shadow vertex *v is a vertex of the boundary of $\mathcal{V}(g)$, or a polygon vertex. Any shadow vertex lies on the boundary of P. We write $\overline{V}$ for the set of all $O(n\ell)$ shadow vertices. Similarly, a* shadow edge *is a line segment between two shadow vertices that separates $\mathcal{V}(g)$ from $P \setminus \mathcal{V}(g)$ for some guard g.*

(b) A shadow chord *$e = (v_1, v_2)$ is a directed segment between two shadow vertices, v_1 and v_2. e subdivides P into two subpolygons, the* inner polygon *$P_i(e) \subset P$ that lies right of e, and the* outer polygon *$P_o(e) \subset P$ that lies left of e.*

(c) A subproblem *(e, Γ) is defined by a shadow chord together with a subset $\Gamma \subset L$ of one or two guards, g_1 and possibly g_2. It asks for a subset of guards $S \subset L$ and a triangulation of $P_i(e)$, such that S is a 2-colorable guard cover of $P_i(e)$, in which the set of guards covering the triangle to the right of e is precisely Γ. If such a set exists, we say that (e, Γ) is solvable.*

Dynamic Programming. Now the idea for our dynamic-programming algorithm is to consider building a valid triangulation of a 2-colorable visibility overlay (whose dual must be a tree by Lemma 4.5), based on shadow chords and their active guards. As each new edge in such a triangulation must be a shadow chord that forms a triangle with a pair of existing edges, we can choose such a pair that is feasible, and recurse. More formally, we consider the Boolean function $B(e, \Gamma) := 1$, if the subproblem defined by (e, Γ) is solvable.

We initialize $B(e, \Gamma) := 1$ for all combinations of e and Γ for which e is a counterclockwise polygon edge, i.e, for which $P_i(e) = e$. Then the update rule is as follows. $B(e, \Gamma) := 1$, if and only if there is a shadow vertex v_3 in $P_i(e)$ with $e' = (v_1, v_3)$ and $e'' = (v_3, v_2)$ with guard sets Γ' and Γ'', such that the subproblems (e', Γ') and (e'', Γ'') are both solvable in a way that stays valid when we cover triangle $\Delta(v_1, v_3, v_2)$ by guard set Γ. (It is straightforward to see the necessary relationship between Γ, Γ' and Γ'': when crossing a shadow edge, the covering guard set must change by the corresponding guard; when crossing any other shadow chord, it must stay the same.)

The recursion terminates with a feasible result, if and only if there is a clockwise polygon edge e with $B(e,\Gamma) = 1$, i.e, a subproblem (e,Γ) for which $P_i(e) = P$, i.e., $B(e_\Gamma) := 1$. Conversely, it is straightforward to see that any feasible result implies that a feasible solution will be found.

Additional Objective Functions. It is not hard to see that the dynamic-programming approach can be extended to computing not just any 2-colorable guard set, but also one that is optimal with respect to an objective functions. In the following, we state this for some of the more interesting ones; of course others exist.

Corollary 4.7 *For a simple polygon P with n vertices and a set L of ℓ candidate locations for guards, we can compute in polyomial time a 2-colorable guard set that is optimal with respect to one of the following objectives.*

- *Minimize/maximize the number of guards.*
- *Minimize/maximize the largest/smallest visibility region.*
- *Minimize/maximize the average area of visibility regions.*
- *Minimize/maximize the largest/smallest overlap between visibility regions.*
- *Minimize/maximize the total overlap between visibility regions.*

This can be done by replacing the Boolean function B and its update rule by an appropriate objective function; for average values, subproblems can be extended appropriately.

4.2 Many Colors

***NP*-hardness.** Our proof of NP-hardness of the general CAGP for simple polygons is based on a reduction from computing a minimum-cardinality set of points for covering a given set of lines; this auxiliary problem is easily seen to be NP-hard by geometric duality applied to a result of Megiddo and Tamir [14], who showed that it is NP-hard to determine the minimum number of lines to cover a set of points in the plane.

Theorem 4.8 *It is NP-hard to determine the chromatic number of a simple polygon, even for arbitrary guard positions.*

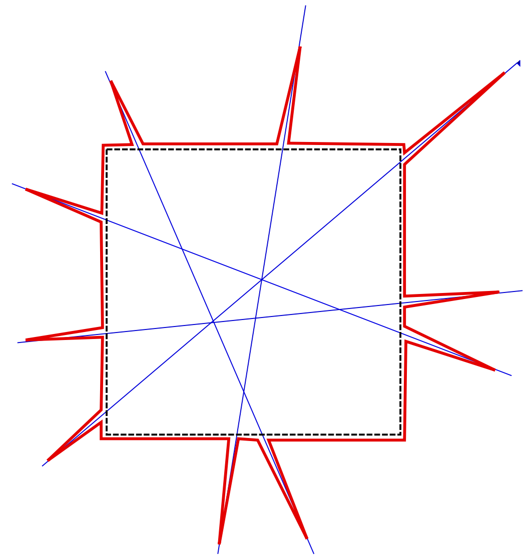


Figure 9: NP-hardness of the general CAGP for simple polygons: A minimum-color guard cover of the spike box (red) corresponds to a minimum-cardinality point cover of a set of lines (blue).

