Closed Quasigeodesics, Escaping from Polygons, and Conflict-Free Graph Coloring

by

Adam Classen Hesterberg

A.B., Princeton University (2011)

Submitted to the Department of Mathematics
in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2018

Signature of Author: ______________________________

Department of Mathematics
May 11, 2018

Certified by: ______________________________

Erik Demaine
Professor of Computer Science and Engineering
Thesis Supervisor

Accepted by: ______________________________

Jonathan Kelner
Mark Hyman, Jr. Career Development Associate Professor of Applied Mathematics
Chairman, Department Committee on Graduate Theses

Closed Quasigeodesics, Escaping from Polygons, and Conflict-Free Graph Coloring

by

Adam Classen Hesterberg

Submitted to the Department of Mathematics
on May 11, 2018 in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy in
Mathematics

ABSTRACT

Closed quasigeodesics

A *closed quasigeodesic* on the surface of a polyhedron is a loop which can everywhere locally be unfolded to a straight line: thus, it's straight on faces, uniquely determined on edges, and has as much flexibility at a vertex as that vertex's curvature. On any polyhedron, at least three closed quasigeodesics are known to exist, by a nonconstructive topological proof. We present an algorithm to find one on any convex polyhedron in time $O(n^2\varepsilon^{-2}L\ell^{-1})$, where ε is the minimum curvature of a vertex, L is the length of the longest side, and ℓ is the smallest distance within a face between a vertex and an edge not containing it.

Escaping from polygons

You move continuously at speed 1 in the interior of a polygon P, trying to reach the boundary. A zombie moves continuously at speed r outside P, trying to be at the boundary when you reach it. For what r can you escape and for what r can the zombie catch you? We give exact results for some P. For general P, we give a simple approximation to within a factor of roughly 9.2504. We also give a pseudopolynomial-time approximation scheme. Finally, we prove NP-hardness and hardness of approximation results for related problems with multiple zombies and/or humans.

Conflict-free graph coloring

A *conflict-free k-coloring* of a graph assigns one of k different colors to some of the vertices such that, for every vertex v, there is a color that is assigned to exactly one vertex among v and v's neighbors. We study the natural problem of the *conflict-free chromatic number* $\chi_{CF}(G)$ (the smallest k for which conflict-free k-colorings exist), with a focus on planar graphs.

Thesis Supervisor: Erik Demaine
Title: Professor of Computer Science and Engineering

Thanks to my advisor, Erik Demaine.

For Canada/USA Mathcamp and my parents, Bev and Tim Hesterberg.

Contents

Chapter 0

Overview

This thesis has three technical chapters.

Chapter 1 is about closed quasigeodesics on polyhedra, and is joint work with Erik Demaine and Jason Ku, with help from discussions with Zachary Abel, Nadia Benbernou, Fae Charlton, Jayson Lynch, Joseph O'Rourke, Diane Souvaine, and David Stalfa.

Chapter 2 is about a game of escaping from polygons, and is joint work with Zachary Abel, Erik Demaine, Martin Demaine, Jason Ku, and Jayson Lynch, with help from discussions with Greg Aloupis and Fae Charlton.

Chapter 3 is about "conflict-free" graph coloring and is joint work with Zachary Abel, Victor Alvarez, Erik Demaine, Sándor Fekete, Aman Gour, Phillip Keldenich, and Christian Scheffer, with help from discussions with Bruno Crepaldi, Pedro de Rezende, Cid de Souza, Stephan Friedrichs, Michael Hemmer, and Frank Quedenfeld. It has appeared in the proceedings of the ACM-SIAM Symposium on Discrete Algorithms [AAG+18].

Chapter 1

Closed Quasigeodesics

This chapter is joint work with Erik Demaine and Jason Ku, with help from discussions with Zachary Abel, Nadia Benbernou, Fae Charlton, Jayson Lynch, Joseph O'Rourke, Diane Souvaine, and David Stalfa.

1.1 Introduction

A *geodesic* on a surface is a path which is, local to every point on it, a shortest path; a *closed geodesic* on a surface is a loop with the same property. Poincaré conjectured in 1905 [Poi05], and Pogorelov [Pog73] and Ballmann [Bal78] independently proved, building on work of Lyusternik and Schnirelmann [LS29], that every smooth surface of genus 0 has at least three non-self-intersecting closed geodesics.

For non-smooth surfaces (say, polyhedra), an analog of a geodesic is a *quasigeodesic*, a path which can locally be unfolded to a straight line. That is, on a face, a quasigeodesic is a straight line; at an edge, it's a straight line after the faces meeting at that edges are unfolded to be flat at that edge; and at a vertex of curvature κ (that is, one at which the sum of the angles is $2\pi - \kappa$), a quasigeodesic entering the vertex at a given angle can exit it anywhere in an angular interval of length κ, as in Figure 1.1. Analogously, a *closed quasigeodesic* is a loop which is quasigeodesic, and a *(quasi)geodesic ray/segment* is a one/two-ended path which is (quasi)geodesic. The same proof of Ballmann's [Bal78] also shows that there are at least three non-self-intersecting closed quasigeodesics on every polyhedron, by approximating it with smooth surfaces.

The proof of existence of those closed quasigeodesics is nonconstructive, and [DO07, Open Problem] asks, in 2007, for a polynomial (or any) algorithm to find one. We provide, in Section 1.2, an algorithm which finds at least one closed quasigeodesic on a convex polyhedron in time $O(n^2\varepsilon^{-2}L\ell^{-1})$, where n is the number of vertices of the polyhedron, ε is the smallest curvature at a vertex, L is the length of the longest side, and ℓ is the smallest distance within a face between a vertex and an edge not containing it. This running time is pseudopolynomial, since L, ℓ^{-1}, and ε^{-1} may be exponential in the length of a binary description of the polyhedron, so this does not resolve the question of a polynomial-time algorithm. Also, a closed quasigeodesic found by our algorithm may be self-intersecting, even though a non-self-intersecting one is guaranteed to exist. In Section 1.3 we discuss some of the difficulties involved in resolving either of these issues.

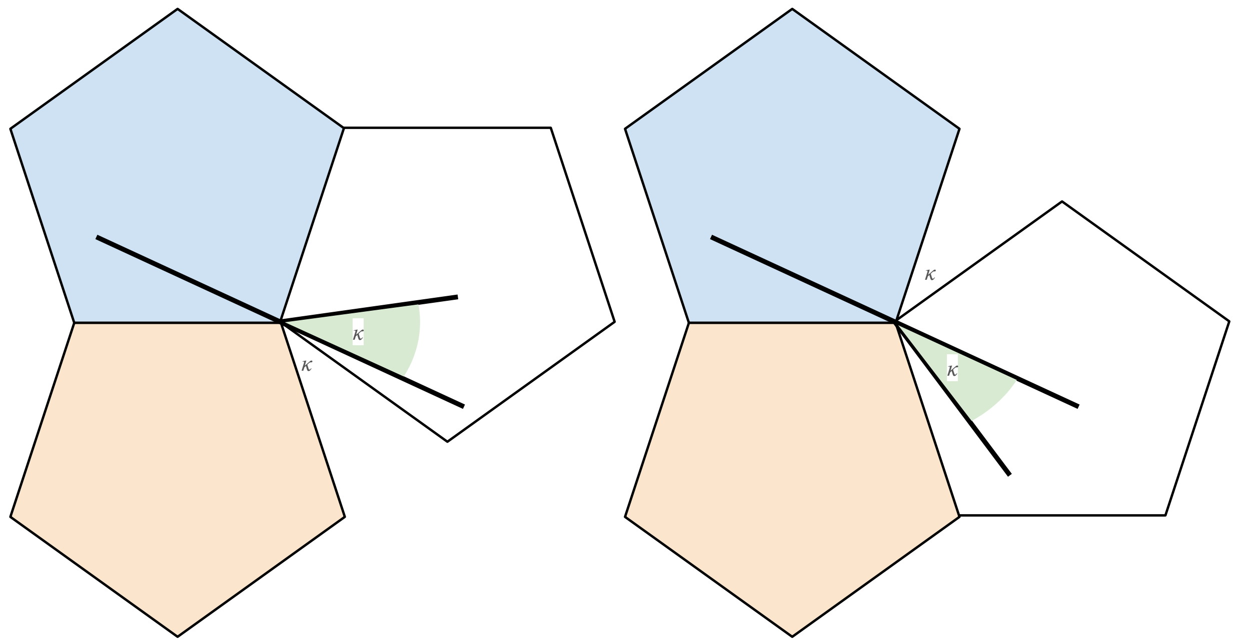


Figure 1.1: At a vertex of curvature κ, there's an angular interval of size κ in which a segment of a quasigeodesic can be extended: the segment of geodesic starting on the left can continue straight in either of the pictured unfoldings or any of the intermediate unfoldings in which the right pentagon touches only at a vertex.

1.2 Algorithm

In this section, we give an algorithm to find a closed quasigeodesic on the surface of a convex polyhedron P.

1.2.1 Outline

The idea of the algorithm is roughly as follows: first, we define a directed graph for which each node[1] is a pair $(V, [\varphi_1, \varphi_2])$ of a vertex V of P and a small interval of directions at it, with an edge from one such pair to another if a geodesic starting at the former vertex and somewhere in the former range of directions can reach the latter vertex and continue everywhere in the latter range of directions. We show how to calculate at least one out-edge from every node of that graph, so we can start anywhere and follow edges until hitting a node twice, giving a closed quasigeodesic.

The key part of this algorithm is to calculate, given a vertex U and a range of directions, another vertex V that can be reached starting from that vertex and in that range of directions, even though reaching V may require crossing superpolynomially many faces. First we prove some lemmas toward that goal.

Definition 1.2.1. If X is a point on the surface of a polyhedron, φ is a direction at X, and $d > 0$, then $R(X, \varphi, d)$ is the geodesic segment starting at X in the direction φ and continuing for a distance d or until it hits a vertex, whichever comes first. We allow $d = \infty$; in that case, $R(X, \varphi, d)$ is a geodesic ray.

[1]We call vertices of the graph "nodes" to distinguish them from vertices of the polyhedron.

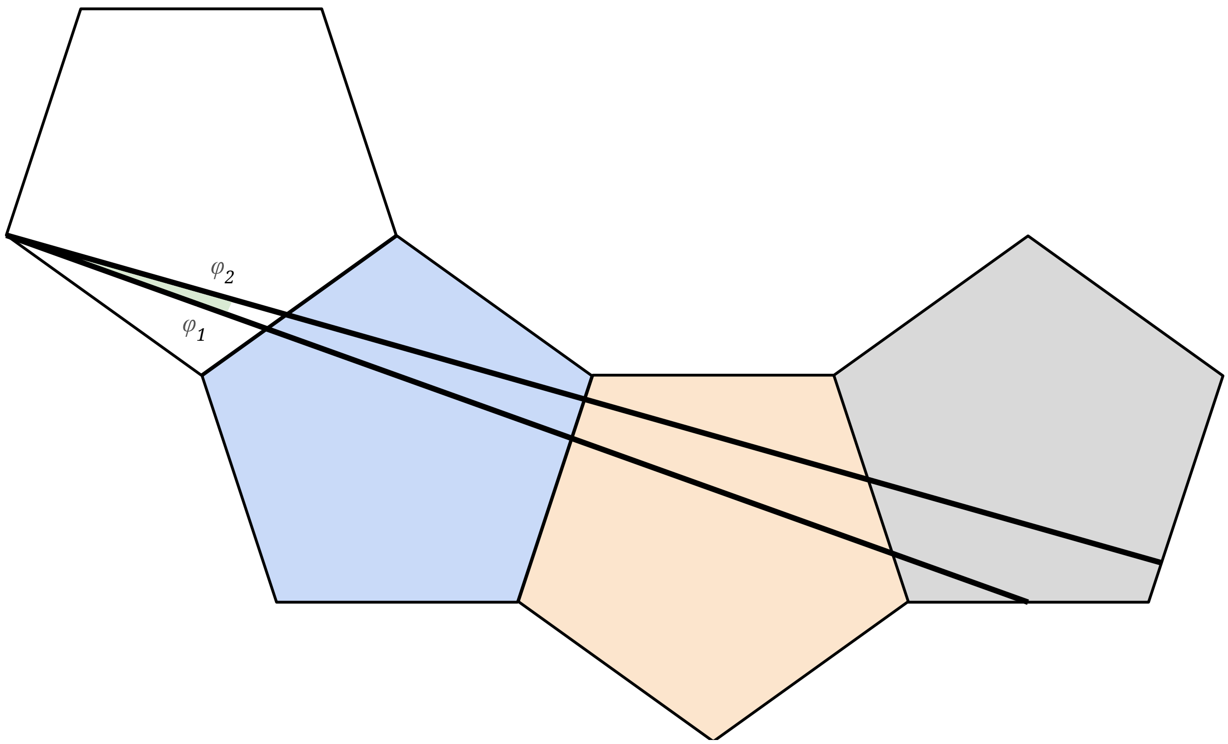


Figure 1.2: A segment of a geodesic is a straight line in the unfolding of the sequence of faces through which it passes, as in this unfolding of a regular dodecahedron.

Definition 1.2.2. If $R(X, \varphi, d)$ is a geodesic segment or ray, the *face sequence* $F(R(X, \varphi, d))$ is the (possibly infinite) sequence of faces that $R(X, \varphi, d)$ visits.

Lemma 1.2.1. *If $R_1 = R(X, \varphi_1, \infty)$ and $R_2 = R(X, \varphi_2, \infty)$ are two geodesic rays from a common starting point X with an angle between them of $\theta \in (0, \pi)$, the face sequences $F(R_1)$ and $F(R_2)$ are distinct, and the first difference between them occurs at most one face after a geodesic distance of $O(\theta^{-1}L)$.*

Proof. Given a (prefix of) $F(R_i)$, the segment of R_i on it is a straight line, so while $F(R_1) = F(R_2)$, the two geodesics R_1 and R_2 form a wedge in a common unfolding, as in Figure 1.2. The distance between the points on the rays at distance d from X is $2d \sin \frac{\theta}{2} > d\theta/\pi$ (since $\frac{\theta}{2} < \frac{\pi}{2}$), so at a distance of $O(\theta^{-1}L)$, that distance is at least L. So either $F(R_1)$ and $F(R_2)$ differ before then, or the next edge that R_1 and R_2 cross can't be the same edge, in which case $F(R_1)$ and $F(R_2)$ differ in the next face, as claimed. □

If we had defined L analogously to ℓ as not just the length of the longest side but the greatest distance within a face between a vertex and an edge not containing it, we could remove the "at most one face after" condition from Lemma 1.2.1.

1.2.2 Extending Quasigeodesic Rays

Although Lemma 1.2.1 gives a bound on the geodesic distance to the first difference in the face sequences (or one face before it), it's not a bound on the number of faces traversed before that difference, which might be large if the two paths come very close to a vertex of high curvature, as in Figure 1.3, or repeat the same sequence of edges many times, as in Figure 1.4.

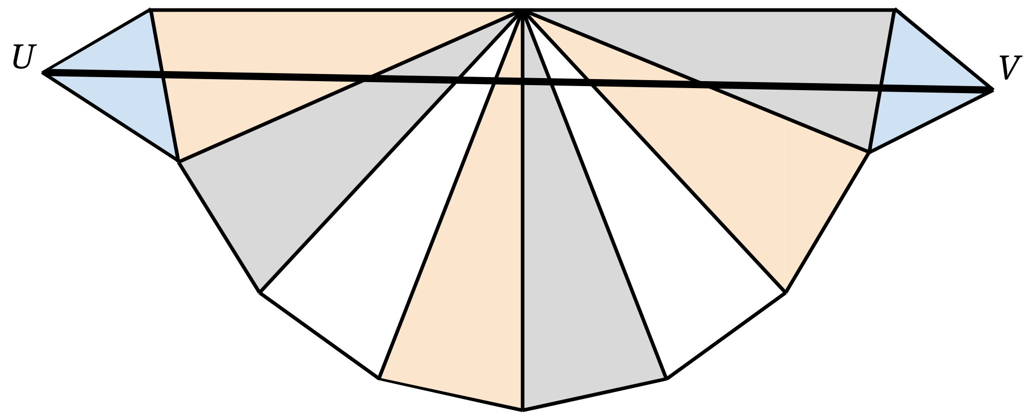


Figure 1.3: Even a short geodesic path between two vertices u and v may cross many edges.

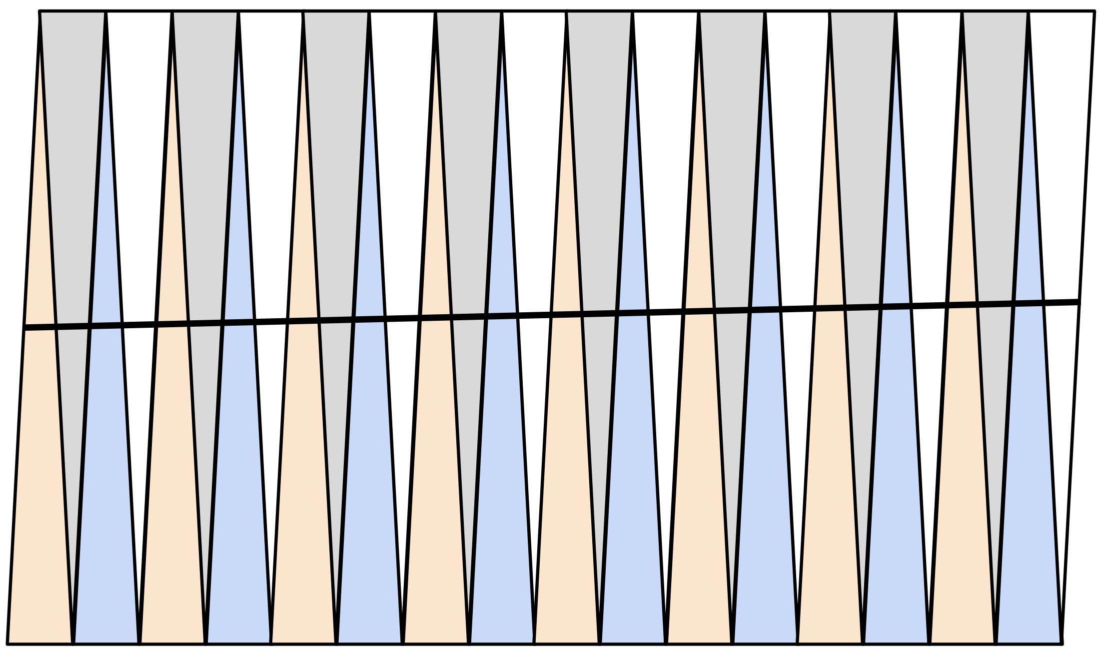


Figure 1.4: If a geodesic path encounters the same edge twice in nearly the same place and nearly the same direction, as is the case for the thick quasigeodesic path through the center of this figure if every fourth triangle is the same face, it may pass the same sequence of faces in the same order a superpolynomial number of times.

Nonetheless, in both of these cases, we can describe a geodesic ray's path efficiently:

Lemma 1.2.2. *Let $R = (X, \varphi, d)$ be a geodesic segment with $d < \ell$. In $O(n)$ time, we can calculate $F(R)$, expressed as a sequence S_1 of $O(n)$ faces, followed by another sequence S_2 of $O(n)$ faces and a distance over which R visits the faces of S_2 periodically*[2]*. Also, we can calculate the face, location in the face, and direction of R at its endpoint other than X.*

Proof. If R enters a face f on an edge e_1 and exits at a point P_2 on an edge e_2, then we claim that *every* time R enters f by e_1, it must exit f by e_2. It can't exit by the same edge e_1 by which it

[2]The length of the sequence of faces may be too large to even write down the number of repetitions.

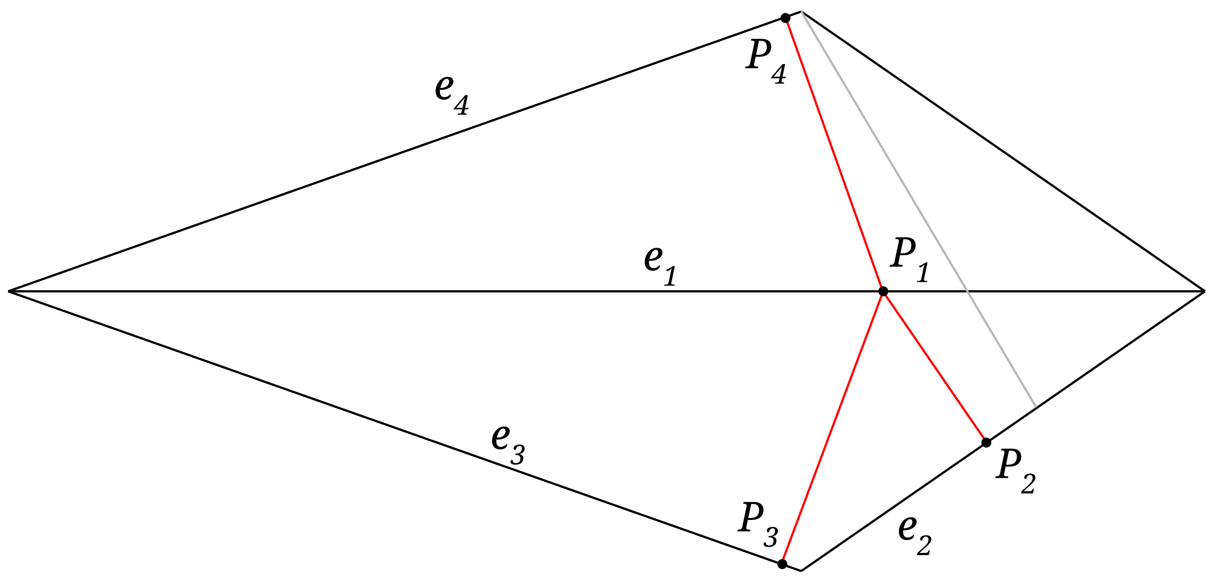


Figure 1.5: If a geodesic visits three edges of the same face, the total distance traveled is at least ℓ.

entered, so suppose for contradiction that in some visit to f, it enters at a point P_1 on the edge e_1 and exits at a point P_3 on another edge e_3, as shown in Figure 1.5. If any two of e_1, e_2, and e_3 are nonincident, then R has gone from a point on one edge to a point on a nonincident edge. By the definition of ℓ, that's a distance of at least ℓ. Otherwise, e_1, e_2, and e_3 are the three edges of a triangular face, and the total geodesic distance is at least $d(P_1, P_2) + d(P_1, P_3)$. Consider the reflection e_4 of e_3 across e_2 and the reflected point P_4 on e_4. The path from P_4 to P_2 via P_1 is at least the distance from P_2 to P_4, which is at least the shortest distance from a point on e_4 to a point on e_2, which is attained at an endpoint of at least one of e_2 and e_4, say an endpoint of e_4. The path making that shortest distance (shown in gray) goes through e_1, so it's at least the distance from e_1 to the opposite vertex, which is at least ℓ, farther than the conditions under which this lemma applies. Hence each edge crossed determines the next edge crossed, so $F(R)$ is periodic after crossing each edge at most once. Also, there are only $O(n)$ edges, so after crossing at most $O(n)$ edges, $F(R)$ repeats periodically with period $O(n)$.

In total time $O(n)$, we can calculate the path of R before it repeats periodically for each face f it enters, as follows:

1. For each edge e of f, we can calculate in how much distance R would cross e, in $O(1)$ time.

2. The edge on which R exits f is the one minimizing that distance. We can, in $O(1)$ time, calculate where on that edge and at what angle R crosses it.

There are $O(n)$ pairs of a face and an edge of that face, so the total amount of computation before the face sequence repeats periodically is $O(n)$. (If R ends at a vertex before then, we calculate so because R exits a face by two edges at the same time.)

Consider the shape formed by the faces of $F(R)$ that repeat periodically, as in the bolded part of Figure 1.6. Copies of this shape attach to each other on copies of a repeated edge e; that is, the entire shape is translated and possibly rotated to identify the copies of e. If there's no rotation, as in Figure 1.6, all copies of each edge e are translates of each other by a constant amount, and

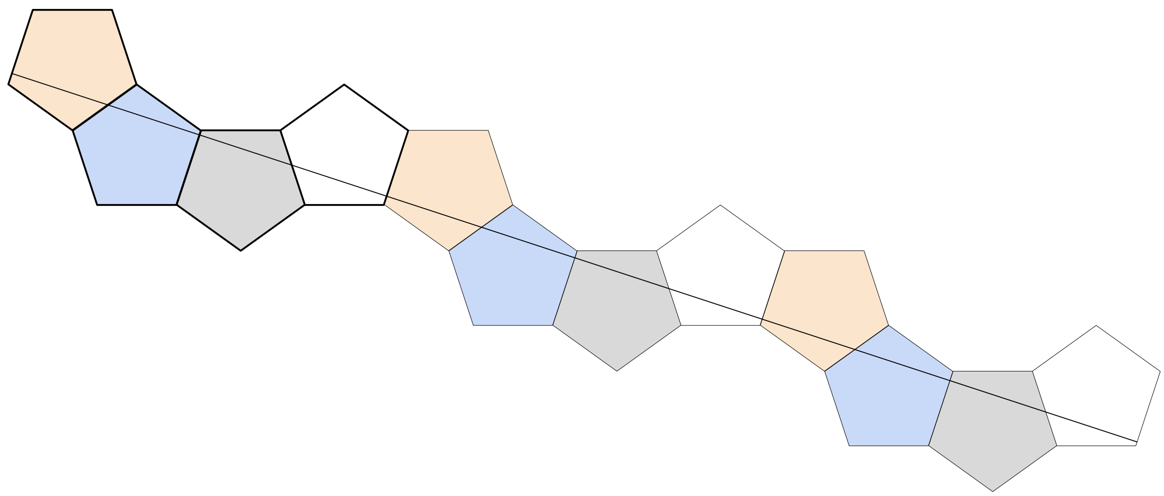


Figure 1.6: When a quasigeodesic path passes through the same sequence of faces several times, the unfolding of the faces it passes through repeats regularly.

we can calculate in $O(1)$ time where in the translated figure the other endpoint of R is and in $O(n)$ time which face that corresponds to and where. If there is rotation, all copies of each edge e are rotations around a consistent center point C (in the plane of the unfolding). Again, we can determine the path in time $O(n)$ by calculating the last time it hits a rotation of each edge e; for each such calculation, we only need to check where the line intersects the circle along which each endpoint of e rotates.

□

Corollary 1.2.3. *In $O(nd\ell^{-1})$ time, we can calculate $R(X, \varphi, d)$.*

Proof. Apply Lemma 1.2.2 to $R = R(X, \varphi, \frac{\ell}{2})$, which gives us the point X' and direction φ' of the endpoint of R other than X. Apply Lemma 1.2.2 to $R(X', \varphi', \frac{\ell}{2})$, and repeat $2d\ell^{-1}$ times. □

1.2.3 Full Algorithm

We are now ready to state the algorithm for finding a closed quasigeodesic in $O(n^2\varepsilon^{-2}L\ell^{-1})$ time:

Theorem 1.2.4. *Let P be a convex polyhedron with n vertices all of curvature at least ε, let L be the length of the longest side, and let ℓ be the least distance between points on edges sharing a face but not a vertex. Then in $O(n^2\varepsilon^{-2}L\ell^{-1})$ time, we can find a closed quasigeodesic on P. We can express such a closed quasigeodesic as a sequence of $O(n^3\varepsilon^{-2}L\ell^{-1})$ subsequences of faces, where for each subsequence we give a distance for which the closed quasigeodesic visits that subsequence of faces periodically.*

Proof. For each vertex V of P, divide the total angle at that vertex (that is, the angles at that vertex in the faces that meet at that vertex) into arcs of size between $\varepsilon/4$ and $\varepsilon/2 < \pi$, making $O(\varepsilon^{-1})$ such arcs at each vertex.

Construct a directed graph G whose nodes are pairs of a vertex of V and one of those arcs, giving the graph $O(n\varepsilon^{-1})$ nodes, with an edge from a node[3] u to a node v if there exists a direction at u that hits v's vertex and can continue from every angle in v's arc.

[3]We use capital letters and the word "vertex" for vertices of a polyhedron and lower-case letters and the word "node" for vertices of a graph.

Let v be a node of G, with corresponding vertex V and angles from φ_1 to φ_2. By Corollary 1.2.3 we can, in $O(n\varepsilon^{-1}L\ell^{-1})$ time, follow each of the rays $R_1 = R(V, \varphi_1)$ and $R_2 = R(V, \varphi_2)$ for a distance of $\varepsilon^{-1}L$ and compare their face sequences $F(R_1)$ and $F(R_2)$. By Lemma 1.2.1, either $F(R_1)$ and $F(R_2)$ differ or we can reach a difference by extending each of R_1 and R_2 to the end of its current face (which we can calculate in $O(n)$ more time). The first difference in the face sequences $F(R_1)$ and $F(R_2)$ determines a vertex reachable in the wedge between R_1 and R_2. That is, given a vertex V and a range of angles at it from φ_1 to φ_2, we can, in $O(n\ell^{-1}L\varepsilon^{-1})$ time, determine a vertex reachable from V via an angle between φ_1 and φ_2. Once we reach such a vertex, a quasigeodesic can exit the vertex anywhere in an angle equal to that vertex's curvature, which is at least ε, so for at least one of the arcs of size at most $\varepsilon/2$ at that vertex, the quasigeodesic can exit anywhere in that arc.

Hence in time $O(n\ell^{-1}L\varepsilon^{-1})$ we can find and follow an out-edge from any node of G. After at most as many such transitions as the number of nodes of G, $O(n\varepsilon^{-1})$, we find a cycle, which is exactly a closed quasigeodesic.

Also, that quasigeodesic is composed of $O(n\varepsilon^{-1})$ edges of the graph. Each of those edges is a geodesic distance of $O(L\varepsilon^{-1})$ plus at most one face. Over the distance of $O(L\varepsilon^{-1})$, each segment of length $\frac{\ell}{2}$ is described by Lemma 1.2.2 as a subsequence of $O(n)$ faces, possibly visited periodically over some geodesic distance. So, each of the edges is described as a sequence of $O(n\varepsilon^{-2}L\ell^{-1})$ faces, with subsequences possibly visited periodically, and the whole geodesic is described as a sequence of $O(n^2\varepsilon^{-3}L\ell^{-1})$ faces, with subsequences possibly visited periodically over specified distances, as desired. $\square$

If D is the greatest diameter of a face, then a closed quasigeodesic found by Theorem 1.2.4 has length $O(n\varepsilon^{-1}(\varepsilon^{-1}L + D))$, because the quasigeodesic visits $O(n\varepsilon^{-1})$ graph nodes, and, by Lemma 1.2.1, goes a distance at most $\varepsilon^{-1}L + D$ between each consecutive pair.

1.3 Conclusion

It has been known for four decades (as in [Bal78]) that every convex polyhedron has a closed quasigeodesic; we give the first algorithm to find one. This algorithm is polynomial in not just the number of vertices of the input polyhedron, but instead also depends on some features of that polyhedron, leaving some questions open.

Question 1. *Theorem 1.2.4 does not necessarily find a* non-self-intersecting *closed quasigeodesic, even though at least three are guaranteed to exist. Is there an algorithm to find one? In particular, can we find the* shortest *closed quasigeodesic?*

Any approach similar to Theorem 1.2.4 is unlikely to resolve this, for several reasons:

1. Parts of a quasigeodesic could enter a vertex at infinitely many angles. Theorem 1.2.4 makes this manageable by grouping similar angles of entry to a vertex, but if similar angles of entry to a vertex are combined, extensions that would be valid for some of them but invalid for others are treated as invalid for all of them. For instance, a quasigeodesic found by Theorem 1.2.4 will almost never turn by the maximum allowed at any vertex, since exiting a vertex at the maximum possible turn from one entry angle to the vertex may mean exiting it with more of a turn than allowed for another very close entry angle. So there are some closed quasigeodesics that Theorem 1.2.4 can't find, and those may include non-self-intersecting ones.

2. Given a vertex and a wedge determined by a range of directions from it, we can find *one* vertex in the wedge, but if we wish to find more than one, the problem becomes more complicated.

When we seek only one vertex, there's only one unfolding of the faces to consider, which the entire wedge stays in until it hits a vertex; when we pass a vertex, the unfoldings on each side of it might be different, so we multiply the size of the problem by 2 every time we pass a vertex. There may, in fact, be exponentially many non-self-intersecting geodesic paths between two vertices: for instance, O'Rourke [O'R18] gives the example of a doubly-covered regular polygon, in which a geodesic path may visit every vertex in order around the cycle but may skip vertices.

Question 2. *Theorem 1.2.4 assumes that arithmetic operations with real numbers can be done in $O(1)$ time, even when the input is given with finitely many bits (say, integer coordinates for the vertices). It may, however, be the case that every vertex unfolds to a point with algebraic coordinates; if so, is there an analog of Theorem 1.2.4 using only arithmetic operations on rational numbers?*

Question 3. *Theorem 1.2.4 is polynomial in not just n but the smallest curvature at a vertex, the length of the longest side, and the shortest distance within a face between a vertex and an edge not containing it. Are all of those necessary? Can the last be simplified to the length of the shortest side?*

Question 4. *Can the algorithm of Theorem 1.2.4 be extended to nonconvex polyhedra P?*

Question 5. *Is there an algorithm to find a closed quasigeodesic passing through a number of faces bounded by a polynomial function of n, ε, L, ℓ, and perhaps the minimum total angle of a polyhedron vertex? Does Theorem 1.2.4 already have such a bound?*

A single quasigeodesic ray may pass through a number of faces not bounded by a function of those parameters before ceasing to cycle periodically: for instance, the geodesic ray of Figure 1.4 does. However, we have no example for which a whole geodesic wedge passes through a number of faces not bounded by a function of those parameters before containing a vertex.

Chapter 2

Escaping from Polygons

This chapter is joint work with Zachary Abel, Erik Demaine, Martin Demaine, Jason Ku, and Jayson Lynch, with help from discussions with Greg Aloupis and Fae Charlton.

2.1 Introduction

In 1961, Richard Guy [Guy61] posed the following classic puzzle, reproduced in [O'B61]:

> Some robbers have stolen the green eye of a little yellow god from a temple on a small island in the middle of a circular lake. As they embark in their boat, they are observed by a solitary guard on the shore, who can run four times as fast as they can row the boat. Can they be sure of reaching the shore and escaping with their loot? If so, how? And what if the guard could move four *and a half* times as fast as the robbers?

The same problem was rethemed by Martin Gardner [Gar65] to be about a maiden on a rowboat. In this chapter, we retheme again and ask about shapes other than a circle:

Problem 1. A human chooses a position in a *human play area*, a subset of a metric space. Then a number n_z of zombies, who can each run r times as fast[1] as the human, choose positions in a *zombie play area*, another subset of the same metric space. The humans and zombies move simultaneously and continuously, staying in their own play areas, with every player having full knowledge of every other player's movement plans. The human wins if they can reach a point of the zombie play area with no zombie at the same point; if the zombies can prevent that for arbitrarily long, then the zombies win. Given such a setup, what is the *critical speed ratio* $r^* \geq 0$ such that the human wins if the zombies are less than r^* times faster and the zombies win if they're more than r^* times faster?

We give names to some common types of human play area and zombie play area:

1. In the "moat model"[2], the human play area is the interior and boundary of a (possibly unbounded) polygon P, and the zombie play area is the boundary of P.
2. In the "standard model", the human play area is a (possibly unbounded) polygon P with its boundary, and the zombie play area is the exterior and boundary of P.

[1] For simplicity, we make the human's speed always 1, and use "speed ratio" and "zombie's speed" interchangeably for r.

[2] So named as if the zombie is trapped in a moat.

3. In the “Jordan model” (which can be applied with either the moat model or the standard model), the human play area is a Jordan region in the plane instead of a polygon. For instance, Guy’s problem is in the Jordan model where the region is a disk.

4. In the “graph model”, the human play area is the edges and vertices of a graph, and the zombie play area is the edges and vertices of another graph (possibly overlapping the human’s).

In this chapter, we investigate the following cases of this problem. Unless specified otherwise, all results in the chapter are for the standard model and the moat model, with $n_z = 1$ zombie.

1. In Section 2.2, we calculate the critical speed ratio in two cases simple enough to calculate it exactly: Guy’s problem (from [O’B61]) and an infinite wedge (in any of the models), both with $n_z = 1$.

2. In Section 2.3, we give bounds on the critical speed ratio r^* that differ by a factor of approximately 9.2504, in the moat model and standard model with $n_z = 1$.

3. Also in Section 2.3, we give a pseudopolynomial-time approximation scheme for the critical speed ratio r^*, in the moat model and the standard model with $n_z = 1$.

4. In Section 2.4, we consider $n_z > 1$, and give miscellaneous results in all the models.

5. In Section 2.5, we prove NP-hardness and hardness of approximation results in the graph model with arbitrary n_z.

2.2 Exact Answers

First we investigate two shapes for which the critical speed ratio can be calculated exactly: a circle and an unbounded intersection of halfplanes.

2.2.1 Circle

Theorem 2.2.1. *Let $\varphi \approx 0.43\pi$ be the angle such that $\tan\varphi = \pi + \varphi$. Then the critical speed ratio r^* for a circle is $\sec\varphi \approx 4.60$.*

This result comes from [O’B61], as does the proof that $r^* \geq \sec\varphi$; we reproduce that proof, flesh out some details, and also prove that $r^* \leq \sec\varphi$:

Proof. First we reproduce the proof from [O’B61] that $r^* \geq \sec\varphi$. That is, we’ll prove that, if $r < \sec\varphi$, then the human can escape. While the human is within distance $d \leq \cos\varphi$ of the center of the circle, its maximum angular speed around the center is greater than the zombie’s, so the human can reach a point opposite the zombie at a distance of $\cos\varphi$ from the center.

Let the human be at a position H at distance $d \geq \cos\varphi$ from the center of the circle, opposite the zombie’s position Z, as in Figure 2.1. We claim that the human can either reach such a position with greater d or escape. First, let the human move straight away from the center of the circle until the zombie is no longer on the same diameter; without loss of generality, let the zombie move counterclockwise. Then let the human pick a point T on the boundary of the circle such that $HT \perp ZT$ and the zombie is moving on the major arc from Z to T, and run straight toward T until they either reach T or the zombie is again diametrically opposite them. Note that the angle at the center of the circle between T and H is $\arccos d$; if the human is at a distance of exactly $\cos\varphi$ from the center of the circle, then the angle at the center of the circle between T and H is

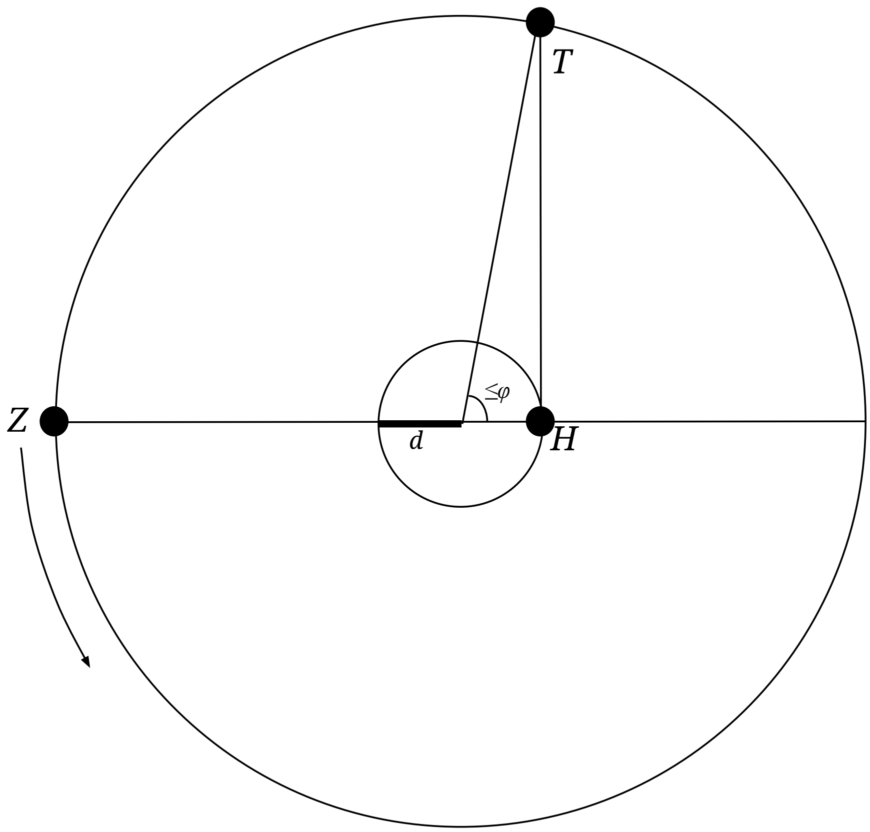


Figure 2.1: Human and zombie strategies at one position in the game on a circle.

φ, and otherwise $\arccos d < \varphi$. If the zombie ever crosses a point antipodal to the human again, then the human has reached the same position with greater d; otherwise, the zombie must travel a distance of at least $\pi + \arccos d$ to reach T, which takes time at least

$$\frac{\pi + \arccos d}{r} > (\pi + \arccos d)\cos\varphi,$$

and the human can get there in time $\sin \arccos d$, and it suffices to show that

$$(\pi + \arccos d)\cos\varphi - \sin\arccos d \geq 0.$$

But if $x \leq \varphi$, then

$$\frac{d\left((\pi + x)\cos\varphi\right)}{dx} = \cos\varphi \leq \cos x = \frac{d\sin x}{dx},$$

so for $d \geq \cos\varphi$, that is, for $\arccos d \leq \varphi$, that expression is minimized at $d = \cos\varphi$, and there it's $(\pi + \varphi)\cos\varphi - \sin\varphi = 0$, as desired. So, the human reaches T first and escapes.

Conversely, if the zombie's speed r is more than $\sec\varphi$ greater than the human's, we claim that the human cannot escape. The zombie's strategy is simple:

1. While the human is within $\frac{1}{r}$ of the center of the circle, stand still. Imagine eating the human's brain, to work up an appetite.

2. While the human is more than $\frac{1}{r}$ from the center of the circle, move *mindlessly* along the shorter arc toward the closest point on the boundary of the circle to the human (breaking ties arbitrarily).

Suppose for contradiction that the human can escape. If the human starts within $\frac{1}{r}$ of the center of the circle, let H (without loss of generality, on the positive x axis) be the last point at distance at most $\frac{1}{r}$ of the center of the circle that the human passed through; otherwise, let the human start on the positive x axis, let $H = (\frac{1}{r}, 0)$, and let the zombie start at $(1, 0)$. Let $T = (\cos\varphi, \sin\varphi)$ be the point at which the human eventually escapes. The human can't get to T faster than by the straight line from H to T, at a distance of $\sqrt{(\cos\varphi - \frac{1}{r})^2 + \sin^2\varphi}$. If the human is outside the circle of radius $\frac{1}{r}$, then the zombie's angular velocity around the center of the circle is greater than the human's, so the arclength between the zombie and the closest point of the human to the boundary of the circle only decreases; that is, the zombie can choose to run in a consistent direction. So the zombie reaches T in time at most $\frac{\pi+\varphi}{r}$ (less if the human started outside the circle of radius $\frac{1}{r}$). But by calculations similar to the above, the zombie gets there first:

$$\begin{aligned}
&\sqrt{\sin^2\varphi + (\cos\varphi - \frac{1}{r})^2} \\
=&\sqrt{1 - 2r^{-1}\cos\varphi + r^{-2}} \\
=&\sqrt{1 - 2\cos\varphi\cos\varphi + \cos^2\varphi + (r^{-1} - \cos\varphi)(r^{-1} + \cos\varphi - 2\cos\varphi)} \\
\geq&\sqrt{1 - \cos^2\varphi} \\
=&\sin\varphi \\
=&\tan\varphi\cos\varphi \\
=&(\pi + \varphi)\cos\varphi \\
\geq&(\pi + \varphi)/r,
\end{aligned}$$

so the zombie catches the human, finishing the proof and the human's brain. □

2.2.2 Wedge

If the human play area is a wedge (an unbounded intersection of halfplanes, whose boundary is simply connected), we can calculate the critical speed ratio exactly:

Theorem 2.2.2. *If P is an unbounded intersection of halfplanes and the angle between the two extreme halfplanes is $2\theta \in (0, \pi]$, then the critical speed ratio r^* is $\csc\theta$.*

Proof. If P is just a halfplane, a zombie of speed $1 = \csc\frac{\pi}{2}$ can win by staying at the projection of the human onto the boundary, since that projection moves at most as fast as the human. A zombie of speed less than 1 loses to a human who moves along the boundary. Otherwise, the two extreme halfplanes are distinct.

Orient the wedge so that the boundaries of the two extreme halfplanes are at angles of $\pm\theta$ from the positive x axis, and their intersection (which may not be in the wedge, if the wedge is bounded by more than two halfplanes) is the origin, as in Figure 2.2.