Proof. Refer to Figure 9. For a given set of lines, construct a "spike box", which is formed by a square that contains all intersection points of the lines, and has two narrow niche extensions for each line, one at either intersection with the square. Note that the visibility regions of any two points in the spike box overlap; thus, minimizing the number of colors is equivalent to minimizing the number of guards. Now any guard cover corresponds to a point cover of the lines; conversely, any line cover can be converted to a guard cover of the spike box: if there is a guard placed in a niche, replace it by one inside the square. $\square$

Approximation. On the positive side, we can establish an $O(\log(\chi_G(P)))$-approximation algorithm for the CAGP in simple polygons with a discrete set of candidate locations.

Theorem 4.9 *For a simple polygon P with a set L of ℓ candidate guard locations, we can find an $O(\chi_G(P)\log(\chi_G(P)))$-colorable guard set in polynomial time.*

The algorithm uses greedy set cover. In each iteration, we use dynamic programming to find an independet set of guard positions that covers a maximum number of uncovered cells in the overlay of all visibility polygons $\mathcal{V}(g)$ for $g \in L$.

5 Conclusion

A number of open problems remain. These include the complexity of the CAGP for simple polygons with $\chi_G(P) = 2$ without fixed a discrete set of guard locations as well as the complexity for $\chi_G(P) = 3$ with a discrete candidate set. Note that the latter remains open for any fixed $\chi_G(P) = k$, as the NP-hardness proof by Erickson and LaValle [9] requires large $\chi_G(P)$. Among the other open questions for fixed guard locations is the complexity of determining the chromatic number of a given guard set in a simple polygon; the claim by Erickson and LaValle stated in [9] that this problem has a polynomial solution is still unproven, as the corresponding conflict graph does *not* have to be chordal: this can be seen from the $n/2$ black locations at the spikes in Figure 1.

Acknowledgements

We thank Cid de Souza, Pedro de Rezende, Maurício Zambon and Bruno Espinosa Crepaldi for helpful and pleasant discussions. J. Mitchell is partially supported by NSF (CCF-1018388) and the US-Israel Binational Science Foundation (project 2010074).

References

[1] A. Bärtschi, S. K. Ghosh, M. Mihalák, T. Tschager, and P. Widmayer. Improved bounds for the conflict-free chromatic art gallery problem. In *Proceedings of the 30th Symposium on Computational Geometry (SoCG)*, pages 144–153, 2014.

[2] A. Bärtschi and S. Suri. Conflict-free chromatic art gallery coverage. In *Proceedings of the 29th International Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 14 of *LIPIcs*, pages 160–171, 2012.

[3] A. Bärtschi and S. Suri. Conflict-free chromatic art gallery coverage. *Algorithmica*, 68(1):265–283, 2014.

[4] M. Biro, J. Iwerks, I. Kostitsyna, and J. S. B. Mitchell. Beacon-based algorithms for geometric routing. In *Proceedings of the 13th International Symposium on Algorithms and Data Structures (WADS)*, volume 8037 of *LNCS*, pages 158–169. Springer, 2013.

[5] D. Borrmann, P. J. de Rezende, C. C. de Souza, S. P. Fekete, A. Kröller, A. Nüchter, and C. Schmidt. Point guards and point clouds: Solving general art gallery problems. In *Proceedings of the 29th Symposium on Computational Geometry (SoCG)*, pages 347–348. ACM, 2013.

[6] L. Erickson. *Visibility analysis of landmark-based navigation.* PhD thesis, University of Illinois at Urbana-Champaign, 2014.

[7] L. H. Erickson and S. M. LaValle. A chromatic art gallery problem. Technical report, University of Illinois at Urbana-Champaign, 2010.

[8] L. H. Erickson and S. M. LaValle. An art gallery approach to ensuring that landmarks are distinguishable. In *Robotics: Science and Systems*, 2011.

[9] L. H. Erickson and S. M. LaValle. How many landmark colors are needed to avoid confusion in a polygon? In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2302–2307. IEEE, 2011.

[10] S. P. Fekete, S. Friedrichs, A. Kröller, and C. Schmidt. Facets for art gallery problems. In *Proceedings of the 19th International Conference on Computing and Combinatorics (COCOON)*, volume 7936 of *LNCS*, pages 208–220. Springer, 2013.

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

[12] A. Kröller, T. Baumgartner, S. P. Fekete, and C. Schmidt. Exact solutions and bounds for general art gallery problems. *Journal of Experimental Algorithms*, 17(1), 2012.

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

[14] N. Megiddo and A. Tamir. On the complexity of locating linear facilities in the plane. *Operations Research Letters*, 1, 1982.

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

[16] T. C. Shermer. Recent results in art galleries. In *Proceedings of the IEEE*, volume 80, pages 1384–1399, 1992.

[17] D. Tozoni, P. J. de Rezende, and C. C. de Souza. The quest for optimal solutions for the art gallery problem: A practical iterative algorithm. In *Proceedings of the 13th Symposium on Experimental Algorithms (SEA)*, volume 7933 of *LNCS*, pages 320–336. Springer, 2013.

[18] J. Urrutia. Art gallery and illumination problems. In J.-R. Sack and J. Urrutia, editors, *Handbook on Computational Geometry*, pages 973–1026. Elsevier, 2000.

[19] M. J. O. Zambon, P. J. de Rezende, and C. C. de Souza. An exact algorithm for the chromatic art gallery problem. In *Proceedings of the 14th Symposium on Experimental Algorithms (SEA)*, 2014. (To appear).