If the zombie's speed r is at least $\csc\theta$ times the human's, then the zombie can stay at the same y coordinate as the human while staying on the boundary: the human's speed in the y coordinate is at most 1, and the zombie's speed in the y coordinate while staying on part of the boundary with slope $\varphi \in [-\theta, \theta]$ is $\frac{r}{\csc\varphi} \geq \frac{\csc\theta}{\csc\theta} = 1$. A point on the boundary is uniquely determined by its y coordinate, so if the human reaches a point on the boundary, then the zombie is there too to catch it.

If the zombie's speed r is less than $\csc\theta$ times the human's, then the human can go to a point $(\frac{1}{\varepsilon}, 0)$ (that is, on the angular bisector of the two halfplanes at a very large distance from each

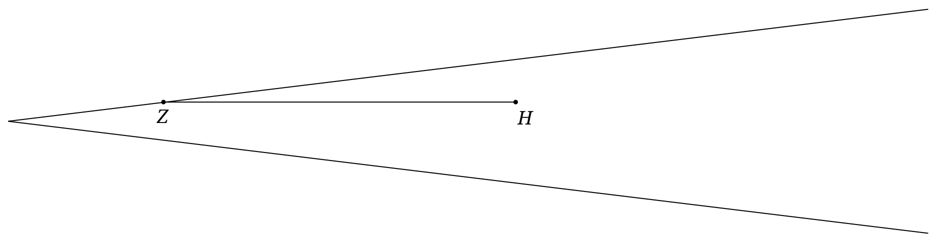


Figure 2.2: Zombie strategy if the play area is a wedge.

of them), for some ε to be chosen later. Without loss of generality, the zombie has a nonpositive y coordinate; then the human moves straight up toward the point $T = (\frac{1}{\varepsilon}, \frac{\tan\theta}{\varepsilon})$, reaching it in time $\frac{\tan\theta}{\varepsilon}$. If the intersection of the boundary with the positive x axis is at $(x_0, 0)$, the zombie's path to the human's escape point T must take it through the x axis at a point no closer to T than $(x_0, 0)$, from which the distance to the human's escape point is $\sqrt{(\frac{1}{\varepsilon} - x_0)^2 + (\frac{\tan\theta}{\varepsilon}))^2}$. For sufficiently small ε, that's close to $\frac{\sec\theta}{\varepsilon}$, so the zombie needs time close to $\frac{\sec\theta}{r\varepsilon}$ to reach T. For $r < \csc\theta$, the human gets there first and escapes, as claimed. $\square$

2.3 Approximation

In the previous section, we found the exact critical speed ratio for specific human play areas P by methods that don't generalize to arbitrary polygons or Jordan regions. We don't have an algorithm to compute the exact critical speed ratio for arbitrary P, but we can approximate it, as the following two results show. All theorems in this section are valid for the standard model and moat model, and have $n_z = 1$ zombies.

2.3.1 $O(1)$-approximation

Theorem 2.3.1. *Let P be any polygon. Then the critical speed ratio r^* is at least $\max_{p,q\in\delta P} \frac{d_z(p,q)}{d_h(p,q)}$ (where d_z and d_h are the geodesic distances in the zombie and human play areas, respectively).*

Proof. Let p and q be points maximizing the expression above. The human can first go to p; if the zombie doesn't go to p as well, the human escapes at p. If the zombie does come to p, the human can run toward q. The human's distance to p is $d_h(p,q)$ and the zombie's is $d_z(p,q)$, so if the zombie's speed is less than $\frac{d_z(p,q)}{d_h(p,q)}$, then the human can reach q first and escape. $\square$

Theorem 2.3.2. *Let P be any polygon. Then the critical speed ratio r^* is at most $9.2504 \max_{p,q\in\delta P} \frac{d_z(p,q)}{d_h(p,q)}$ (where d_z and d_h are the geodesic distances in the zombie and human play areas, respectively).*

Proof. Divide the polygon into regions by its medial axis; that is, each region is associated with an edge of the polygon and is the set of points inside the polygon closest to that edge of the polygon, as shown in Figure 2.3. Also, for each region, define the *fringe* of that region to be the union, over points p inside the region, of the circle centered at p with radius $x \cdot d(p, \delta P)$, where $d(p, \delta P)$ is the distance from p to the nearest point on the boundary of P and $x \approx 0.465$ is a fringe size parameter.

Let the zombie's strategy be as follows:

1. At all times, the zombie has a target edge e such that it attempts to be at the closest point on e to the human. Initially, this edge is the one closest to the human.

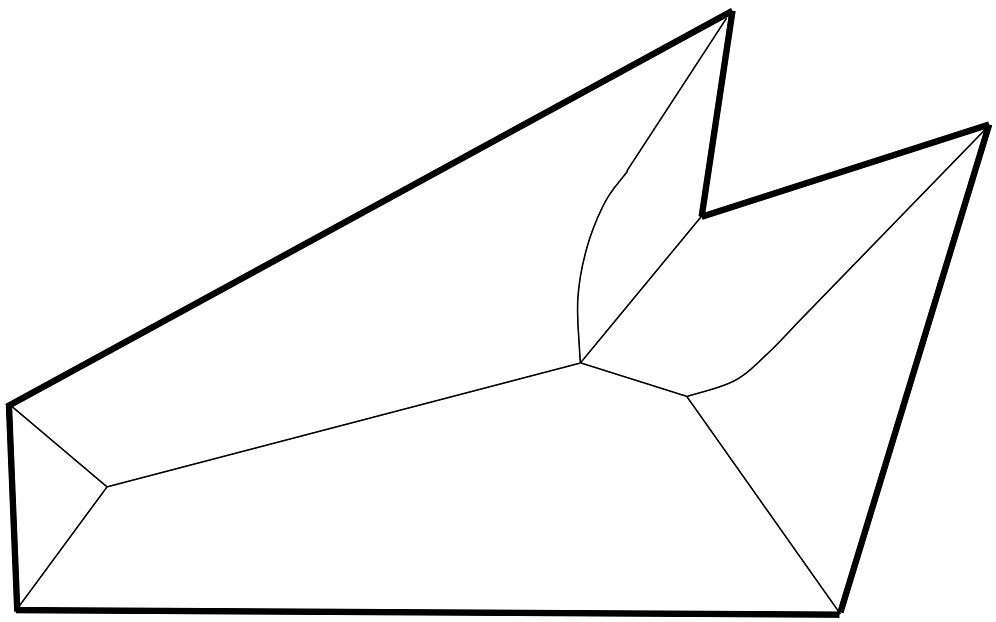


Figure 2.3: A polygon and its medial axis.

2. When the human exits the fringe of the medial axis region corresponding to e, the zombie runs to the closest point on the boundary to the human. If that point is on edge f, the zombie switches its target edge to f.

This strategy works as long as, when the human leaves the fringe of the medial axis region for the zombie's target edge e, the zombie can run into position for the medial axis region R for its new target edge f before the human leaves the fringe of R (triggering another strategy change) or reaches the boundary and escapes.

First, we define some points, as in Figure 2.4. Let h be the point at which the human leaves the fringe (drawn in blue) of a region R (drawn in red) with corresponding edge e. Then h is in the fringe of R because it's in a circle centered at a point o in R; if p is the closest point to o on e, then $d(o,h) = x \cdot d(o,p)$. Also, let z be the closest point on R's edge to h, which is where the zombie stands when the human exits the fringe at h, let q be the closest point to o on δP, and let f be q's edge. Also, let θ be the angle between (the extensions of) e and f, so $\pi - \theta$ is the angle at o between oq and op.

When the human leaves at h, their distance to the boundary is $d(h,q) = d(o,q) - d(o,h) = d(o,q)(1-x) = d(o,p)(1-x)$. So, to leave the fringe of their new region, the human must go a distance of at least $d(o,p)x(1-x)$. Before they do, the zombie must be in position for the new strategy, which requires moving at most:

1. $d(z,p)$ to return to p. Since z is the closest point on its edge to h, it's at least as close to p as the projection of h onto e (possibly closer, if e doesn't extend that far). The length of that projection is $d(o,h)\sin\theta = d(o,p)x\sin\theta$, so that's an upper bound on the zombie's distance to return to p.

2. $d_z(p,q)$ to reach q.

3. $d(o,p)x(1-x)$ to match the human's move (projected onto f).

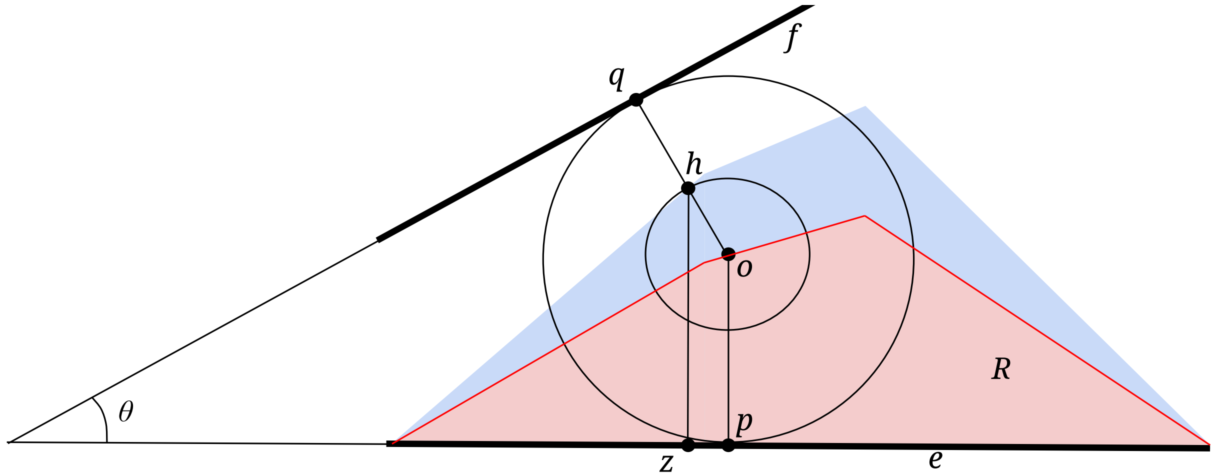


Figure 2.4: A section of a polygon with a region defined by the medial axis and its fringe region.

So, if the zombie's speed is enough to travel those three distances in the time the human travels a distance of $d(o,p)x(1-x)$, the zombie can be in position in time for the human's next region change. That is, the critical speed ratio r^* is at most

$$\frac{d(o,p)x\sin\theta + d_z(p,q) + d(o,p)x(1-x)}{d(o,p)x(1-x)} = 1 + \frac{\sin\theta}{1-x} + \frac{d_z(p,q)}{d(o,p)x(1-x)}.$$

Also, since a closest point to o on δP is p, the circle centered at o with radius $d(o,p)$ is contained in P, so $d_z(p,q) \geq (\pi-\theta)d(o,p)$, the distance from p to q along the circle centered at o. Also, that circle (and hence P) contains the line segment from p to q, so $d_h(p,q) \leq 2d(o,p)\cos\frac{\theta}{2}$. So $\frac{d_z(p,q)}{d_h(p,q)} \geq \frac{\pi-\theta}{2\cos\frac{\theta}{2}}$, so the critical speed ratio is at most

$$\max_{p,q\in\delta P}\frac{d_z(p,q)}{d_h(p,q)}\max_\theta\left(\left(1+\frac{\sin\theta}{1-x}\right)\left(\frac{2\cos\frac{\theta}{2}}{\pi-\theta}\right)+\frac{2\cos\frac{\theta}{2}}{x(1-x)}\right).$$

Having chosen the fringe size parameter $x \approx 0.465$, that expression is maximized at roughly $\theta = 0.24\pi$ with the value $9.2504\max_{p,q\in\delta P}\frac{d_z(p,q)}{d_h(p,q)}$, so the zombie can win if it's faster than that, as claimed. ☐

2.3.2 Pseudopolynomial-Time Approximation Scheme

Although Theorem 2.3.2 describes the critical speed ratio in terms of the polygon, it's not an algorithm (since finding the pair (p,q) of points maximizing $\frac{d_z(p,q)}{d_h(p,q)}$ may take some work) nor can it approximate arbitrarily closely. To remedy those flaws, we also give a pseudopolynomial-time approximation scheme for the critical speed ratio r^* for any polygon P; that is, given P and $\varepsilon > 0$, we describe a scheme for approximating r^* to within a factor of $1+\varepsilon$ in time polynomial in ε^{-1} and the coordinates of P. (Since the side lengths of P can be exponential in the length of their encoding, the approximation scheme is only pseudopolynomial.)

First, we define a discrete analogue of the game.

Definition 2.3.1. Let P be a closed subset of the plane whose boundary is a union of line segments, and let $\varepsilon > 0$. The *ε-discretization* of P is the graph whose vertices are the following points p in the plane:

1. If p is on an edge of P, the distance to one endpoint of that edge is a multiple of ε^2.
2. If p is in the interior of P, x and y are both multiples of ε^2.

There is an edge between two vertices if and only if the distance between the corresponding points is at most ε.

Definition 2.3.2. Let P be a closed subset of the plane whose boundary is a union of line segments, let $\varepsilon > 0$, and let z and h be positive integers. The (P, ε, z, h) *discrete game* is as follows:

1. First, the human chooses a vertex, their *start location*, of the ε-discretization of P.
2. Second, if $hull(P)$ and $int(P)$ are the convex hull and interior of P, respectively, the zombie chooses a vertex, their *start location*, of a graph: the ε-discretization of $hull(P) \setminus int(P)$ if the game is in the standard model, or the ε-discretization of δP if the game is in the moat model.
3. The human and zombie alternate turns, starting with the human.
4. In the human's turn, the human moves to a vertex at distance at most h in the graph from their current vertex.
5. In the zombie's turn, the zombie moves to a vertex at distance at most z in the graph from their current vertex.
6. If, at the end of the zombie's turn, the human is at a vertex on the edge of P, and the zombie is not at the same vertex, the human wins.

There is no loss condition for the human, but we say the human loses if they can never win.

Theorem 2.3.3. *For every polygon P there exists an $\varepsilon_0 > 0$, such that ε_0^{-1} is polynomial in the coordinates of P and if r^* is the critical speed ratio for P, then for all $\varepsilon \in (0, \varepsilon_0)$ and for all integers $z, h \in (0, \varepsilon^{-1})$, the human wins the (P, ε^5, z, h) game if $z/h \in [1, r^* \frac{1}{(1+\varepsilon)^3})$ and the zombie wins if $z/h > r^*(1+\varepsilon)^3$.*

In particular, we will prove Theorem 2.3.3 for any ε_0 such that:

1. There's a point in P at distance more than ε_0 from the nearest boundary. We can calculate a lower bound on this by triangulating P, choosing any of that triangulation's triangles, and using the inradius of that triangle. The inradius is the area divided by half the perimeter, and both of those are polynomial functions of the input coordinates, so this bound on ε_0 is polynomial in the coordinates of P.
2. No disk of radius $2\sqrt{\varepsilon_0}$ contains two edges not sharing a vertex. We can calculate a lower bound on this: the minimum distance between two edges not sharing a vertex is attained either by a pair of vertices (and we can calculate the minimum distance between pairs of vertices) or by the perpendicular from a vertex v to an edge (u, w). The length of that perpendicular is the area of the triangle with vertices u, v, and w divided by the distance from u to w, and those are both polynomial in u, v, and w, so this bound on ε_0 has length (in bits) polynomial in the length (in bits) of P.

3. $\varepsilon_0 < 1/(2r(P)^2)$ if $r(P)$ is the critical speed ratio for P. We can calculate a bound on this depending only on P by Theorem 2.3.2 as follows: the critical speed ratio is between $\max_{p,q\in\delta P} \frac{d_z(p,q)}{d_h(p,q)}$ and $9.2504 \max_{p,q\in\delta P} \frac{d_z(p,q)}{d_h(p,q)}$. If p and q are on the same edge, then $\max_{p,q\in\delta P} \frac{d_z(p,q)}{d_h(p,q)}$ is the cosecant of half the angle between them, as in Theorem 2.2.2; if not, then $d_h(p,q)$ is at least the minimum distance between two points on edges not sharing a vertex, which is polynomial as above, and $d_z(p,q)$ is at most the perimeter of P, which is polynomial, giving an upper bound on $\max_{p,q\in\delta P} \frac{d_z(p,q)}{d_h(p,q)}$.

Before we prove Theorem 2.3.3, we note that this implies the existence of a pseudopolynomial-time approximation scheme. First, note that we can solve a (P,ε,z,h) discrete game in time polynomial in ε^{-1}:

1. Each graph has polynomial size: the area of the convex hull of P and the length of the perimeter of P are both polynomial, so the sizes of the graphs are polynomial, so the game can only be in polynomially many states, described by a human vertex, a zombie vertex, and whose turn it is.

2. We can calculate all legal transitions between pairs of game states in polynomial time.

3. We can calculate all winning positions in the discrete game: First mark as human wins all game states for which the human is at a vertex corresponding to a point on δP, the zombie is not there, and it's the human's turn to move. Then, for at most as many rounds as the (polynomial) number of possible game states, mark each game state as a human win if either

   (a) it's the human's turn and they can move to any game state already marked as a human win, or

   (b) it's the zombie's turn and every game state they can move to is already marked as a human win.

   After at most as many rounds as the number of game states, every game state from which the human wins will be so marked since, at each round, either at least one game state not previously marked as a human win will be or no new game states will be marked and every following round will be the same. In each round, we do polynomially much work, making this scheme polynomial.

4. The human wins the discrete game if and only if there's a human starting position (x,y) such that for every zombie starting position (x',y'), the state with the human at (x,y), the zombie at (x',y'), and the zombie to move is a human win.

Second, given an ε, we can approximate the critical speed ratio to within $(1+\varepsilon)^6$ by binary search:

1. The critical speed ratio is between 1 and $\frac{1}{2\sqrt{\varepsilon_0}}$ (as determined above in the definition of ε_0). That is, if $h_0 = \lfloor \varepsilon^{-5} \rfloor$, so there is some integer $z_0 \in [h, \frac{h}{2\sqrt{\varepsilon_0}}]$ such that the human wins the $(P,\varepsilon^5,z_0,h_0)$ discrete game and the zombie wins the $(P,\varepsilon^5,z_0+1,h_0)$ discrete game. Binary search for z_0, which takes at most $\log_2(\frac{1}{2\varepsilon^5\sqrt{\varepsilon_0}}) = O(\varepsilon^{-1})$ games, each of which takes time polynomial in ε^{-1}.

2. The interval in which the theorem says nothing about the winner of the (P,ε^5,z,h) game is a factor of $(1+\varepsilon)^6$, so the previous step tells us the critical speed ratio r^* to within a factor of $(1+\varepsilon)^6$.

The key part of proving Theorem 2.3.3 is to prove the seemingly innocuous claim that if the human can win the continuous game at all, and then the zombie becomes slightly slower, the human can win with a bit of time to spare; this is Lemma 2.3.7. To prove this, though, we first need some other lemmas:

Lemma 2.3.4. *If the human wins the continuous game, then there exists $\varepsilon > 0$ (not necessarily bounded by a function of ε_0) such that the zombie is at distance at least ε when the human wins.*

Proof. When the human wins, the zombie is not at the human's location, so the zombie's distance to the human is some positive number. □

Lemma 2.3.5. *If the human wins the continuous game at a speed ratio r, then there exists $\varepsilon > 0$ (not necessarily bounded by a function of ε_0) such that the human can commit to moving in a straight line for the last ε of their movement, and still win.*

Proof. Suppose that when the human wins at a point $p \in \delta P$ the zombie is at distance ε. At time less than $\varepsilon\frac{1}{2r+3}$ before the human wins, the human is at distance less than $\varepsilon\frac{1}{2r+3}$ from p, and the zombie is at distance more than $\varepsilon\frac{r+2}{2r+3}$ from the human (because in time less than $\varepsilon\frac{1}{2r+3}$, the distance between the zombie and human decreases by less than $\varepsilon\frac{r+1}{2r+3}$ and reaches at least $\varepsilon\frac{2r+3}{2r+3}$). Therefore, if the human runs straight toward p at that point, they either

1. get to it first and win, or
2. hit another point on the boundary, and also win because in time less than $\varepsilon\frac{1}{2r+3}$, the distance between the human and zombie decrease by less than $\varepsilon\frac{r+1}{2r+3}$ from its starting value of $\varepsilon\frac{r+2}{2r+3}$.

Either way, the human ran straight for some positive distance. □

Lemma 2.3.6. *If the zombie has a winning strategy that leaves the convex hull of P, then it has a winning strategy that doesn't.*

Proof. Let Zarathustra be the zombie with a winning strategy that leaves the convex hull of P. Another zombie, Zane, can win without leaving the convex hull by simulating Zarathustra's strategy and staying at the closest point on the convex hull to Zarathustra. That closest point can't move faster than Zarathustra does, since the closest point to Zarathustra on every edge of the convex hull moves at most as fast as Zarathustra does, and the closest point moves continuously. □

Definition 2.3.3. Let $G_{\text{distance } \varepsilon}$ be the game which is the same as the original game, except that the human wins by reaching the boundary while the zombie is at a distance greater than ε. (Perhaps the human needs time to start up a getaway car?) In particular, G_0 is the original (continuous) game.

Lemma 2.3.7. *If P is a polygon, then there exists $\varepsilon_0 > 0$ (the same as in Theorem 2.3.3) such that for all $\varepsilon \in (0, \varepsilon_0)$, if the human wins the continuous game G_0 in a polygon P at a speed ratio r, then the human wins the game $G_{distance\ \varepsilon^3}$ at speed ratio $r\frac{1}{1+\varepsilon}$.*

Proof. The human should start at some point h at distance more than ε from the nearest boundary; ε_0 was chosen small enough that such a place exists. The human can still win G_0: if they could win by some other starting position, the human can immediately run to that position; wherever the zombie is after that run, the zombie could have started, so the human can simulate their winning strategy from that starting position to win.

If, from that starting position, there's a point p on the boundary such that the human can win G_0 with speed ratio r by committing to running straight to p (that is, if there's a point $p \in \delta P$ such that $r \cdot d_h(h,p) < d_z(z,p)$, where z is the zombie's starting position), then the human can win $G_{\text{distance } \varepsilon^2}$ with speed ratio $r\frac{1}{1+\varepsilon}$ by running straight to that point. The human's time to get there is $d_h(h,p)$, in which time the zombie moves at most $r\frac{1}{1+\varepsilon} \cdot d_h(h,p) < \frac{1}{1+\varepsilon} \cdot d_z(z,p)$, leaving a distance of at least $\frac{\varepsilon}{1+\varepsilon} \cdot d_z(z,p) > \frac{\varepsilon}{2r} \cdot d_h(h,p) > \frac{\varepsilon^2}{2r} > \varepsilon^3$, as desired.

Otherwise, the human can't immediately win G_0 with speed ratio r by picking a point on δP and running straight to it, but can eventually win by doing so by Lemma 2.3.5, so consider the human's position h and zombie's position z at the last time when the human can't so win.

When the human is at h and the zombie at z, there's at least one point on the boundary that the human can reach in the same time as the zombie, since, by the choice of h, after any positive amount of movement, there's a point on the boundary that the human can get there in less time than the zombie.

If there's any such boundary point at distance more than ε from h, then by the same calculation as above, the human can win $G_{\text{distance } \varepsilon^3}$ at speed ratio $r\frac{1}{1+\varepsilon}$ by running straight to it. Otherwise, every such boundary point is within ε of h.

We chose ε_0 small enough that no disk of radius $2\sqrt{\varepsilon_0}$ contains two edges not sharing a vertex. If all such boundary points are within ε of h, note that the disk of radius $2\sqrt{\varepsilon_0}$ centered at h contains at least one edge (one with such a boundary point) and at most two; if two, they share a vertex. Note that the zombie's distance to any such boundary point is at most $r\varepsilon$. As above, we have $\varepsilon < \varepsilon_0 < 1/(2r(P)^2) \leq 1/(2r^2) < r^{-2}$ (where $r(P)$ is the critical speed ratio for P), so the zombie's distance to each of them is at most $r\varepsilon < \sqrt{\varepsilon}$, and the human's distance to each of them is also at most that, so z is in a circle of radius $2\sqrt{\varepsilon_0}$ centered at h.

But we claim that the human can't win at all with at most ε more movement, much less by committing to moving straight to one of those boundary points within ε of h, contradicting the choice of h. The zombie can use the following strategy: keep the line between it and the human parallel to hz, and use any remaining movement to move toward the human, if the zombie isn't already as close as it can get (that is, on δP). If a zombie follows this strategy, the only relevant distances are the distances on the line between the human and zombie. For any direction the human runs in, the zombie's distance on that line decreases at least as fast (as a fraction of its total distance) as the human's does (as a fraction of its total distance); otherwise, the human could win by running straight in that direction, but we assumed the human couldn't yet win by doing so.

So as long as the human and zombie stay within that circle of radius $2\sqrt{\varepsilon_0}$, the human can't win, contradicting the assumption that, a moment later, the human could win by running straight a distance at most ε.

In every surviving case, the human can win $G_{\text{distance } \varepsilon^3}$ with speed ratio $r\frac{1}{1+\varepsilon}$, as desired. □

Definition 2.3.4. Let $G_{\text{delay } \varepsilon}$ be the game where the human can only see and react to where the zombie was a time ε ago (perhaps due to the finite speed of neural impulses in the human's delicious brain?); the human wins by reaching a point on the boundary that the zombie can't get to even with ε more time.

Lemma 2.3.8. *If the human has a winning strategy in $G_{distance\ \varepsilon}$, then the human has a winning strategy in $G_{delay\ \varepsilon/2r}$, where r is the speed ratio.*

Proof. If the human, Alice, has a winning strategy in $G_{\text{distance } \varepsilon}$, we wish to construct a winning strategy for the human, Bob, in $G_{\text{delay } \varepsilon/2r}$. Let Alice's zombie move exactly as Bob's did a time $\frac{\varepsilon}{2r}$ ago, and let Bob move exactly as Alice does; Bob can do so because Alice's moves at time t depend only on the position of Alice's zombie at time at most t, which is the position of Bob's zombie at time at most $t - \frac{\varepsilon}{2r}$, which Bob knows at time t. Since Alice has a winning strategy in her game, she has a winning strategy against any zombie strategy, in particular, against this taking-orders-from-Bob strategy. So, Alice wins her game, that is, she reaches the boundary while her zombie is still at a distance at least ε. Bob reaches the boundary at the same time. Since Alice's zombie is at distance more than ε from her, Bob's zombie was at distance more than ε a time $\frac{\varepsilon}{2r}$ ago, and in that much time the zombie can only move a distance less than $\frac{\varepsilon}{2}$. That leaves the zombie at a distance more than $\frac{\varepsilon}{2}$, so the zombie can't get to Bob's position even with $\frac{\varepsilon}{2r}$ more time, as claimed. □

By the composition of Lemmas 2.3.4, 2.3.5, 2.3.7, and 2.3.8, if the human wins at a speed ratio r in a polygon P, then there exists $\varepsilon_0 > 0$, depending only on P, such that for all $\varepsilon \in (0, \varepsilon_0)$, the human wins $G_{\text{delay } \varepsilon^3/2r}$ with speed ratio $r\frac{1}{(1+\varepsilon)}$; also, since $2r\varepsilon < 1$, the human wins $G_{\text{delay } \varepsilon^4}$ with that speed ratio.

We are now ready to prove Theorem 2.3.3.

Proof. Let P be a polygon with critical speed ratio r^*, choose ε_0 small enough for Lemma 2.3.8, and let $\varepsilon \in (0, \varepsilon_0)$. Then, by the definition of r^*, the human wins the continuous game at a speed ratio of $r^*\frac{1}{1+\varepsilon}$. So, by Lemma 2.3.8, the human has a winning strategy in the delayed-information game $G_{\text{delay } \varepsilon^4}$ with speed ratio $r^*\frac{1}{(1+\varepsilon)^2}$.

Suppose z and h satisfy the conditions of Theorem 2.3.3, that is, they're integers with $0 < z, h < \varepsilon^{-1}$ and $z/h < r^*\frac{1}{(1+\varepsilon)^3}$. We'll construct a winning strategy for the human, Bob, in the (P, ε^5, z, h) discrete game. Bob will simulate an Alice playing the $G_{\text{delay } \varepsilon^4}$ game; if Bob has made m moves in his game, he'll make that correspond to a time $m\frac{z}{h}\frac{1}{r^*}\varepsilon^5(1+\varepsilon)^2$ in Alice's simulated $G_{\text{delay } \varepsilon^4}$ game.

Bob's strategy is to follow Alice as closely as he can, by ensuring that after m moves he is within ε^8 of Alice's position at time $m\frac{z}{h}\frac{1}{r^*}\varepsilon^5(1+\varepsilon)^2$. To enact this strategy, Bob needs to find a vertex within ε^8 of any point in P, be able to move there in time, and be able to answer Alice's questions about where the zombie is.

1. First, we claim that there is a point of the discrete game within ε^8 of any point in P where Alice could be: if the circle of radius ε^9 centered at Alice's position is contained in P, then it contains a point of $\varepsilon^{10}\mathbb{Z}^2$, which is a point of the graph; otherwise, Alice is within ε^9 of the boundary, in which case the circle of radius ε^8 centered at Alice contains at least $2\varepsilon^8 - 2\varepsilon^9 > \varepsilon^8$ of the boundary, and there are points on the boundary spaced at distance between ε^{10} and $2\varepsilon^{10}$, so one of them is within that circle.

2. Second, we claim that Bob can follow Alice's movement in time at most $\frac{z}{h}\frac{1}{r^*}\varepsilon^5(1+\varepsilon)^2$ in one step. In time $\frac{z}{h}\frac{1}{r^*}\varepsilon^5(1+\varepsilon)^2$, Alice moves a distance at most $\frac{z}{h}\frac{1}{r^*}\varepsilon^5(1+\varepsilon)^2$. Bob starts within ε^8 of Alice's starting position and ends within ε^8 of Alice's ending position, so by the triangle inequality, the distance between Bob's starting and ending vertices is at most
$$\begin{aligned}\frac{z}{h}\frac{1}{r^*}\varepsilon^5(1+\varepsilon)^2 + 2\varepsilon^8 &\le \frac{z}{h}\frac{1}{r^*}(\varepsilon^5(1+\varepsilon)^2 + \varepsilon^8 r^*) \\ &< \frac{z}{h}\frac{1}{r^*}(\varepsilon^5(1+\varepsilon)^2 + \varepsilon^6) \\ &< \frac{z}{h}\frac{1}{r^*}(\varepsilon^5(1+\varepsilon)^3) \\ &< \varepsilon^5,\end{aligned}$$
so Bob's starting vertex and desired ending vertex are adjacent in the graph, and he can keep up with the simulated Alice.

3. When Alice asks where the zombie is, she's only allowed to ask about times at least ε^4 ago. A time of ε^4 in her game corresponds to at least $\lfloor \frac{h}{z} r^* \frac{\varepsilon^4}{\varepsilon^5} \frac{1}{(1+\varepsilon)^2} \rfloor \geq \lfloor \frac{1}{\varepsilon} \rfloor \geq h$ steps for Bob in Bob's game. Bob and his zombie alternate, with Bob taking h steps for every z of his zombie's, so if Alice asks about a time at least h steps ago in Bob's game, the zombie has moved in Bob's game since then, so Bob knows its position on the graph for his game up until the time Alice asks about.

4. One step by the zombie in Bob's game corresponds to a time $\frac{1}{r^*}\varepsilon^5(1+\varepsilon)^2$ in Alice's simulated $G_{\text{delay } \varepsilon^4}$ game. In that much time, Alice's zombie, which has speed $r^* \frac{1}{(1+\varepsilon)^2}$, is allowed to move a distance of ε^5. Every pair of adjacent vertices in Bob's zombie's graph correspond to points at distance at most ε^5 for Alice's zombie, so Alice's zombie can keep up with the position of Bob's zombie.

So, Bob can simulate Alice's game of $G_{\text{delay } \varepsilon^4}$ with speed ratio $r^* \frac{1}{(1+\varepsilon)^2}$. Alice has a winning strategy for that game; that is, she reaches a point on the boundary when her zombie can't get there in time ε^4. In time ε^4, her zombie can move a distance of $r^* \varepsilon^4 \frac{1}{(1+\varepsilon)^2}$, so her zombie is at least that distance away, corresponding to at least $r^* \frac{1}{\varepsilon} \frac{1}{(1+\varepsilon)^2} > \frac{z}{h} \frac{1}{\varepsilon}(1+\varepsilon) > \frac{z}{h} \frac{1}{\varepsilon} > z$ steps in Bob's zombie's graph, so Bob's zombie can't reach Bob in the turn after Bob reaches the boundary following Alice's strategy, and Bob wins, as desired.

For the other direction, if $z/h > r^*(1+\varepsilon)^3$, we need to show that a human can't win the (P, ε^5, z, h) game, so suppose for contradiction that Alice has a winning strategy in that game; we'll construct a winning strategy for the human, Bob, in the original game with a speed ratio of $\frac{z}{h} \frac{1}{(1+\varepsilon)^3} > r^*$, contradicting the definition of r^*. Bob will make it so that one step for Alice in the simulated (P, ε^5, z, h) correspond to a time of ε^5 in his game; one step for Alice is a distance of at most ε^5, so Bob can keep up with the simulated Alice. When Alice's zombie needs to be given instructions at the end of a block of h of Alice's moves, corresponding to time $h\varepsilon^5$ for Bob, Bob will use Bob's zombie's moves from the last $h\varepsilon^5$ time, with each of the zombie's z steps in that time corresponding to $\frac{h}{z}\varepsilon^5$ time in Bob's game. Bob will instruct Alice's simulated zombie to, at each step, move to a point within ε^8 of where Bob's zombie is. There always is such a point, by the same argument as for Bob in the other direction of this proof, with the added note that we can assume, by Lemma 2.3.6, that the zombie is in the convex hull (where there are points of the zombie's graph). Bob's zombie moves a distance of at most $\frac{h}{z}\varepsilon^5 r^* < \varepsilon^5 \frac{1}{(1+\varepsilon)^3}$ in that time, so the distance between the point corresponding to Alice's zombie's start vertex and the point corresponding to Alice's zombie's target vertex is at most $\varepsilon^5 \frac{1}{(1+\varepsilon)^3} + 2\varepsilon^8$ by the triangle inequality, and that's at most ε^5, so Alice's zombie can keep up with Bob's. By assumption, Alice wins her game, so she reaches the boundary when her zombie can't, even with z more steps. Those z steps let Alice's zombie's time catch up with Bob's zombie's time (when Alice moves h steps and then Alice's zombie moves z steps, those correspond to the same time interval in Bob's game), so when Bob reaches the boundary, Bob's zombie isn't there and he wins, as desired. □

This proof of Theorem 2.3.3 suffices for a polynomial-time approximation scheme for the critical speed ratio for any particular polygon P, as discussed just after the statement of Theorem 2.3.3.

2.4 Multiple Zombies

In the previous section, we discussed approximating the critical speed ratio below which a human can win and above which they can't. In the next section, we'll prove the computational hardness of calculating or approximating that critical speed ratio under slightly different sets of assumptions; in particular, all of the hardness proofs will require that there be multiple zombies, not just one, such that any one of them can block the human's escape; some will also require that there be multiple humans, who win if at least one escapes. To make the hardness proofs of the next section more satisfying, therefore, we first, in this section, discuss what we can determine when there are multiple zombies and possibly multiple humans.

Theorem 2.4.1. *Every human can escape in a game with multiple humans if and only if the lone human could escape in the same game with only one human.*

Proof. If one human can escape in the game with only one human, all the humans can stay together, moving as that one human would, and escape. If the zombies can keep a lone human from escaping, they can ignore all but one of the humans and keep that human from escaping. □

Given this result, we'll assume that, if there are multiple humans, the goal is for at least one human to escape, perhaps to call for help.

Theorem 2.4.2. *If there are n_z zombies and $n_h > n_z$ humans, then one human can always escape.*

Proof. Each of the humans can stand at a distinct one of n_h spots along the boundary. At at least $n_h - n_z$ of those spots, there's no zombie, so the humans at those spots escape. □

2.4.1 Approximation Algorithms

Theorem 2.3.3 in Section 2.3.1 still gives a pseudopolynomial approximation scheme if there are multiple (but $O(1)$) humans and/or zombies. The proof is essentially the same: we can solve a discrete game with $O(1)$ zombies, and the critical speed ratio is bounded above by the critical speed ratio for one zombie.

One side of Theorem 2.3.2 has an analogue:

Theorem 2.4.3. *If P is a polygon and there are n_z zombies and one human, then the zombies win if their speed is at least the minimum over partitions of the boundary into (not necessarily connected) regions of*

$$\max_{p,q \text{ in same region}} \frac{d_z(p,q)}{d_h(p,q)}.$$

Proof. Each zombie can ignore all of the boundary but the part assigned to it and use the strategy of Theorem 2.3.2. □

However, for the other side we have no analogue.

Open Problem 1. *Does there exist $c > 0$ such that if P is a polygon and there are n_z zombies and one human, then the human wins if the zombies' speed is less than the minimum over partitions of the boundary into (not necessarily connected) regions of*

$$c \cdot \max_{p,q \text{ in same region}} \frac{d_z(p,q)}{d_h(p,q)}.$$

2.4.2 Slow Zombies

When there was only one zombie, it only made sense to consider cases where the zombie was faster than the human; if the zombie is the same speed as the human or slower and there's any convex vertex, a human standing near it can win. With multiple zombies, it's still only nontrivial if the zombies are at least as fast as the human, since if the human's faster, it can stand near any edge and win. However, the case where the humans and the zombies have the same speed becomes interesting.

Theorem 2.4.4. *If the speeds of the human and zombies are equal, and the exterior of the polygon can be divided into n_z convex regions that cover the boundary of the polygon, then the zombies win.*

Proof. Each zombie can stay in one region, staying at the closest point in that region to the human. The closest point in a convex region to the human can't move faster than the human can, so the zombies can keep up with this strategy. If the human reaches the boundary, there's a zombie region containing that boundary, and therefore a zombie at the closest point in that region to the human, which is the human's location itself. So, the human can't escape. □

Corollary 2.4.5. *If a polygon P has n vertices and there are n zombies with the same speed as the human, the human can't win.*

Proof. One zombie can cover each edge. □

Theorem 2.4.6. *If P is a convex n-gon and there's one human and $\lceil \frac{n}{2} \rceil + 1$ zombies, all with the same speed, then the human can win.*

Proof. The human should start at any vertex h on the boundary. Let h' be the point opposite h on δP, that is, the point for which the zombie distance from h is maximal. The points h and h' split δP into two sections, at least one of which must have at least $\lceil \frac{n}{2} \rceil$ vertices (counting h but not h'). The human should run along that section of perimeter. If at some vertex there's only one zombie, then the human can approach one edge not at the vertex, forcing the zombie to that edge, then shortcut through the polygon to a point near the vertex but on the other edge and escape. If there are two zombies, the human can do the same thing to ensure that at least one of them is behind the human when the human moves on to the next vertex. So, for each of the $\lceil \frac{n}{2} \rceil$ vertices, there must be at least one new zombie guard, plus one zombie guard at the center, and these must all be distinct because the zombies from h don't have time to run around past h' before the human gets there. □

Although Theorem 2.4.6 is, like all other results in this chapter for which the model is unspecified, true for both the standard model and the moat model as defined in Section 2.1, we can make a slightly stronger statement in the moat model, with the same proof: even if P is nonconvex, if it has c *convex* vertices, then the human can escape from $\lceil \frac{c}{2} \rceil + 1$ zombies of the same speed as theirs.

There is no analogous lower bound, because 4 zombies suffice to guard polygons like the one in Figure 2.5 with arbitrarily many vertices. Two zombies can stay on the top and two on the bottom; each of those can be assigned to guard every other triangular region of the convex hull outside P.

2.5 Computational Complexity

In this section, we prove NP-hardness and hardness of approximation results, as specified in Table 2.1, for problems of escaping from zombies with various combinations of parameters:

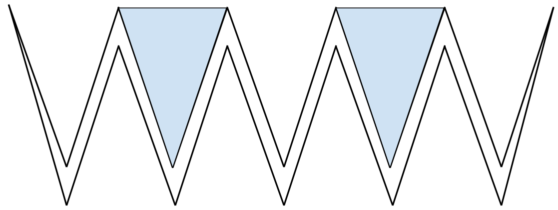


Figure 2.5: A polygon guardable by 4 zombies with speed equal to the human's, with the (disconnected) region for one zombie to guard shaded blue.

1. There could be one human, as in the original problem, or many, as discussed in Section 2.4.
2. There could be one zombie, as in the original problem, or many, as discussed in Section 2.4.
3. In the original problem, the human and zombie moved in a polygon with boundary and its complement with boundary, respectively. Here we reduce the space from a 2-dimensional polygon to a 1-dimensional graph, on which we might also be able to impose the additional constraint that that graph be planar or connected.
4. In the original problem, a human could move into a spot where a zombie was (but not declare victory); here we may make zombies block human movement.

Theorem 2.5.1. *In a game in the graph model (see Section 2.1) in which the zombies win if a zombie is ever at the same place as a human and there are multiple humans of which only one needs to escape, it's* NP*-hard to decide whether the humans win, even if the graph is planar. Since the zombies' movement is irrelevant*[3]*, it's* NP*-hard to distinguish a critical speed ratio of* 0 *from* ∞*.*

Proof. We reduce from the Planar Vertex Cover problem of finding a set of at most k vertices in a planar graph such that every edge contains at least one of them, which [Lic82] shows to be NP-hard. Given an instance of Planar Vertex Cover consisting of a planar graph G with e edges and a target number of vertices k, make a zombie problem with a drawing of that graph for the humans, an exit vertex (where a zombie could stand and block movement) on each edge, k humans, and $e-1$ zombies; the zombies have nowhere to move.

If there exists a planar vertex cover, the humans can start at the vertices corresponding to it; then there's at least one edge that no zombie starts on, and a human who starts at a vertex contained in that edge can escape by that edge.

If the humans can win the zombie problem, consider the connected components of the drawing of G minus the exit vertices; there's one of them per vertex of G. Each human starts in one of them, and can't change between them except by passing through an exit vertex, at which point they could just escape. They start in at most k vertices, which we choose as the vertex cover. If there's any edge those vertices don't cover, the zombies can choose to start everywhere but that

[3]Perhaps these are zombie plants?

Humans	Zombies	Geometry	Zombies block	Result	Theorem
Multiple	Multiple	Planar graph	Yes	NP-hard, inapproximable	Theorem 2.5.1
Multiple	Multiple	Connected graph	No	Exp-APX-hard	Theorem 2.5.2
1	Multiple	Graph	Yes	2-inapproximable	Theorem 2.5.3

Table 2.1: Complexity results and the assumptions they require.

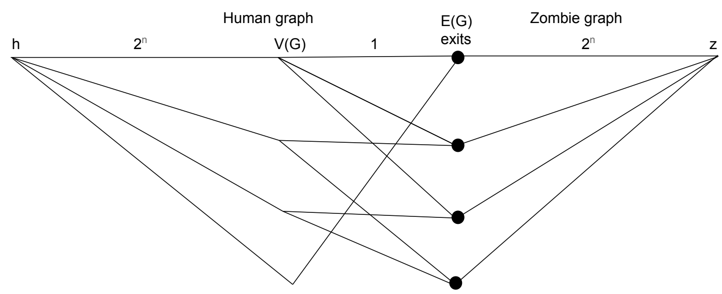


Figure 2.6: A graph for which it's EXP-APX-hard to determine the critical speed ratio.

edge's exit vertex, and the humans can't escape; since the humans win, they must cover every edge, as desired. □

Theorem 2.5.2. *In a game in the graph model in which the zombies win if a zombie is ever at the same place as a human and there are multiple humans of which only one needs to escape, it's* EXP-APX-*hard to find the critical speed ratio* r^*. *That is, unless* P = NP*, there's no polynomial-time algorithm to approximate* r^* *to within a factor exponential in the input length.*

The exponential inapproximability comes from the lengths of the edges. If the edge lengths are integers at most L, then the same proof shows that it's hard to approximate r^* to within a factor of L.

Proof. We reduce from the Vertex Cover problem of finding a set of at most k vertices in a graph such that every edge contains at least one of them, which is one of Karp's original 21 NP-hard problems (from [Kar72]). Given an instance of Vertex Cover consisting of a graph G with e edges and a target number of vertices k, make a zombie problem with k humans, $e-1$ zombies, and graphs as shown in Figure 2.6: have a human-accessible vertex for each vertex in $V(G)$, edges of length 2^n connecting each of them to a common vertex h, an exit vertex for each edge in $E(G)$, edges of length 2^n connecting each of them to a common vertex z, and edges of length 1 connecting each edge to its incident vertices.

If there exists a vertex cover, then the humans can start at the vertices corresponding to it. Then there's at least one edge that no zombie starts within 2^n of, and a human who starts at a vertex contained in that edge can escape by that edge in time 1, so the critical speed ratio is at least 2^n.

If there's no vertex cover, then we claim that zombies of speed 4 can win. At all times, there's an exit that there's no human within a distance 2^{n-1} of (otherwise the set of vertices that humans are closest to is a vertex cover). The zombies should start at $e-1$ exits including every exit there's a human within 2^{n-1} of. Whenever a human comes within 2^{n-1} of the unguarded exit, there's a zombie at an exit that no human's near; that zombie should run to the newly-threatened exit, a distance of 2^{n+1}, which the zombie of speed 4 can cover before the human either reaches the exit or goes back through h to threaten another exit.

So, if we could determine whether the critical speed ratio is at most 4 or at least 2^n, we could solve the vertex cover problem, making this problem EXP-APX-hard, as desired. □

Theorem 2.5.3. *In a game in the graph model in which the zombies win if a zombie is ever at the same place as a human, it's* NP*-hard to approximate the critical speed ratio* r *to within a factor of* 2.

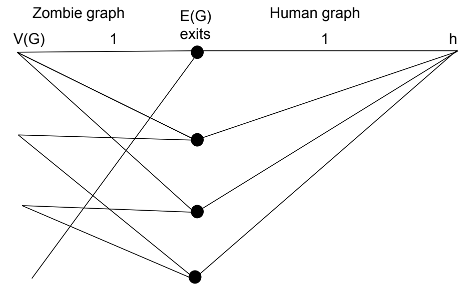


Figure 2.7: A graph with one human for which it's NP-hard to determine the critical speed ratio.

Proof. We again reduce from the Vertex Cover problem of finding a set of at most k vertices in a graph such that every edge contains at least one of them, which is one of Karp's original 21 NP-hard problems (from [Kar72]). Given an instance of Vertex Cover consisting of a graph G with e edges and a target number of vertices k, make a zombie problem with 1 human, $k-1$ zombies, and graphs as shown in Figure 2.6: have a zombie-accessible vertex for each vertex in $V(G)$, an exit vertex for each edge in $E(G)$, human-accessible edges of length 1 connecting each of them to a common vertex h, and edges of length 1 connecting each edge to its incident vertices.

If there's a vertex cover, the zombies can place themselves at the vertices of it, and whenever the human moves toward an exit, the zombie on the vertex that covers the edge corresponding to that exit can move to block it (and move back as the human does, staying exactly as close to the exit as the human is); in this way, even zombies of the same speed as the human can prevent human escape.

If there's no vertex cover, the human can start at h, and there's an exit that no zombie is within distance 2 of: only a zombie within distance 1 of a vertex contained in that exit's corresponding edge is within distance two of the exit, and the regions within distance 1 of each vertex are disjoint, so if there were a zombie within distance 2 of every vertex, that'd give a vertex cover. The human can run straight to that exit, and not even a speed 2 zombie can catch it.

So it's NP-hard to distinguish a critical speed ratio of at most 1 from one at least 2, as claimed. $\square$

2.6 Open Problems

The following are some possible directions for further work:

1. Section 2.3 gives only a pseudopolynomial-time approximation scheme for the critical speed ratio for a polygon P. Is this the best one can do, or is there an approximation scheme whose time depends only polynomially on at least the length of the description of P, if not also on $\log \frac{1}{\varepsilon}$?

2. Theorem 2.3.2 bounds the critical speed ratio between 1 and 9.2504 times $\max_{p,q \in \delta P} \frac{d_z(p,q)}{d_h(p,q)}$. What's the range of possible values of that constant? (For a circle, it's $4.60/(\pi/2) \approx 2.93$, but we conjecture that it's higher for an equilateral triangle.)

3. Is there an analogue of Theorem 2.3.2 describing the critical speed ratio to within a constant factor when there are *two* (or $O(1)$) zombies?

   The most obvious analogue, using a 2nd-order Voronoi diagram, does not work: if P is a long, thin rectangle with one long side subdivided, one zombie should stay on each side, but a 2nd-order Voronoi diagram might put both zombies on one side.

The other most obvious analogue would have one zombie attempts to guard the edge the human is closest to, the second zombie greedily guards whatever point the first zombie would have the most trouble reaching, and both zombies delay changing their strategies by the use of fringe regions as in Theorem 2.3.2, but the human might exit multiple fringes simultaneously, which seems hard for the zombies to account for without paying an extra factor equal to the number of zombies.

4. We've calculated the exact critical speed ratio for circles and for unbounded intersections of halfplanes, but for even for the simplest bounded intersection of halfplanes, an equilateral triangle, we can't calculate the exact speed ratio.

Chapter 3

Conflict-Free Graph Coloring

This chapter is joint work with Zachary Abel, Victor Alvarez, Erik Demaine, Sándor Fekete, Aman Gour, Phillip Keldenich, and Christian Scheffer, with help from discussions with Bruno Crepaldi, Pedro de Rezende, Cid de Souza, Stephan Friedrichs, Michael Hemmer, and Frank Quedenfeld. It has appeared in the proceedings of the ACM-SIAM Symposium on Discrete Algorithms [AAG+18].

3.1 Introduction

Coloring the vertices of a graph is one of the fundamental problems in graph theory, both scientifically and historically. Proving that four colors always suffice to color a planar graph [AH77a, AH77b, RSST97] was a tantalizing open problem for more than 100 years; the quest for solving this challenge contributed to the development of graph theory, but also to computers in theorem proving [Wil13]. A generalization that is still unsolved is the *Hadwiger Conjecture* [Had43]: A graph is k-colorable if it has no K_{k+1} minor.

Over the years, there have been many variations on coloring, often motivated by particular applications. One such context is wireless communication, where "colors" correspond to different frequencies. This also plays a role in robot navigation, where different beacons are used for providing direction. To this end, it is vital that in any given location, a robot is adjacent to a beacon with a frequency that is unique among the ones that can be received. This notion has been introduced as *conflict-free coloring*, formalized as follows. For any vertex $v \in V$ of a simple graph $G = (V, E)$, the *closed neighborhood* $N[v]$ consists of all vertices adjacent to v and v itself. A *conflict-free k-coloring* of G assigns one of k different colors to a (possibly proper) subset $S \subseteq V$ of vertices, such that for every vertex $v \in V$, there is a vertex $y \in N[v]$, called the *conflict-free neighbor* of v, such that the color of y is unique in the closed neighborhood of v. The *conflict-free chromatic number* $\chi_{CF}(G)$ of G is the smallest k for which a conflict-free coloring exists. Observe that $\chi_{CF}(G)$ is bounded from above by the proper chromatic number $\chi(G)$ because in a proper coloring, every vertex is its own conflict-free neighbor.

Similar questions can be considered for *open neighborhoods* $N(v) = N[v] \setminus \{v\}$.

Conflict-free coloring has received an increasing amount of attention. Because of the relationship to classic coloring, it is natural to investigate the conflict-free coloring of planar graphs. In addition, previous work has considered either general graphs and hypergraphs (e.g., see [PT09]) or geometric scenarios (e.g., see [HKS+15]); we give a more detailed overview further down. This adds to the relevance of conflict-free coloring of planar graphs, which constitute the intersection of general graphs and geometry. In addition, the subclass of outerplanar graphs is of interest, as it corresponds to subdividing simple polygons by chords.

There is a spectrum of different scientific challenges when studying conflict-free coloring. What are worst-case bounds on the necessary number of colors? When is it NP-hard to determine the existence of a conflict-free k-coloring, when polynomially solvable? What can be said about approximation? Are there sufficient conditions for more general graphs? And what can be said about the bicriteria problem, in which also the number of colored vertices is considered? We provide extensive answers for all of these aspects, basically providing a complete characterization for planar and outerplanar graphs.

3.1.1 Our Contribution

We present the following results; items 1-7 are for closed neighborhoods, while items 8-11 are for open neighborhoods.

1. For general graphs, we provide the conflict-free variant of the Hadwiger Conjecture: If G does not contain K_{k+1} as a minor, then $\chi_{CF}(G) \leq k$.

2. It is NP-complete to decide whether a planar graph has a conflict-free coloring with *one* color. For outerplanar graphs, this question can be decided in polynomial time.

3. It is NP-complete to decide whether a planar graph has a conflict-free coloring with *two* colors. For outerplanar graphs, two colors always suffice.

4. Three colors are sometimes necessary and always sufficient for conflict-free coloring of a planar graph.

5. For the bicriteria problem of minimizing the number of colored vertices subject to a given bound $\chi_{CF}(G) \leq k$ with $k \in \{1, 2\}$, we prove that the problem is NP-hard for planar and polynomially solvable in outerplanar graphs.

6. For planar graphs and $k = 3$ colors, minimizing the number of colored vertices does not have a constant-factor approximation, unless P = NP.

7. For planar graphs and $k \geq 4$ colors, it is NP-complete to minimize the number of colored vertices. The problem is fixed-parameter tractable (FPT) and allows a PTAS.

8. Four colors are sometimes necessary and always sufficient for conflict-free coloring with open neighborhoods of planar bipartite graphs.

9. It is NP-complete to decide whether a planar bipartite graph has a conflict-free coloring with open neighborhoods with k colors for $k \in \{1, 2, 3\}$.

10. Eight colors always suffice for conflict-free coloring with open neighborhoods of planar graphs.

3.1.2 Related Work

In a geometric context, the study of conflict-free coloring was started by Even, Lotker, Ron, and Smorodinsky [ELRS03] and Smorodinsky [Smo03], who motivate the problem by frequency assignment in cellular networks: There, a set of n base stations is given, each covering some geometric region in the plane. The base stations service mobile clients that can be at any point in the total covered area. To avoid interference, there must be at least one base station in range using a unique frequency for every point in the entire covered area. The task is to assign a frequency to each base station minimizing the number of frequencies. On an abstract level, this induces a coloring problem on a hypergraph where the base stations correspond to the vertices and there is an hyperedge between some vertices if the range of the corresponding base stations has a non-empty common intersection.

If the hypergraph is induced by disks, Even et al. [ELRS03] prove that $\mathcal{O}(\log n)$ colors are always sufficient. Alon and Smorodinsky [AS06] extend this by showing that each family of disks, where each disk intersects at most k others, can be colored using $\mathcal{O}(\log^3 k)$ colors. Furthermore, for unit disks, Lev-Tov and Peleg [LTP09] present an $\mathcal{O}(1)$-approximation algorithm for the number of colors. Horev et al. [HKS10] extend this by showing that any set of n disks can be colored with $\mathcal{O}(k \log n)$ colors, even if every point must see k distinct unique colors. Abam et al. [AdBP08] discuss the problem in the context of cellular networks where the network has to be reliable even if some number of base stations fault, giving worst-case bounds for the number of colors required.

For the dual problem of coloring a set of points such that each region from some family of regions contains at least one uniquely colored point, Har-Peled and Smorodinsky [HPS05] prove that with respect to every family of pseudo-disks, every set of points can be colored using $\mathcal{O}(\log n)$ colors. For rectangle ranges, Elbassioni and Mustafa [EM06] show that it is possible to add a sublinear number of points such that a conflict-free coloring with $\mathcal{O}(n^{3/8\cdot(1+\varepsilon)})$ colors becomes possible. Ajwani et al. [AEGR07] complement this by showing that coloring a set of points with respect to rectangle ranges is always possible using $\mathcal{O}(n^{0.382})$ colors. For coloring points on a line with respect to intervals, Cheilaris et al. [CGRS14] present a 2-approximation algorithm, and a $\left(5 - \frac{2}{k}\right)$-approximation algorithm when every interval must see k uniquely colored vertices. Hoffman et al. [HKS$^+$15] give tight bounds for the conflict-free chromatic art gallery problem under rectangular visibility in orthogonal polygons: $\Theta(\log \log n)$ are sometimes necessary and always sufficient. Chen et al. [CFK$^+$07] consider the online version of the conflict-free coloring of a set of points on the line, where each newly inserted point must be assigned a color upon insertion, and at all times the coloring has to be conflict-free. Also in the online scenario, Bar-Nov et al. [BNCOS10] consider a certain class of k-degenerate hypergraphs which sometimes arise as intersection graphs of geometric objects, presenting an online algorithm using $\mathcal{O}(k \log n)$ colors.

On the combinatorial side, some authors consider the variant in which all vertices need to be colored; note that this does not change asymptotic results for general graphs and hypergraphs: it suffices to introduce one additional color for vertices that are left uncolored in our constructions. Regarding general hypergraphs, Ashok et al. [ADK15] prove that maximizing the number of conflict-freely colored edges in a hypergraph is FPT when parameterized by the number of conflict-free edges in the solution. Cheilaris et al. [CSS11] consider the case of hypergraphs induced by a set of planar Jordan regions and prove an asymptotically tight upper bound of $\mathcal{O}(\log n)$ for the conflict-free list chromatic number of such hypergraphs. They also consider hypergraphs induced by the simple paths of a planar graph and prove an upper bound of $\mathcal{O}(\sqrt{n})$ for the conflict-free list chromatic number. For hypergraphs induced by the paths of a simple graph G, Cheilaris and Tóth [CT11] prove that it is coNP-complete to decide whether a given coloring is conflict-free if the input is G. Regarding the case in which the hypergraph is induced by the neighborhoods of a simple graph G, which resembles our scenario, Pach and Tárdos [PT09] prove that the conflict-free chromatic number of an n-vertex graph is in $\mathcal{O}(\log^2 n)$. Glebov et al. [GST14] extend this from an extremal and probabilistic point of view by proving that almost all $G(n, p)$-graphs have conflict-free chromatic number $\mathcal{O}(\log n)$ for $p \in \omega(1/n)$, and by giving a randomized construction for graphs having conflict-free chromatic number $\Theta(\log^2 n)$. In more recent work, Gargano and Rescigno [GR15] show that finding the conflict-free chromatic number for general graphs is NP-complete, and prove that the problem is FPT w.r.t. vertex cover or neighborhood diversity number.

3.2 Preliminaries

For every vertex $v \in V$, the *open neighborhood* of v in G is denoted by $N_G(v) := \{w \in V(G) \mid vw \in E(G)\}$, and the *closed neighborhood* is denoted by $N_G[v] := N_G(v) \cup \{v\}$. We sometimes write $N(v)$ instead of $N_G(v)$ when G is clear from the context.

A partial k-coloring of G is an assignment $\chi : V' \to \{1, \dots, k\}$ of colors to a subset $V' \subseteq V(G)$ of the vertices. χ is called *closed-neighborhood conflict-free k-coloring* of G iff, for each vertex $v \in V$, there is a vertex $w \in N_G[v] \cap V'$ such that $\chi(w)$ is unique in $N_G[v]$, i.e., for all other $w' \in N_G[v] \cap V'$, $\chi(w') \neq \chi(w)$. We call w the conflict-free neighbor of v. Analogously, χ is called *open-neighborhood conflict-free k-coloring* of G iff, for each vertex $v \in V$, there is a conflict-free neighbor $w \in N_G(v)$.

In order to avoid confusion with *proper k-colorings*, i.e., colorings that color all vertices such that no adjacent vertices receive the same color, we use the term *proper coloring* when referring to this kind of coloring. The minimum number of colors needed for a proper coloring of G, also known as the chromatic number of G, is denoted by $\chi_P(G)$, whereas the minimum number of colors required for a closed-neighborhood conflict-free coloring of G (G's *closed-neighborhood conflict-free chromatic number*) is written as $\chi_{CF}(G)$. The *open-neighborhood conflict-free chromatic number* of G is $\chi_O(G)$. To improve readability we sometimes omit the type of neighborhood if it is clear from the context.

Note that, because every vertex satisfies $v \in N[v]$, every proper coloring of G is also a closed-neighborhood conflict-free coloring of G, and thus $\chi_{CF}(G) \leq \chi_P(G)$. The same does not hold for open neighborhoods. There is no constant factor $c_1 > 0$ such that either $c_1 \cdot \chi_O(G) \leq \chi_P(G)$ or $c_1 \cdot \chi_P(G) \leq \chi_O(G)$ holds for all graphs G.

For closed neighborhoods, we define the *conflict-free domination number* $\gamma^k_{CF}(G)$ of G to be the minimum number of vertices that have to be colored in a conflict-free k-coloring of G. We set $\gamma^k_{CF}(G) = \infty$ if G is not conflict-free k-colorable. Because the set of colored vertices is a dominating set, the conflict-free domination number satisfies $\gamma^k_{CF}(G) \geq \gamma(G)$ for all k, where $\gamma(G)$, the domination number of G, is the size of a minimum dominating set of G. Moreover, for any graph, there is a $k \leq \gamma(G)$ such that $\gamma^k_{CF}(G) = \gamma(G)$.

We denote the complete graph on n vertices by $K_n := (\{1, \dots, n\}, \{\{u, v\} \mid u, v \in \{1, \dots, n\}, u \neq v\})$, and the complete bipartite graph on n and m vertices as $K_{n,m}$. We define the graph $K_n^{-3} := (V(K_n), E(K_n) \setminus E(K_3))$, which is obtained by removing any three edges forming a single triangle from a K_n.

We also provide a number of results for outerplanar graphs. An outerplanar graph is a graph that has a planar embedding for which all vertices belong to the outer face of the embedding. An outerplanar graph is called *maximal* iff no edges can be added to the graph without losing outerplanarity [BH94]. Maximal outerplanar graphs can also be characterized as the graphs having an embedding corresponding to a polygon triangulation, which illustrates their particular relevance in a geometric context. In addition, maximal outerplanar graphs exhibit a number of interesting graph-theoretic properties. Every maximal outerplanar graph is chordal, a 2-tree and a series-parallel graph. Also, every maximal outerplanar graph is the visibility graph of a simple polygon.

For some of our NP-hardness proofs, we use a variant of the planar 3-SAT problem, called Positive Planar 1-in-3-SAT. This problem was introduced and shown to be NP-complete by Mulzer and Rote [MR08], and consists of deciding whether a given positive planar 3-CNF formula allows a truth assignment such that in each clause, exactly one literal is true.

Definition 3.2.1 (Positive planar formulas)**.**
A formula ϕ in 3-CNF is called *positive planar* iff it is both *positive* and *backbone planar*. A formula ϕ is called *positive* iff it does not contain any negation, i.e. iff all occurring literals are positive. A

formula ϕ, with clause set $C = \{c_1, \dots, c_l\}$ and variable set $X = \{x_1, \dots, x_n\}$, is called *backbone planar* iff its associated graph $G(\phi) := (X \cup C, E(\phi))$ is planar, where $E(\phi)$ is defined as follows:

- $x_i c_j \in E(\phi)$ for a clause $c_j \in C$ and a variable $x_i \in X$ iff x_i occurs in c_j,
- $x_i x_{i+1} \in E(\phi)$ for all $1 \leq i < n$.

The path formed by the latter edges is also called the *backbone* of the formula graph $G(\phi)$.

3.3 Closed Neighborhoods: Conflict-Free Coloring of General Graphs

In this section we consider the Conflict-Free k-Coloring problem on general simple graphs with respect to closed neighborhoods. In § 3.3.1, we prove that this problem is NP-complete for any $k \geq 1$. In § 3.3.2, we provide a sufficient criterion that guarantees conflict-free k-colorability. In § 3.3.3, we consider the conflict-free domination number and prove that, for any $k \geq 3$, there is no constant-factor approximation algorithm for γ_{CF}^k.

3.3.1 Complexity

Theorem 3.3.1. Conflict-Free k-Coloring *is NP-complete for any fixed $k \geq 1$.*

Membership in NP is clear. For $k \geq 3$, we prove NP-hardness using a reduction from proper k-Coloring. For $k \in \{1, 2\}$, refer to § 3.4, where we prove Conflict-Free k-Coloring of planar graphs to be NP-complete for $k \in \{1, 2\}$.

Central to the proof is the following lemma that enables us to enforce certain vertices to be colored, and both ends of an edge to be colored using distinct colors.

Lemma 3.3.2. *Let G be any graph, $u, v \in V(G)$ and $vu = e \in E(G)$. If $N(v)$ contains two disjoint and independent copies of a graph H with $\chi_{CF}(H) = k$, not adjacent to any other vertex $w \in G$, every conflict-free k-coloring of G colors v. If the same holds for u and in addition, $N_G(u) \cap N_G(v)$ contains two disjoint and independent copies of a graph J with $\chi_{CF}(J) = k - 1$, not adjacent to any other vertex $w \in G$, every conflict-free k-coloring of G colors u and v with different colors.*

Proof. Assume towards a contradiction that there was a conflict-free k-coloring χ that avoids coloring v. Then, due to the copies of H being independent, disjoint and not connected to any other vertex, the restriction of χ to the vertices of each of the two copies must induce a conflict-free coloring on H. As $\chi_{CF}(H) = k$, this implies that χ uses k colors on each copy. Therefore, in the open neighborhood of v, there are at least two vertices colored with each color. This leads to a contradiction, because v cannot have a conflict-free neighbor.

For the second proposition, suppose there was a conflict-free coloring assigning the same color to u and v. Without loss of generality, let this color be 1. As every vertex of the two copies of J now sees two occurrences of color 1, color 1 can not be the color of the unique neighbor of any vertex of J, and any occurrence of color 1 on the vertices of J can be removed. Therefore, we can assume each of the two copies of J to be colored in a conflict-free manner using the colors $\{2, \dots, k\}$. Observe that, due to $\chi_{CF}(J) = k - 1$, each of these colors must be used at least once in each copy. This implies that both u and v see each color at least twice: The two copies of J enforce two occurrences of the colors $\{2, \dots, k\}$, and color 1 is assigned to both u and v, which are connected by an edge. This is a contradiction, and therefore, both u and v must be colored with distinct colors. ☐

Next, we give an inductive construction of graphs, G_k, with $\chi_{CF}(G_k) = k$. The proof of NP-hardness relies on this hierarchy.

1. The first graph G_1 of the hierarchy consists of a single isolated vertex. G_2 is a $K_{1,3}$ with one edge subdivided by another vertex, or, equivalently, a path of length 3 with a leaf vertex attached to one of the inner vertices.

2. Given G_k and G_{k-1}, G_{k+1} is constructed as follows for $k \geq 2$:

   - Take a complete graph $G = K_{k+1}$ on $k+1$ vertices.
   - To each vertex $v \in V(K_{k+1})$, attach two disjoint and independent copies of G_k, adding an edge from v to every vertex of both copies of G_k.
   - For each edge $e = vw \in E(K_{k+1})$, add two disjoint and independent copies of G_{k-1}, adding an edge from v and w to every vertex of both copies.

The number of vertices of the graphs G_k obtained by the above construction satisfies the recursive formula

$$|G_1| = 1, |G_2| = 5, |G_{k+1}| = (k+1) \cdot (2|G_k| + k|G_{k-1}| + 1),$$

which is in $\Omega\left(2^k\right)$ and $\mathcal{O}\left(2^{k \log k}\right)$. Figure 3.1 depicts the graph G_3, which in addition to being planar is a series-parallel graph.

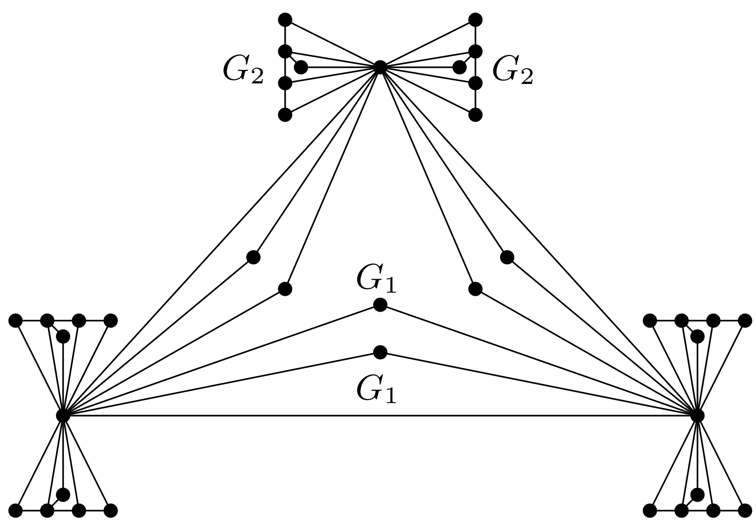


Figure 3.1: The graph G_3.

Lemma 3.3.3. *For G_k constructed in this manner, $\chi_{CF}(G_k) = k$.*

Proof. The proof uses induction over k. Application of Lemma 3.3.2 implies that all vertices of the K_{k+1} underlying G_{k+1} have to be colored using different colors. Therefore, $\chi_{CF}(G_{k+1}) \geq k+1$. By coloring all $k+1$ vertices of the underlying K_{k+1} with a different color, we obtain a conflict-free $(k+1)$-coloring of G_{k+1}, implying $\chi_{CF}(G_{k+1}) \leq k+1$. □

Lemma 3.3.4. *For $k \geq 2$, k-*Coloring $\preccurlyeq$ Conflict-Free k-Coloring. *Therefore, for $k \geq 3$,* Conflict-Free k-Coloring *is NP-complete.*

Proof. Given a graph G for which to decide proper k-colorability for a fixed k. We construct a graph G' that is conflict-free k-colorable iff G is k-colorable. G' is constructed from G by attaching two copies of G_k to each vertex $v \in V(G)$, by adding an edge from v to each vertex of the copies of G_k. For each edge $uv \in E(G)$, we attach two copies of G_{k-1} to both endpoints of uv by adding an edge from u and v to all vertices of both copies. As k is fixed, $|G_k|$ and $|G_{k-1}|$ are constant, implying that G' can be constructed in polynomial time.

A proper k-coloring of G induces a conflict-free k-coloring of G' by leaving all other vertices uncolored. On the other hand, by Lemma 3.3.2, a conflict-free k-coloring χ of G' colors all vertices $v \in V(G)$ and for every edge, the colors of both endpoints are distinct. Therefore, the restriction of χ to $V(G)$ is a proper k-coloring of G. □

3.3.2 A Sufficient Criterion for k-Colorability

In this section we present a sufficient criterion for conflict-free k-colorability together with an efficient heuristic that can be used to color graphs satisfying this criterion with k colors in a conflict-free manner. This heuristic is called *iterated elimination of distance-3-sets* and is detailed in Algorithm 1. The main idea of this heuristic is to iteratively compute maximal sets of vertices at pairwise (link) distance at least 3, coloring all vertices in one of these sets using one color, and then removing these vertices and their neighbors until all that remains is a collection of disconnected paths, which can then be colored using one color.

Algorithm 1 Iterated elimination of distance-3-sets

$i \leftarrow 1, \chi \leftarrow \emptyset$
Remove all isolated paths from G
while G is not empty **do**
    $D \leftarrow \emptyset$
    For each component of G, select some vertex v and add it to D
    while there is a vertex w at distance ≥ 3 from all vertices in D **do**
        Choose w at distance exactly 3 from some vertex in D
        $D \leftarrow D \cup \{w\}$
    $\forall u \in D : \chi(u) \leftarrow i$
    $i \leftarrow i + 1$
    Remove $N[D]$ from G
    Remove all isolated paths from G
Color all removed isolated paths using color i

Theorem 3.3.5. *Let G be a graph and $k \geq 1$. If G has neither K_{k+2} nor K_{k+3}^{-3} as a minor, G admits a conflict-free k-coloring that can be found in polynomial time using iterated elimination of distance-3 sets.*

Proof. For $k = 1$, a graph G with neither a K_3 nor a $K_4^{-3} = K_{1,3}$ minor consists of a collection of isolated paths. A path on $3n$ vertices can be colored with one color by coloring the middle vertex of every three vertices. This does not color the vertices at either end, so up to two vertices can be removed from the path to get colorings for paths on $3n - 1$ and $3n - 2$ vertices.

For $k \geq 2$, we use induction as follows: First, we color an inclusion-wise maximal subset $D \subseteq V$ of vertices at pairwise distance at least 3 to each other using color 1. This set D is chosen such that each vertex $v \in D$ is at distance exactly 3 from some $v' \in D$. Coloring D provides a conflict-free neighbor of color 1 to every vertex in $N[D]$. Therefore, the vertices in $N[D]$ are covered and can be removed from the graph. The remaining graph consists of vertices at distance 2 to some vertex in D; we call these vertices *unseen* in the remainder of the proof. We show that the remaining graph has no K_{k+1} and no K_{k+2}^{-3} as a minor. By induction, iterated elimination of distance 3 sets requires $k-1$ colors to color the remaining graph, and thus k colors suffice for G.

If the graph is disconnected, iterated elimination of distance 3 sets works on all components separately, so we can assume G to be connected. We claim that there is no set U of unseen vertices that is a cutset of G. Suppose there were such a cutset U and let H be any component of $G \setminus U$ not containing v, the first selected vertex during the construction of D. At least one vertex of H is colored: every vertex in U is at distance at least two from every colored vertex not in H, therefore, every vertex in H is at distance at least three from every colored vertex not in H. Consider the iteration where the first vertex w of H is added to the set of colored vertices D. At this point, w is at distance exactly 3 from some colored vertex not in H. However, this implies w is adjacent to some vertex from U, contradicting the fact that all vertices in U are unseen.

Now, suppose for the sake of contradiction that the set W of unseen vertices contains a K_{k+1} or K_{k+2}^{-3} minor. W is not the whole graph, because at least one vertex is colored, so there must be a vertex v not in the K_{k+1} or K_{k+2}^{-3} minor. For every vertex $w \in W$, there is a path from v to w that intersects W only at w. Otherwise, $W \setminus \{w\}$ would be a cutset separating v from w. So, if the graph induced by W had a K_{k+1} or K_{k+2}^{-3} minor, we could contract $G \setminus W$ to a single vertex, which would be adjacent to all vertices in W, yielding a K_{k+2} or K_{k+3}^{-3} minor of G, which does not exist. □

Observe that G_{k+1} contains a K_{k+3}^{-3} as a minor, but not a K_{k+2}, proving that just excluding K_{k+2} as a minor does not suffice to guarantee k-colorability. Moreover, note that K_{k+1} is a minor of K_{k+2} and K_{k+3}^{-3}.

This yields the following corollary, which is the conflict-free variant of the Hadwiger Conjecture.

Corollary 3.3.6. *All graphs that do not have K_{k+1} as a minor are conflict-free k-colorable.*

3.3.3 Conflict-Free Domination Number

In this section we consider the problem of minimizing the number of colored vertices in a conflict-free k-coloring for a fixed k, which is equivalent to computing γ_{CF}^k. We call the corresponding decision problem k-Conflict-Free Dominating Set. We show that approximating the conflict-free domination number in general graphs is hard for any fixed k. In § 3.5 we discuss the k-Conflict-Free Dominating Set problem for planar graphs.

Theorem 3.3.7. *Unless P = NP, for any $k \geq 3$, there is no polynomial-time approximation algorithm for $\gamma_{CF}^k(G)$ with constant approximation factor.*

Proof. We use a reduction from proper k-Coloring for the proof. Assume towards a contradiction that there was a polynomial-time approximation algorithm for $\gamma_{CF}^k(G)$ with approximation factor $c \geq 1$. Let G be a graph on n vertices for which we want to decide k-colorability. For each vertex v of G, add $M := (n+1)(c+1)$ vertices u_v to G and connect them to v. For each edge vw of G, add M vertices u_{vw} to G and connect them to both v and w. Let G' be the resulting graph. Clearly, the size of G' is polynomial in the size of G. Additionally, G' is planar if G is, and G' has a conflict-free k-coloring of size n iff G is properly k-colorable: Any proper k-coloring of G is a conflict-free k-coloring of G', as every vertex added to G is either adjacent to two distinctly colored vertices of G, or adjacent to just one vertex of G. Conversely, let χ be a conflict-free coloring of G', coloring just n vertices. If χ did not assign a color to some vertex v of G, it would have to color all $M \geq n+1$ neighbors of v. If χ assigned the same color to any pair v, w of vertices adjacent in G, it would have to color all M vertices adjacent only to v and w. Therefore, χ is a proper coloring of G. Running a c-approximation algorithm $\mathcal{A}$ for γ_{CF}^k on G' results in an approximate value $\mathcal{A}(G') \leq c \cdot \gamma_{CF}^k(G')$. We have $\mathcal{A}(G') \leq c \cdot n < M$ if G is k-colorable, and $\mathcal{A}(G') \geq M$ if G is not; thus we could decide proper k-colorability in polynomial time. □

3.4 Closed Neighborhoods: Planar Conflict-Free Coloring

This section deals with the Planar Conflict-Free k-Coloring problem which consists of deciding conflict-free k-colorability for fixed k on planar graphs. Due to the 4-color theorem, we immediately know that every planar graph is conflict-free 4-colorable. This naturally leads to the question of whether there are planar graphs requiring 4 colors or whether fewer colors might already suffice for a conflict-free coloring, which we address in the following two sections.

3.4.1 Complexity

For $k \in \{1, 2\}$ colors, we show that the problem of deciding conflict-free k-colorability on planar graphs is NP-complete. This implies that 2 colors are not sufficient.

Theorem 3.4.1. *Deciding planar conflict-free 1-colorability is NP-complete.*

Proof. Membership in NP is obvious. The proof of NP-hardness is done by reduction from the problem Positive Planar 1-in-3-SAT. From a positive planar 3-CNF formula ϕ with clauses $C = \{c_1, \ldots, c_l\}$ and variables $X = \{x_1, \ldots, x_n\}$ we construct in polynomial time a graph $G_1(\phi)$ such that ϕ is 1-in-3-satisfiable iff $G_1(\phi)$ admits a conflict-free 1-coloring.

First, find and fix a planar embedding d of $G(\phi)$. $G_1(\phi)$ is constructed from $G(\phi)$ and d as follows: For every variable x_i, there is a cycle $Z_i = (z_{i,1}, \ldots, z_{i,12})$ of length 12. The vertices $z_{i,1}, z_{i,4}, z_{i,7}, z_{i,10}$ are referred to as *true* vertices of Z_i, all other vertices are *false* vertices. Moreover, vertices $z_{i,1}, z_{i,2}, z_{i,3}$ are called *upper* vertices of Z_i, and vertices $z_{i,7}, z_{i,8}, z_{i,9}$ are called *lower* vertices of Z_i. Additionally, vertices $z_{i,4}, z_{i,5}, z_{i,6}$ are called *right* vertices of Z_i and $z_{i,10}, z_{i,11}, z_{i,12}$ are called *left* vertices of Z_i.

For each clause c_j, there is a cycle $(c_{j,1}, \ldots, c_{j,4})$ of length 4 in $G_1(\phi)$. To each variable x_i for $i \in \{2, \ldots, n-1\}$, we associate two disjoint sequences $U_i = \left(u_j\right)_{j=1}^{|U_i|}$ and $L_i = \left(l_j\right)_{j=1}^{|L_i|}$ of clauses x_i appears in. The sequences are constructed using a clockwise (with respect to d) enumeration of the edges of x_i in $G(\phi)$, starting with $x_{i-1}x_i$. Let $(x_{i-1}x_i, x_ic_{j_1}, \ldots, x_ic_{j_\lambda}, x_ix_{i+1}, x_ic_{j_{\lambda+1}}, \ldots, x_ic_{j_\mu})$ be the sequence of edges encountered in this manner and set $U_i := (c_{j_1}, \ldots, c_{j_\lambda})$ and $L_i := (c_{j_{\lambda+1}}, \ldots, c_{j_\mu})$. For $i \in \{1, n\}$, L_i is empty and U_i contains all clauses x_i appears in, again in clockwise order. In $G_1(\phi)$, the clauses and variables are connected such that for each clause c_j that x_i occurs in, either the upper or the lower *true* vertex of x_i is adjacent to $c_{j,1}$. More precisely, for variable x_i, if $c_j = u_m$, we add the edge $c_{j,1}z_{i,1}$ to connect the upper true vertex to the clause. If $c_j = l_m$, we add $c_{j,1}z_{i,7}$ to connect the lower true vertex to the clause. Because the order of edges around each vertex is preserved by the construction, the graph $G_1(\phi)$ obtained in this way can be embedded in the plane by a suitable adaptation of d. See Figure 3.2 for an example of the construction.

Now we prove that $G_1(\phi)$ is conflict-free 1-colorable iff ϕ is 1-in-3-satisfiable. Regarding necessity, a valid truth assignment $b : X \to \mathbb{B}$ yields a valid conflict-free coloring by coloring the vertex $c_{j,3}$ of every clause, coloring all true vertices of variables with $b(x_i) = 1$ and coloring the

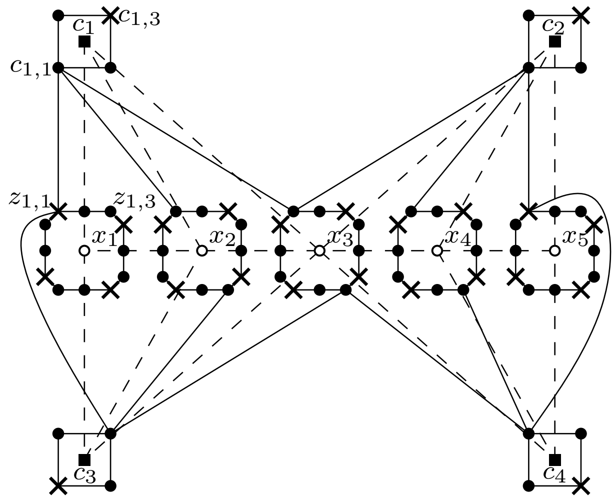


Figure 3.2: A formula graph $G(\phi)$ (dashed) and the corresponding $G_1(\phi)$ (solid).

false vertices $z_{i,3}, z_{i,6}, z_{i,9}, z_{i,12}$ of all other variables. Thus, in every cycle Z_i, every third vertex is colored, providing a conflict-free neighbor to every vertex of Z_i. Moreover, in each clause, by virtue of $c_{j,3}$ being colored, vertices $c_{j,2}, c_{j,3}, c_{j,4}$ have a conflict-free neighbor. Because b is a valid truth assignment, for each clause, the vertex $c_{j,1}$ is adjacent to exactly one colored true vertex. Therefore, the coloring constructed in this way is conflict-free.

Regarding sufficiency, we first argue that the vertices $c_{j,1}, c_{j,2}, c_{j,4}$ can never be colored: If $c_{j,1}$ receives a color, then $c_{j,3}$ still enforces that one of $c_{j,2}, c_{j,3}, c_{j,4}$ is colored, leading to a contradiction in either case. If $c_{j,2}$ receives a color, then $c_{j,4}$ cannot have a conflict-free neighbor and vice versa. Therefore, no clause vertex can be the conflict-free neighbor of any vertex of Z_i. Thus, the conflict-free neighbor of *every* vertex of Z_i must itself be a vertex of Z_i. Moreover, the conflict-free neighbor of every vertex $c_{j,1}$ must be a true vertex. Thus, there are exactly three ways to color each cycle Z_i: either by coloring the true vertices (one possibility), or by coloring every other false vertex (two possibilities). A valid conflict-free 1-coloring of $G_1(\phi)$ satisfies the property that for each clause c_j, exactly one of the true vertices adjacent to $c_{j,1}$ is colored. Hence, a valid conflict-free 1-coloring of $G_1(\phi)$ induces a valid truth assignment b by setting $b(x_i) = 1$ iff all true vertices of x_i are colored. $\square$

Theorem 3.4.2. *It is NP-complete to decide whether a planar graph admits a conflict-free 2-coloring.*

The proof requires the gadget $G_{\leq 1}$ depicted in Figure 3.3. $G_{\leq 1}$ consists of three vertices v, w_1, w_2 forming a triangle. Each edge ux of the triangle has two corresponding vertices y^1_{ux}, y^2_{ux}, each connected to u and x. Furthermore, both w_1 and w_2 are attached to two copies of a cycle on 4 vertices, where every vertex of both cycles is adjacent to the corresponding w_i. $G_{\leq 1}$ can be used to enforce that the vertices connected to its central vertex v are colored using at most one distinct color:

Lemma 3.4.3. *Let $G = (V, E)$ be any graph, let $v \in V$ and let G' be the graph resulting from adding a copy of $G_{\leq 1}$ to G by identifying v in G with v in $G_{\leq 1}$. Then (1) G' is planar if G is, and (2) every conflict-free 2-coloring of G' leaves v uncolored and uses at most one color on $N_G[v]$.*

Proof. The planarity of G' follows from the planarity of G by the observation that $G_{\leq 1}$ is planar and can be embedded in any face incident to v in a planar embedding of G. Now consider a conflict-free 2-coloring χ of G'. χ must color both w_1 and w_2. Otherwise, χ restricted to each of the two 4-cycles adjacent to w_i must be a valid conflict-free 2-coloring. However, as C_4 requires at least 2 different colors, w_i then sees two occurrences of both colors, and thus cannot have a conflict-free neighbor anymore. Furthermore, $\chi(w_1) \neq \chi(w_2)$, as otherwise, $y^1_{w_1w_2}$ and $y^2_{w_1w_2}$ must both be colored with the other color; but then, w_1 and w_2 again see two occurrences of both colors. By an analogous argument, χ must not color v. Moreover, χ cannot use more than one color on $N_G[v]$, because v already sees one occurrence of each color, so adding another occurrence of both colors would yield a conflict at v. □

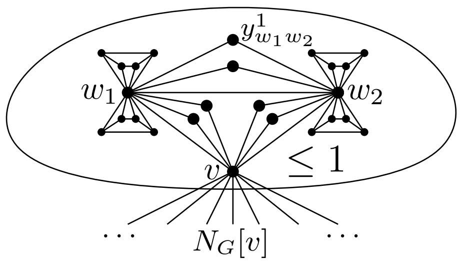


Figure 3.3: Gadget $G_{\leq 1}$

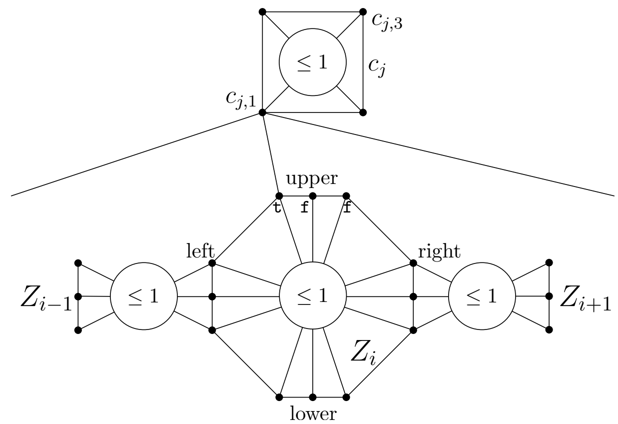


Figure 3.4: Clause and variable gadget for $k = 2$

Proof of Theorem 3.4.2. NP-hardness is proven by constructing, in polynomial time, a planar graph $G_2(\phi)$ from the graph $G_1(\phi)$ used in the hardness proof for $k = 1$, such that $G_2(\phi)$ is conflict-free 2-colorable iff $G_1(\phi)$ is conflict-free 1-colorable.

The construction is carried out by adding a gadget $G_{\leq 1}$ to every variable cycle Z_i of $G_1(\phi)$, to every clause cycle and between the right and left vertices of two adjacent variable cycles Z_i and Z_{i+1}. This is depicted in Figure 3.4. More precisely, for every cycle Z_i, we add one copy of gadget $G_{\leq 1}$, and connect its central vertex v to all vertices of the cycle. In a planar embedding of $G_2(\phi)$, these gadgets can be embedded within the face defined by the cycles Z_i and thus do not harm planarity. By Lemma 3.4.3, this enforces that on every cycle, only one color can be used. Moreover, for every edge x_ix_{i+1} in $G(\phi)$, we add one copy of $G_{\leq 1}$ that we connect to the right vertices of x_i and the left vertices of x_{i+1}. This preserves planarity because these gadgets and the added edges can be embedded in the face crossed by x_ix_{i+1} in some fixed embedding d of $G(\phi)$. As one of the right vertices of x_i and one of the left vertices of x_{i+1} must be colored, this enforces that the same single color must be used to color all cycles Z_i. Finally, we add a copy of $G_{\leq 1}$ to every clause c_j and connect it to $c_{j,1}, \ldots, c_{j,4}$. Again, this preserves planarity because the gadget may be embedded in the face defined by $(c_{j,1}, \ldots, c_{j,4})$.

We now argue that $G_2(\phi)$ is conflict-free 2-colorable iff $G_1(\phi)$ is conflict-free 1-colorable. A 1-coloring of $G_1(\phi)$ induces a 2-coloring of $G_2(\phi)$ by copying the color assignment and coloring the internal vertices of the added gadgets as described in the proof of Lemma 3.4.3. Now, let $G_2(\phi)$ be conflict-free 2-colorable and fix a valid 2-coloring χ. In each clause, χ must color $c_{j,3}$ and neither of $c_{j,1}, c_{j,2}$ nor $c_{j,4}$ can be colored. Therefore, no clause vertex can be the conflict-free neighbor of any vertex of Z_i. Thus, the conflict-free neighbor of *every* vertex of Z_i must itself be a vertex of Z_i. Moreover, the conflict-free neighbor of every vertex $c_{j,1}$ must be a true vertex. As there is only one color available to color all cycle vertices of all variables, the restriction of χ to the vertices of $G_1(\phi)$ yields a valid 1-coloring except for the fact that some $c_{j,3}$ might use a different color than the one used for the variables. However, this can be fixed by simply replacing all occurring colors with one single color. Hence, $G_2(\phi)$ is conflict-free 2-colorable iff $G_1(\phi)$ is conflict-free 1-colorable. $\square$

3.4.2 Sufficient Number of Colors

As shown above, it is NP-complete to decide whether a planar graph has a conflict-free k-coloring for $k \in \{1, 2\}$. On the positive side, we can establish the following result, which follows from the more general results discussed in § 3.3.2.

Corollary 3.4.4 (of Theorem 3.3.5)**.** *Every outerplanar graph is conflict-free* 2*-colorable and every planar graph is conflict-free* 3*-colorable. Moreover, such colorings can be computed in polynomial time.*

Outerplanar graphs are not the only interesting graph class for which one might suspect two colors to be sufficient. Two other interesting subclasses of planar graphs are series-parallel graphs and pseudomaximal planar graphs. However, each of these classes contains graphs that do not admit a conflict-free 2-coloring: The graph G_3 as defined in § 3.3 is an example of a series-parallel graph requiring three colors. Figure 3.5 depicts a maximal outerplanar graph O_9 satisfying $\chi_{CF}(O_9) = 2$. This graph can be used to obtain a pseudomaximal planar graph M with $\chi_{CF}(M) = 3$ by adding two copies of O_9 to the neighborhood of every vertex of a triangle, similar to the construction of G_3, and adding gadgets on the inside of the triangle as depicted in Figure 3.6.

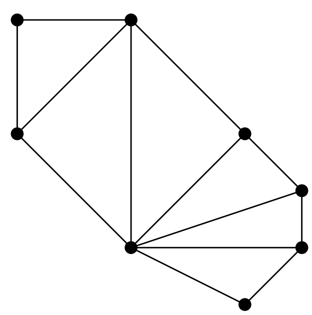


Figure 3.5: The maximal outerplanar graph O_9.

Furthermore, observe that Theorem 3.4.4 does not hold if every vertex must be colored. In this case, there are outerplanar graphs requiring 3 colors for a conflict-free coloring. One can obtain an example of such a graph by adding a chord to a cycle of length 5.

3.5 Closed Neighborhoods: Planar Conflict-Free Domination

In this section we consider the decision problem k-Conflict-Free Dominating Set for planar graphs. In § 3.5.1, we deal with the cases when $k \in \{1, 2\}$ for planar and outerplanar graphs, and we give a polynomial time algorithm to compute an optimal conflict-free coloring of outerplanar graphs with $k \in \{1, 2\}$ colors. Section 3.5.2 discusses the problem for $k \geq 3$.

3.5.1 At Most Two Colors

We start by pointing out that, for every conflict-free 1-colorable graph G, $\gamma^1_{CF}(G) = \gamma(G)$ holds. Moreover, Corollary 3.5.1 discusses the complexity of k-Conflict-Free Dominating Set and

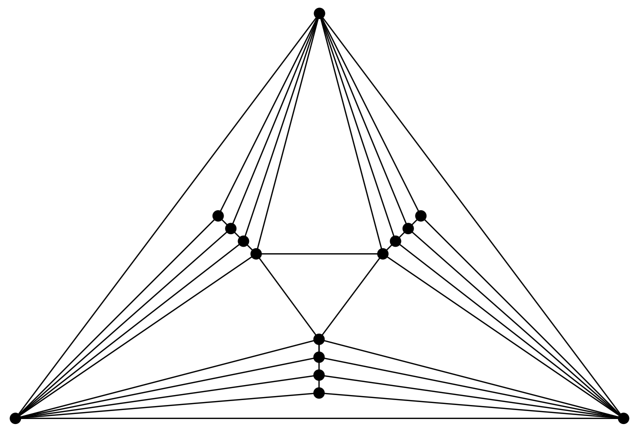


Figure 3.6: The pseudomaximal planar graph M, without the O_9 gadgets.

Theorem 3.5.2 states positive results for outerplanar graphs.

Corollary 3.5.1 (of Theorems 3.4.1 and 3.4.2)**.**
k-Conflict-Free Dominating Set *is NP-complete for* $k \in \{1, 2\}$ *for planar graphs.*

Theorem 3.5.2. *Let* $k \in \{1, 2\}$ *and let* G *be an outerplanar graph. We can decide in polynomial time whether* $\chi_{CF}(G) \le k$*. Moreover, we can compute a conflict-free* k*-coloring of* G *that minimizes the number of colored vertices in* $\mathcal{O}(n^{4k+1})$ *time.*

The proof of Theorem 3.5.2 relies on a polynomial-time algorithm that computes a k-coloring of the input outerplanar graph G if and only if such a coloring exists (which thus solves the decision problem). In the following, we describe our algorithm.

Let $G = (V, E)$ be an outerplanar graph. Let $\chi : V' \subseteq V(G) \to \{0, 1, \dots, k\}$ be a partial coloring of the vertices of G and let $v \in V$. Observe that χ defined like this differs from the definition given earlier in the introduction. We call a pair $\mathcal{C}_v = [\chi(v), S_v]$ a *configuration* of v, where $\chi(v) \in \{0, 1, \dots, k\}$ denotes the color of v. If $\chi(v) = 0$, we regard v as uncolored. The set $S_v \subseteq N[v]$ is the set of conflict-free neighbors of v, along with their colors. That is, *every* $w \in S_v$ is a *conflict-free neighbor* of v under χ. For $e = uv \in E$ we call a pair $\mathcal{C}_e = [\mathcal{C}_u, \mathcal{C}_v]$ a *configuration* of e. By $\mathcal{C}_e^w$ we denote the configuration of an endpoint $w \in \{u, v\}$ of e. Observe that if χ was conflict-free, then $S_v \neq \emptyset$, and $\mathcal{C}_u$ and $\mathcal{C}_v$ do not *conflict* with each other. For the latter property we say that $\mathcal{C}_u$ and $\mathcal{C}_v$ are *compatible* and we denote this by $\mathcal{C}_u \leftrightarrow \mathcal{C}_v$. If $\mathcal{C}_e^v = \mathcal{C}_{e'}^v$ for a pair $e = uv, e' = vw$ of incident edges, then we say $\mathcal{C}_{e'}$ *is compatible with* $\mathcal{C}_e$. The following observation is straightforward:

Observation 3.5.3. Let G be an outerplanar graph. Let $\mathcal{C} = \{\mathcal{C}_1, \dots, \mathcal{C}_{|E|}\}$ be a set of configurations over the edges of G using k colors. If for *every* pair $e = uv$, $e' = vw$ of incident edges, $\mathcal{C}_u \leftrightarrow \mathcal{C}_v$ and $\mathcal{C}_v \leftrightarrow \mathcal{C}_w$ holds *and* $\mathcal{C}_{e'}$ is compatible with $\mathcal{C}_e$, then a conflict-free k-coloring can be obtained from $\mathcal{C}$.

Now let $v \in V(G)$. Observe that the number of different configurations $\mathcal{C}_v = [\chi(v), S_v]$ is upper-bounded by $O(n^k)$, as there cannot be more than $\binom{|N[v]|}{k} \cdot k!$ different sets S_v. Thus the following observation is straightforward.

Observation 3.5.4. Let $G = (V, E)$ be an outerplanar graph and let $e = uv \in E$. The number of different configurations $\mathcal{C}_e = [\mathcal{C}_u, \mathcal{C}_v]$ is upper-bounded by $O(n^{2k})$.

We can now describe our algorithm, which is based on non-serial dynamic programming. For the sake of simplicity, let us assume that the weak dual $G^* = (V^*, E^*)$ of the outerplanar graph G is connected. This means that G^* is a tree. It is well-know that, in general, the weak dual graph of an outerplanar graph G is a forest [Sys79]. We discuss later how to convert this forest into a tree as long as G is connected.

Let us root G^* at an arbitrary dual vertex $r \in V^*$. Thus, each dual vertex has a unique parent vertex on the path from the vertex to r. For an edge $e = vw \in E^*$, where v is the parent of w, we consider the subtree $\mathcal{T}_e$ rooted at w. Let G_e be the primal subgraph of G whose dual graph is $\mathcal{T}_e$.

We define a *window* b as the edge or vertex in the primal graph G separating two faces f_1, f_2. Observe that b corresponds to an edge e in the dual graph G^*. If f_1^* and f_2^* are two (dual) vertices in the dual graph, then the corresponding faces f_1 and f_2 only have b in common, see Figure 3.7. Assume that f_2 has been conflict-free k-colored. Then, to color f_1 in a conflict-free manner, we would need all the possible configurations of the window b allowed by the conflict-free coloring of the face f_2. The algorithm performs dynamic programming starting by computing *all* possible configurations of the leaves of G^* and propagating them towards the root in a compatible manner (conflict-freely).

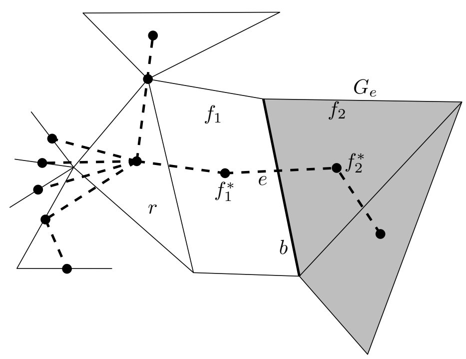


Figure 3.7: Graph construction of faces, windows, and the corresponding dual (sub)graphs. The shaded are corresponds to already processed faces of G (the past). The face f_1 is the face to be processed next (the present). Edge b is the window between f_1 and f_2. The rest of the graph corresponds to faces to be processed in the future.

Let f be a face of G and f^* be the corresponding dual vertex in G^*. Let b be the window of f and let $e = b^*$ be the dual edge of b connecting f^* to its parent $p = p(f^*)$. For any configuration $\mathcal{C}_b$, we compute the score $\mathcal{S}(\mathcal{C}_b)$, which is the number of colored vertices corresponding to $\mathcal{C}_b$ in the conflict-free k-coloring of the subgraph G_e. We store the pairs $\big(\mathcal{C}_w, \mathcal{S}(\mathcal{C}_w)\big)$ which are then combined with the other children of p to compute the compatible configurations of p. Given a window w of a face f_l, the algorithm GenerateScore computes $\mathcal{S}(\mathcal{C}_w)$ for a given configuration $\mathcal{C}_w$. Let f_l consist of the edges $\langle e_1 = (u_1, v_1), \ldots, e_\ell = (u_\ell, v_\ell)\rangle$ where, without loss of generality, $w = e_1$ if w is an edge. Otherwise $w = u_1$ if w is a vertex. Also, let $L(e_i)$ be the set of all possible configurations of the edge e_i. By $\mathcal{C}_{u_1}^S$ we denote the number of conflict-free neighbors of u_1 given the configuration $\mathcal{C}_{u_1}$, i.e., if $\mathcal{C}_{u_1} = (\chi(u_1), S_{u_1})$, then $\mathcal{C}_{u_1}^S = |S_{u_1}|$. The algorithm populates a family $\{P_i\}$ of sets containing pairs of compatible configurations and their scores. In the algorithm GenerateScore, $\delta(\mathcal{C}_{e_i}, \mathcal{C}_{e_{i-1}})$ is the number of newly-colored vertices resulting from combining the two compatible configurations C_{e_i} and $C_{e_{i-1}}$.

Lemma 3.5.5. *For a fixed $k \geq 1$, we can compute the scores $\mathcal{S}(C_b)$ for all configurations C_b of all* windows *b in $\mathcal{O}(n^{4k+1})$ time.*

Algorithm 2 Processing a configuration of a window

function GenerateScore($\mathcal{C}_{e_1}$, $f = \langle e_1 = (u_1, v_1), \ldots, e_\ell = (u_\ell, v_\ell)\rangle$)
  $P_1 \leftarrow \{(\mathcal{C}_{e_1}, \mathcal{C}^S_{u_1})\}$
  for $i = 2, \ldots, \ell$ **do**
    $P_i \leftarrow \emptyset$
    for $(\mathcal{C}_{e_{i-1}}, h) \in P_{i-1}$ **do**
      for $\mathcal{C}_{e_i} \in L(e_i)$ **do**
        if $\mathcal{C}_{e_{i-1}}$ is compatible with $\mathcal{C}_{e_i}$ **then**
          $P_i \leftarrow P_i \cup \{(\mathcal{C}_{e_i}, h + \delta(\mathcal{C}_{e_i}, \mathcal{C}_{e_{i-1}}))\}$
  $\mathcal{S}(\mathcal{C}_{e_1}) \leftarrow \infty$
  for $(\mathcal{C}_{e_\ell}, h) \in P_\ell$ **do**
    if $\mathcal{C}_{e_\ell}$ is compatible with $\mathcal{C}_{e_1}$ **then**
      $\mathcal{S}(\mathcal{C}_{e_1}) \leftarrow \min\{\mathcal{S}(\mathcal{C}_{e_1}), h\}$

Proof. We process the dual graph G^* starting from the leaves. Let b be the window between the two faces f_1 and f_2. The window corresponds to an edge between the a dual vertex and its parent in the dual graph. Let $f_1 = \langle e_1 = (u_1, v_1), \ldots, e_\ell = (u_\ell, v_\ell)\rangle$ such that $e_1 = b$, $v_\ell = u_1$, and $v_i = u_{i+1}$ for $i \in \{1, \ldots, \ell - 1\}$. We compute $\mathcal{S}(C_b)$ by applying Algorithm 2. Inductively, we can compute the score for all configurations of all windows going up in the dual graph in this manner.

For each window there are at most $\mathcal{O}(n^{2k})$ configurations. This implies that for each pair of edges, there are at most $\mathcal{O}(n^{4k})$ pairs of configurations. As Algorithm 2 considers $\mathcal{O}(n)$ pairs of edges overall, we obtain a running time of $\mathcal{O}(n^{4k+1})$ for the algorithm. □

Proof of Theorem 3.5.2. By applying the approach of Algorithm 2 we can compute the scores of all windows of the graph G. At the root node we have a set of configuration for each window that results in the minimum number of colored vertices in the whole graph. Such a set can be obtained by backtracking. Combining this with Observation 3.5.3, we get a conflict-free coloring with a minimal number of colored vertices for the graph G, if and only if $\chi_{CF}(G) \leq k$. □

What remains to be discussed is how we treat the case in which G^* is not a tree but a forest (assuming G is connected). The dual G^* becomes disconnected if G has cut edges or cut vertices. In such a case, we use the following construction depicted in Figure 3.8 to connect the components of G^* to obtain a tree.

(1) For a cut vertex v, let $\langle f_1, \ldots, f_t\rangle$ be the t faces containing v. Let $\langle f_1^*, \ldots, f_t^*\rangle$ be the corresponding vertices in G^*. We make one of f_i^* a parent to all the others by adding an edge between them. Note that this does not create a cycle because G is outerplanar.

(2) If we have a cut edge, we consider the cut edge as a face. In this way, for a cut edge, we have a vertex in the dual graph.

3.5.2 Approximability for Three or More Colors

In § 3.4.2 we stated that every planar graph is conflict-free 3-colorable. In this section we deal with conflict-free 3-colorings of planar graphs that, additionally, minimize the number of colored vertices.

Theorem 3.5.6. *Let $k \geq 3$ and let G be a planar graph. The following holds:*

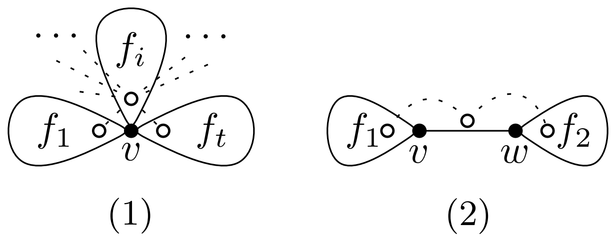


Figure 3.8: Two cases leading to a forest: (1) a cut vertex, (2) a cut edge.

(1) Unless P = NP, there is no polynomial-time approximation algorithm providing a constant-factor approximation of $\gamma^3_{CF}(G)$ for planar graphs. 3-Conflict-Free Dominating Set *is NP-complete for planar graphs.*

(2) For $k \geq 4$, k-Conflict-Free Dominating Set *is NP-complete. Also, $\gamma^k_{CF}(G) = \gamma(G)$, and the problem is fixed-parameter tractable with parameter $\gamma^k_{CF}(G)$. Furthermore, there is a PTAS for $\gamma^k_{CF}(G)$.*

(3) If G is outerplanar, then $\gamma^k_{CF}(G) = \gamma(G)$ and there is a linear-time algorithm to compute $\gamma^k_{CF}(G)$.

The proof of Theorem 3.5.6 is based on the following polynomial-time algorithm, which transforms a dominating set D of a planar graph G into a conflict-free k-coloring of G, coloring only the vertices of D: Let D be a dominating set of a planar graph G. Every vertex $v \in V(G) \setminus D$ is adjacent to at least one vertex in D. Pick any such vertex $u \in D$ and contract the edge $uv \in E(G)$ towards u. Repeat this until only the vertices from D remain. Because G is planar, the graph $G' = (D, E')$ obtained in this way is planar, as G' is a minor of G. By the 4-coloring theorem, we can compute a proper 4-coloring of G'.

Lemma 3.5.7. *The 4-coloring generated by this procedure induces a conflict-free 4-coloring of G.*

Proof. Every vertex $u \in D$ is a conflict-free neighbor to itself as its color does not appear in $N_G(u)$. Let $v \in V(G) \setminus D$ be some uncolored vertex, and let $u \in D$ be the vertex that v was contracted towards by the algorithm. In G', this contraction made u adjacent to all other vertices in $N_G(v) \cap D$, which guarantees that the color of u is unique in $N_G(v) \cap D$. As $V(G) \setminus D$ remains uncolored, the color of u is thus unique in $N_G[v]$. □

Proof of Theorem 3.5.6. Proposition (1) follows from Theorem 3.3.7 of § 3.3.3: The reduction used there preserves planarity and proper planar 3-coloring is NP-complete. For (2), $\gamma^k_{CF}(G) = \gamma(G)$ implies NP-hardness in planar graphs because planar minimum dominating set is NP-hard. Moreover, the coloring algorithm lets us apply any approximation scheme for planar dominating set to conflict-free k-coloring. We obtain a PTAS for the conflict-free domination number by applying our coloring algorithm to the dominating set produced by the PTAS of Baker and Hill [BH94]. Additionally, Alber et al. [AFN04] proved that planar dominating set is FPT with parameter $\gamma(G)$, implying that computing the planar conflict-free domination number for $k \geq 4$ is FPT with parameter $\gamma^k_{CF}(G)$. For (3), the class of outerplanar graphs is properly 3-colorable in linear time and closed under taking minors. Kikuno et al. [KYK83] present a linear time algorithm for finding a minimum dominating set in a series-parallel graph, which includes outerplanar graphs. The result follows by combining this linear time algorithm with the coloring algorithm mentioned above, but using just three colors instead of four. □

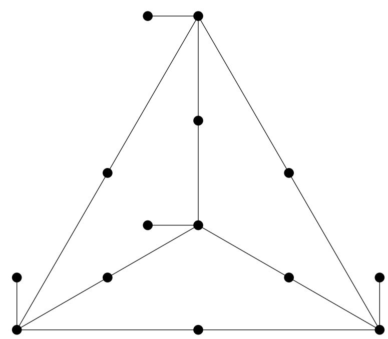


Figure 3.9: The graph G' resulting from applying the reduction to K_4. This bipartite planar graph has $\chi_O(G') = 4$.

3.6 Open Neighborhoods: Planar Conflict-Free Coloring

In this section we discuss the problem of conflict-free coloring with open neighborhoods. Recall that an open-neighborhood conflict-free coloring is a coloring of some vertices of a graph G such that every vertex has a conflict-free neighbor in its open neighborhood $N(v)$. In some settings, this problem is a natural alternative to the closed-neighborhood variant; for instance, when guiding a robot from one location to another, a uniquely colored beacon at the robot's current position may be insufficient.

Note that isolated vertices are problematic for this variant of conflict-free coloring; therefore, in the following, we assume that G does not contain isolated vertices. Moreover, we observe the following.

Observation 3.6.1. Let G be a graph, $v, w \in V(G)$, and $\deg(v) = 1$, $\deg(w) = 2$. Then, for any number k of colors, in any conflict-free k-coloring, the unique neighbor of v must be colored. Moreover, the two neighbors of w cannot have the same color.

This leads to a straightforward reduction from proper coloring to conflict-free coloring. Given a graph G, adding an otherwise isolated neighbor to each original vertex and placing a vertex with degree 2 on every original edge yields a graph G' with $\chi_O(G') = \chi_P(G)$. See Figure 3.9 for an example of this reduction. The resulting graph G' is bipartite. Furthermore, the reduction preserves planarity, implying that bipartite planar graphs may require at least 4 colors in a conflict-free coloring. Moreover, even though this reduction does not necessarily preserve outerplanarity, applying it to a K_3 yields an outerplanar graph that requires at least 3 colors. For *bipartite* planar and outerplanar graphs, these bounds are tight.

Corollary 3.6.2. *It is NP-complete to decide whether a bipartite planar graph G is open-neighborhood conflict-free* 3*-colorable.*

Theorem 3.6.3. *Every bipartite planar graph is open-neighborhood conflict-free* 4*-colorable. For bipartite outerplanar graphs, three colors are sufficient.*

Proof. Let $G = (V_1 \cup V_2, E)$ be a bipartite planar graph with partitions V_1 and V_2; the proof proceeds analogously for outerplanar graphs. We construct two minors G_1 and G_2 of G, to each of which we apply the planar four-color theorem. We build G_1 by merging all vertices $v \in V_2$ into an arbitrarily chosen neighbor $v_1(v) \in V_1$. Because G is bipartite and does not contain isolated vertices, it is possible to continue this process until no vertices from V_2 remain. G_2 is constructed analogously, merging all vertices $v \in V_1$ into an arbitrarily chosen neighbor $v_2(v) \in V_2$. Each of the two resulting graphs G_i contains exactly the vertices from V_i. Moreover, as a minor of G, G_i is planar and therefore has a proper coloring with four colors. We assign the colors from this coloring to the vertices in V_i.

It remains to show that this induces an open-neighborhood conflict-free coloring of G. Let v be a vertex of G. W.l.o.g., assume $v \in V_1$. During the construction of G_2, v was merged into its neighbor $v_2(v) \in V_2$. Therefore in G_2, $v_2(v)$ is adjacent to all other neighbors of v in G. Because all neighbors of v are in V_2, this implies that the color of $v_2(v)$ is unique in $N_G(v)$, and $v_2(v)$ is a conflict-free neighbor of v. □

On the other hand, for non-bipartite planar graphs, we can show the following upper bound on the number of colors.

Theorem 3.6.4. *Every planar graph has an open-neighborhood conflict-free coloring using at most eight colors.*

Proof. Let $G = (V, E)$ be a planar graph. Analogous to the proof of Theorem 3.6.3 we proceed by producing two minors G_1 and G_2 of G, to each of which we apply the planar four-color theorem. However, without the assumption of bipartiteness, we cannot use the same set of four colors for G_1 and G_2, leading to a conflict-free coloring with eight colors.

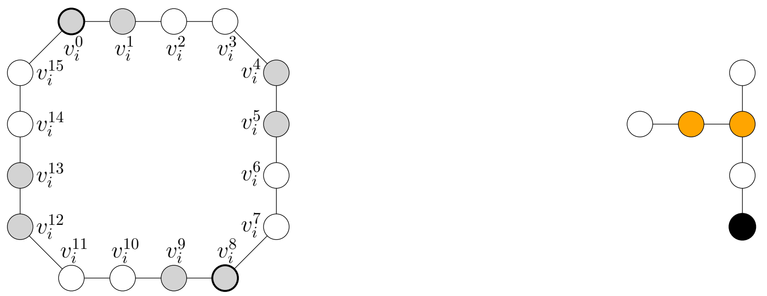
We start by constructing an independent dominating set V_1 of G. Let $V_2 := V \setminus V_1$. We construct the minor G_i of G by contracting each vertex $v \in V_{3-i}$ into an arbitrarily chosen neighbor $v_i(v) \in V_i$. Then we apply the planar four-color theorem to G_1 and G_2 with colors $\{1, 2, 3, 4\}$ and $\{5, 6, 7, 8\}$. To build a conflict-free coloring of G, we assign to each $v \in V_i$ its color in the proper coloring of G_i. This results in a conflict-free coloring because $v_{3-i}(v)$ is a conflict-free neighbor of v. □

Similar to the situation for closed neighborhoods, open neighborhood conflict-free coloring is hard even for $k = 1$ and $k = 2$. For closed neighborhoods, a conflict-free 1-coloring corresponds to a dominating set consisting of vertices at pairwise distance at least 3. For open neighborhoods, a conflict-free 1-coloring corresponds to a matching whose vertices form a dominating set and are at pairwise distance at least 3 (except for those adjacent in the matching).

Theorem 3.6.5. *It is NP-complete to decide whether a bipartite planar graph G is open-neighborhood conflict-free* 1*-colorable.*

Proof. We prove hardness using a reduction from Positive Planar 1-in-3-SAT. In a manner similar to the proof of Theorem 3.4.1, from a positive planar 3-CNF formula ϕ with clauses $C = \{c_1, \dots, c_l\}$ and variables $X = \{x_1, \dots, x_n\}$ and its plane formula graph $G(\phi)$, we construct in polynomial time a bipartite planar graph $G'_1(\phi)$ such that ϕ is 1-in-3-satisfiable iff $\chi_O(G'_1(\phi)) = 1$. The graph $G'_1(\phi)$ has one *variable cycle* $v_i^0 \cdots v_i^{15}$ of length 16 for each variable x_i. There are exactly four ways to color a variable cycle; see Figure 3.10. Two of these color v_i^0 and v_i^8; using one of these colorings for the variable cycle of x_i correspond to setting x_i to `true`. Leaving v_i^0 and v_i^8 uncolored corresponds to setting x_i to `false`. For each clause c_j, $G'_1(\phi)$ contains a copy of the clause gadget depicted in Figure 3.10. We can compute an embedding of the formula graph $G(\phi)$ in which the variable vertices are placed on a horizontal line. The clause vertices are embedded above and below this horizontal line. If a clause c_j is embedded below the variables, we connect its black vertex to vertex v_i^8 of all variables occurring in c_j; otherwise, we use v_i^0. An example of this construction is depicted in Figure 3.11.

If ϕ is 1-in-3-satisfiable, coloring the variable cycles according to a satisfying assignment and the clause gadgets according to Figure 3.10 yields a coloring of $G'_1(\phi)$ in which the black vertex



(a) A *variable cycle*, with a conflict-free 1-coloring that corresponds to setting the variable to `true`. All conflict-free 1-colorings of a variable cycle result from this coloring by shifting the groups of colored vertices around the cycle. The vertices v_i^0 and v_i^8 that may be connected to the clause gadgets are drawn with a bold outline.

(b) A clause gadget. The orange vertices must be colored in any conflict-free 1-coloring. The white vertices cannot be colored. The black vertex cannot be colored, but does not have a conflict-free neighbor within the gadget. It is connected to the variables occurring in the clause, thus enforcing that exactly one of them is set to true.

Figure 3.10: Variable and clause gadgets for the reduction.

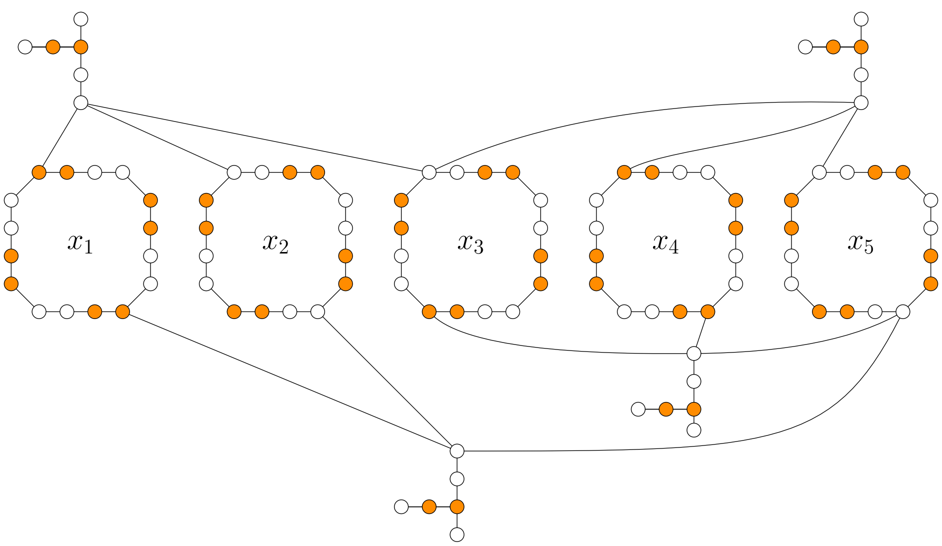


Figure 3.11: The graph $G_1'(\phi)$ resulting from applying the reduction to $\big\{\{x_1, x_2, x_3\}, \{x_1, x_2, x_5\}, \{x_2, x_4, x_5\}, \{x_3, x_4, x_5\}\big\}$, and an open-neighborhood conflict-free 1-coloring (orange vertices) corresponding to setting x_1 and x_4 to `true`.

of each clause is adjacent to exactly one colored neighbor. This coloring is an open-neighborhood conflict-free 1-coloring of ϕ. On the other hand, let $G_1'(\phi)$ have an open-neighborhood conflict-free 1-coloring χ. In each clause gadget, χ colors exactly the two orange vertices from Figure 3.10. Therefore, the black vertex of each clause has to be adjacent to exactly one colored variable vertex. Setting the variables corresponding to variable cycles with colored vertices v_i^0 and v_i^8 to `true` thus yields a 1-in-3-satisfying assignment for ϕ. $\square$

The same holds for $k = 2$ colors, but the restriction to bipartite planar graphs requires a slightly more sophisticated argument.

Theorem 3.6.6. *It is NP-complete to decide whether a bipartite planar graph G is open-neighborhood conflict-free 2-colorable.*

Proof. Again we prove hardness using a reduction from POSITIVE PLANAR 1-IN-3-SAT. From a positive planar 3-CNF formula ϕ with clauses $C = \{c_1, \ldots, c_l\}$ and variables $X = \{x_1, \ldots, x_n\}$ and its plane formula graph $G(\phi)$, we construct in polynomial time a bipartite planar graph $G_2'(\phi)$ such that ϕ is 1-in-3-satisfiable iff $\chi_O(G_2'(\phi)) \leq 2$. The graph $G_2'(\phi)$ has a *variable path* $v_i^1 v_i^2 v_i^3$ of length 3 for each variable x_i. For each clause c_j, there is a clause gadget as depicted in Figure 3.12; this gadget contains a distinguished *clause vertex*. The gadget prevents the clause vertex from being colored and cannot be used to provide a conflict-free neighbor to the clause vertex. We connect vertex v_i^1 to the clause vertex of c_j with an edge iff x_i occurs in c_j; the other vertices of clause gadgets and variable gadgets are not connected to any vertex outside their respective gadget. Therefore, variable vertex v_i^1 can provide a conflict-free neighbor to the clause vertex of c_j iff x_i occurs in c_j.

We still have to enforce that the color of the conflict-free neighbor of the clause vertex is the same for all clauses. To this end, we connect the clause vertices using the *equality gadget* depicted in Figure 3.13. This gadget ensures that the conflict-free neighbors of the two clause vertices connected by it have the same color in any conflict-free 2-coloring. We cannot add this gadget between all pairs of clause vertices because this would destroy planarity. Instead, we compute a spanning tree T on the clause vertices that could be added to $G_2'(\phi)$, preserving planarity. Then, for each edge $c_a c_b$ of T, we add a copy of the equality gadget to $G_2'(\phi)$, using it to connect the clause vertices c_a and c_b. Because adding the edges of T preserves planarity, the graph resulting from adding the gadgets is planar as well. Moreover, because the equality gadget works transitively and T is connected, the conflict-free neighbors of all clause vertices must receive the same color in any conflict-free 2-coloring.

It remains to prove that such a T always exists. For this purpose, consider the plane formula graph $G(\phi)$, including the backbone of the formula. Because only one vertex of each variable or clause gadget is connected to vertices outside the gadget, these gadgets do not influence the planarity of $G_2'(\phi)$. Therefore, if adding T preserves the planarity of $G(\phi)$, it also preserves the planarity of $G_2'(\phi)$. As root of T, we choose an arbitrary clause vertex r on the boundary of the unbounded face of $G(\phi)$. We add an edge from r to all other clause vertices on the boundary of the unbounded face to T. Now we consider the connected component R of r in T. Either $R = V(T)$, in which case we are done, or there must be a vertex $v \in R$ that lies on a face whose boundary contains a vertex $w \notin R$. For each such vertex v, we add an edge to all such vertices $w \notin R$. We iterate this procedure until we are done.

Let ϕ be 1-in-3-satisfiable and let Γ be the set of true variables in a 1-in-3-satisfying assignment of ϕ. We construct a conflict-free 2-coloring of $G_2'(\phi)$ by assigning color 1 to v_i^1 and v_i^2 for all $x_i \in \Gamma$ and to v_i^3 and v_i^2 for $x_i \notin \Gamma$. The vertices in equality gadgets that are adjacent to clause vertices receive color 2. All other vertices in the gadgets are colored as sketched in Figures 3.12 and 3.13. All clause vertices are adjacent to exactly one variable vertex carrying color 1 and thus have a conflict-free neighbor. Therefore, the coloring constructed in this way is a valid conflict-free 2-coloring.

Now assume that $G_2'(\phi)$ has a conflict-free 2-coloring χ. By the argument above, the conflict-free neighbor of each clause vertex is a variable vertex v_i^1. Moreover, all clause vertices have a conflict-free neighbor of the same color; w.l.o.g., color 1. Therefore, each clause vertex is adjacent

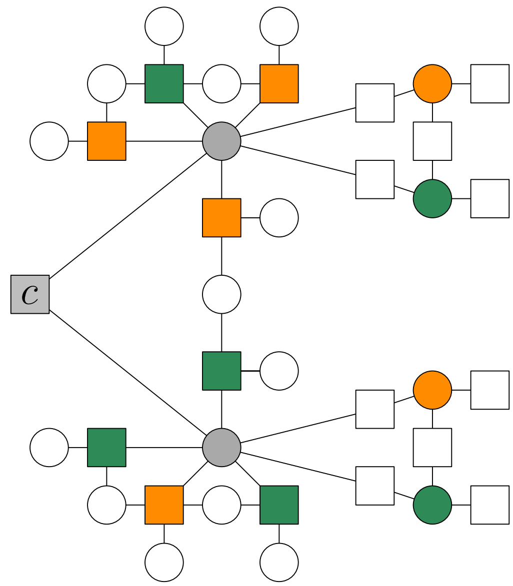


Figure 3.12: The bipartite clause gadget with clause vertex c; the components of the bipartition are indicated using squares and circles. Gray vertices cannot receive a color. Vertices colored green or orange must be colored. Except for automorphisms and swapping colors, orange vertices have to receive color 1 and green vertices have to receive color 2. White vertices may be colored or may remain uncolored; it is straightforward to extend the depicted coloring to a conflict-free 2-coloring of the gadget (except for c) by coloring the white vertices of degree 1. By construction, one of c's neighbors has three neighbors of color 1 and a conflict-free neighbor of color 2 (and vice versa for c's other neighbor). In total, the gadget guarantees that c remains uncolored and cannot have a colored neighbor within the gadget.

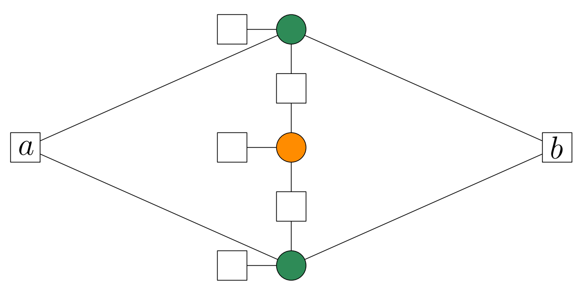


Figure 3.13: The *equality gadget* that can be used to connect two terminal vertices (marked a and b) in the same partition of a bipartite graph. It adds two occurrences of the same color to the neighborhoods of a and b, thereby forcing the conflict-free neighbor of a and b to have the same color.

to exactly one variable vertex with color 1, and the set of variables x_i where $\chi(v_i^1) = 1$ induces a satisfying assignment of ϕ. $\square$

3.7 Conclusion

A spectrum of open questions remain. Many of them are related to general graphs, in particular with our sufficient condition for general graphs. For every $k \geq 2$, G_{k+1} provides an example that excluding K_{k+2} as a minor is not sufficient to guarantee k-colorability. However, for $k \geq 2$ we have no example where excluding K_{k+3}^{-3} as a minor does not suffice.

With respect to open-neighborhood conflict-free coloring, several open questions remain. Are four colors always sufficient for general planar graphs? Are three colors always sufficient for outer-planar graphs?

Another variant of our problem arises from requiring that *all* vertices must be colored. It is clear that one extra color suffices for this purpose; however, it is not always clear that this is also necessary, in particular, for planar graphs. Adapting our argument to this situation does not seem straightforward, especially because there are outerplanar graphs requiring three colors in this setting.

In addition, there is a large set of questions related to geometric versions of the problem. What is the worst-case number of colors for straight-line visibility graphs within simple polygons? It is conceivable that $\Theta(\log \log n)$ is the right answer, just like for rectangular visibility, but this is still an open problem, just like complexity and approximation. Other questions arise from considering geometric intersection graphs, such as unit-disk intersection graphs, for which necessary and sufficient conditions, just like upper and lower bounds, would be quite interesting.

Bibliography

[AAG$^+$18] Zachary Abel, Victor Alvarez, Aman Gour, Adam Hesterberg, Erik D. Demaine, Sándor P. Fekete, Phillip Keldenich, and Christian Scheffer. Three colors suffice: Conflict-free coloring of planar graphs. *Proceeedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2017)*, 2018. doi 10.1137/1.9781611974782.127.

[AdBP08] Mohammad Ali Abam, Mark de Berg, and Sheung-Hung Poon. Fault-tolerant conflict-free colorings. In *Proc. 20th Canadian Conference on Computational Geometry (CCCG'08)*, pages 13–16, 2008.

[ADK15] Pradeesha Ashok, Aditi Dudeja, and Sudeshna Kolay. Exact and FPT algorithms for max-conflict free coloring in hypergraphs. In *Proc. 26th International Symposium on Algorithms and Computation*, pages 271–282, 2015.

[AEGR07] Deepak Ajwani, Khaled Elbassioni, Sathish Govindarajan, and Saurabh Ray. Conflict-free coloring for rectangle ranges using $O(n^{.382})$ colors. In *SPAA '07: Proc. 19th ACM Symposium on Parallelism in Algorithms and Architectures*, pages 181–187, 2007.

[AFN04] Jochen Alber, Michael R Fellows, and Rolf Niedermeier. Polynomial-time data reduction for dominating set. *Journal of the ACM*, 51(3):363–384, 2004.

[AH77a] K. Appel and W. Haken. Every planar map is four colorable. Part I. Discharging. *Illinois J. Math.*, 21:429–490, 1977.

[AH77b] K. Appel and W. Haken. Every planar map is four colorable. Part II. Reducibility. *Illinois J. Math.*, 21:491–567, 1977.

[AS06] Noga Alon and Shakhar Smorodinsky. Conflict-free colorings of shallow discs. In *Proc. 22nd Annual Symposium on Computational Geometry*, pages 41–43. ACM, 2006.

[Bal78] Hans Werner Ballmann. Der Satz von Lusternik und Schnirelmann. *Bonner Mathematische Schriften*, 102:1–25, 1978.

[BH94] Brenda S. Baker. and Murray Hill. Approximation algorithms for NP-complete problems on planar graphs. *Journal of the ACM*, 41(1):153–180, 1994.

[BNCOS10] Amotz Bar-Noy, Panagiotis Cheilaris, Svetlana Olonetsky, and Shakhar Smorodinsky. Online conflict-free colouring for hypergraphs. *Combinatorics, Probability and Computing*, 19(04):493–516, 2010.

[CFK$^+$07] K. Chen, A. Fiat, H. Kaplan, M. Levy, J. Matousek, E. Mossel, J. Pach, M. Sharir, S. Smorodinsky, U. Wagner, and E. Welzl. Online conflict-free coloring for intervals. *SIAM J. Computing*, 36:1342–1359, 2007.

[CGRS14] Panagiotis Cheilaris, Luisa Gargano, Adele A Rescigno, and Shakhar Smorodinsky. Strong conflict-free coloring for intervals. *Algorithmica*, 70(4):732–749, 2014.

[CSS11] Panagiotis Cheilaris, Shakhar Smorodinsky, and Marek Sulovsky. The potential to improve the choice: list conflict-free coloring for geometric hypergraphs. In *Proc. 27th Annual Symposium on Computational Geometry*, pages 424–432. ACM, 2011.

[CT11] Panagiotis Cheilaris and Géza Tóth. Graph unique-maximum and conflict-free colorings. *Journal of Discrete Algorithms*, 9(3):241–251, 2011.

[DO07] Erik D. Demaine and Joseph O'Rourke. Reducibility among combinatorial problems. In *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, chapter 24 Geodesics: Lyusternik-Schnirelmann, pages 372–375. Cambridge University Press, Cambridge, 2007.

[ELRS03] G. Even, Z. Lotker, D. Ron, and S. Smorodinsky. Conflict-free colorings of simple geometric regions with applications to frequency assignment in cellular networks. *SIAM Journal on Computing*, 33(1):94–136, 2003.

[EM06] Khaled Elbassioni and Nabil H Mustafa. Conflict-free colorings of rectangles ranges. In *Annual Symposium on Theoretical Aspects of Computer Science*, pages 254–263. Springer, 2006.

[Gar65] Martin Gardner. Letters. *Scientific American*, 213(5):10–12, November 1965. Reproduced in [Gar90].

[Gar90] Martin Gardner. *Mathematical Carnival*. Penguin Books, London, 1990.

[GR15] Luisa Gargano and Adele A. Rescigno. Complexity of conflict-free colorings of graphs. *Theoretical Computer Science*, 566:39–49, 2015.

[GST14] Roman Glebov, Tibor Szabó, and Gábor Tardos. Conflict-free coloring of graphs. *Combinatorics, Probability and Computing*, 23:434–448, 2014.

[Guy61] Richard K. Guy. The jewel thief. *NABLA*, 8:149–150, September 1961.

[Had43] Hugo Hadwiger. Über eine Klassifikation der Streckenkomplexe. *Vierteljschr. Naturforsch. Ges. Zürich*, 88:133–143, 1943.

[HKS10] Elad Horev, Roi Krakovski, and Shakhar Smorodinsky. Conflict-free coloring made stronger. In *Proc. 12th Scandinavian Symposium and Workshop on Algorithm Theory*, volume 6139, pages 105–117, 2010.

[HKS+15] Frank Hoffmann, Klaus Kriegel, Subhash Suri, Kevin Verbeek, and Max Willert. Tight bounds for conflict-free chromatic guarding of orthogonal art galleries. In *31st International Symposium on Computational Geometry*, volume 34 of *LIPIcs*, pages 421–435, 2015.

[HPS05] Sariel Har-Peled and Shakhar Smorodinsky. Conflict-free coloring of points and simple regions in the plane. *Discrete & Computational Geometry*, 34(1):47–70, 2005.

[Kar72] Richard Karp. Reducibility among combinatorial problems. In R. E. Miller, J. W. Thatcher, J. D., and Bohlinger, editors, *Complexity of Computer Computations*, pages 85–103. Springer, Boston, 1972.

[KYK83] Tohru Kikuno, Noriyoshi Yoshida, and Yoshiaki Kakuda. A linear algorithm for the domination number of a series-parallel graph. *Discrete Applied Mathematics*, 1983.

[Lic82] David Lichtenstein. Planar formulae and their uses. *SIAM Journal on Computing*, 11(2):329–343, 1982.

[LS29] Lazar Lyusternik and Lev Schnirelmann. Sur le probléme de trois géodésiques fermées sur les surfaces de genre 0. *Comptes Rendus de l'Académie des Sciences de Paris*, 189:269–271, 1929.

[LTP09] Nissan Lev-Tov and David Peleg. Conflict-free coloring of unit disks. *Discrete Applied Mathematics*, 157(7):1521–1532, 2009.

[MR08] Wolfgang Mulzer and Günter Rote. Minimum-weight triangulation is NP-hard. *Journal of the ACM*, 55(2):11, 2008.

[O'B61] Thomas H. O'Beirne. Christmas puzzles and paradoxes. *The New Scientist*, 266:753, December 1961.

[O'R18] Joseph O'Rourke. Personal communication, 2018. Alluded to in [DO07].

[Pog73] Aleksej Vasilevich Pogorelov. *Extrinsic Geometry of Convex Surfaces*. 1973.

[Poi05] Henri Poincaré. Sur les lignes géodésiques des surfaces convexes. *Transactions of the American Mathematical Society*, 6(3):237–274, 1905.

[PT09] J. Pach and G. Tárdos. Conflict-free colourings of graphs and hypergraphs. *Combinatorics, Probability and Computing*, 18(05):819–834, September 2009.

[RSST97] N. Robertson, D. Sanders, P. Seymour, and R. Thomas. The four-colour theorem. *J. Combinatorial Theory Series B*, 70:2–44, 1997.

[Smo03] S. Smorodinsky. *Combinatorial Problems in Computational Geometry*. PhD thesis, School of Computer Science, Tel-Aviv University, 2003.

[Sys79] Maciej M. Sysło. Characterizations of outerplanar graphs. *Discrete Mathematics*, 26:47–53, 1979.

[Wil13] R. Wilson. *Four colours suffice: How the map problem was solved*. Princeton University Press, 2013.