Connectivity Properties of the Flip Graph After Forbidding Triangulation Edges

by

Reza Bigdeli

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2022

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Reza Bigdeli was the sole author of the Sections 1, 2, 3.1, 3.2.3,3.3.1, 3.3.2, and 4.1 which were written under the supervision of Prof. Anna Lubiw and were not written for publication.

The rest of the chapters are joint work with my supervisor Prof. Anna Lubiw and appeared in the Young Researchers Forum of the Symposium on Computational Geometry and on arXiv [5].

Abstract

The ***flip graph*** for a set P of points in the plane has a vertex for every triangulation of P, and an edge when two triangulations differ by one ***flip*** that replaces one triangulation edge by another. The flip graph is known to have some connectivity properties: (1) the flip graph is connected; (2) connectivity still holds when restricted to triangulations containing some *constrained* edges between the points; (3) for P in general position of size n, the flip graph is $\lceil \frac{n}{2} - 2 \rceil$-connected, a recent result of Wagner and Welzl (SODA 2020).

We introduce the study of connectivity properties of the flip graph when some edges between points are *forbidden*. An edge e between two points is a ***flip cut edge*** if eliminating triangulations containing e results in a disconnected flip graph. More generally, a set X of edges between points of P is a ***flip cut set*** if eliminating all triangulations that contain edges of X results in a disconnected flip graph. The ***flip cut number*** of P is the minimum size of a flip cut set.

We give a characterization of flip cut edges that leads to an $O(n \log n)$ time algorithm to test if an edge is a flip cut edge and, with that as preprocessing, an $O(n)$ time algorithm to test if two triangulations are in the same connected component of the flip graph. For a set of n points in convex position (whose flip graph is the 1-skeleton of the associahedron) we prove that the flip cut number is $n - 3$.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Triangulations

Given a set P of n points in the plane, which may include collinear points, an ***edge*** of P is a line segment pq that intersects P in exactly the two endpoints p and q. A ***triangulation*** of P is a maximal set of non-crossing edges. To give an intuition of what a triangulation of a point set would look like, first we introduce the *convex hull* of a point set. The ***convex hull*** of a point set P consists of the edges pq such that one of the two closed half-planes determined by the line through pq contains all the points of P. This means that the Convex Hull can be seen as the smallest container of the points in the point set. Any triangulation of the point set contains all the edges of the Convex Hull of that point set. Then, edges are added to the points inside the hull in a way that finally the area inside the hull is divided into *empty* triangles. A triangle is ***empty*** if there do not exist any points of P in the triangle. This is why it is called a triangulation; we add edges until the whole area is divided into triangle regions. Figure 1.1 shows some point sets and a valid triangulation of each point set. Triangulations have important applications in graphics and mesh generation [4, 13] and are of significant mathematical interest [12].

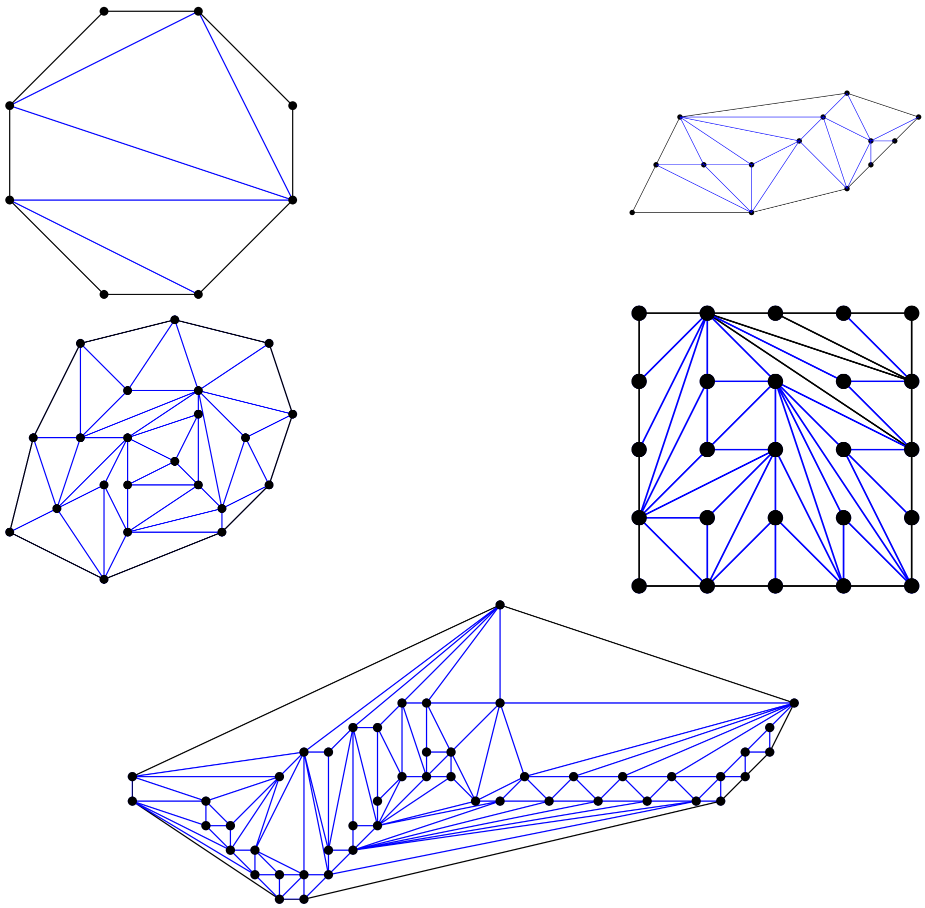


Figure 1.1: Some point sets and one of their valid triangulations. The black edges are the edges of the convex hull. Adding the blue edges, triangulating the point set will be completed.

A fundamental approach to understanding triangulations is by means of *flips*. A ***flip*** operates on a triangulation by removing one edge pq and adding another edge uv to obtain a new triangulation—of necessity, the edges pq and uv will cross and their four endpoints will form a convex quadrilateral with no other points of P inside it. For example, in Figure 1.4, edge a_1b_1 can be flipped to uv. We say we can ***transform*** or ***reconfigure*** one triangulation into another one, if there exists a sequence of flips after which the first triangulation is converted into the other one. In 1972, Lawson [23, 24] proved that any triangulation of point set P can be reconfigured to any other triangulation of P by a sequence of flips.

1.2 Reconfiguration and Flip Graph

A ***graph*** G is a mathematical tool used for expressing the relation between some objects. It has a wide range of applications in networks, combinatorial optimization, machine learning, and so on. Each graph consists of some ***nodes*** or ***vertices*** and some ***edges***. Each vertex represents an object in the environment we are studying. There exists an edge between vertices of the graph if the objects those vertices are representing satisfy a specific relation. For example, consider the people in a university. We want to construct a graph that shows the friendship relation between people at the university. So, each person would be represented with a node and we place an edge between nodes that correspond to two people who are friends. This would be the friendship graph of the mentioned university.

We denote a graph by $G = (V, E)$ where set $V = \{v_1, v_2, ..., v_n\}$ is the set of the graph's vertices and E denotes the set of graph edges. We call a sequence of vertices $v_1, v_2, .., v_{l+1}$ a path of length l between v_1 and v_{l+1} if there exists an edge between every two consecutive vertices v_i and v_{i+1} for $i = 1, \ldots, l$. The length of this path is the number of edges along this path. A ***shortest path*** between two vertices u and v is defined as a path with the smallest length and the length of this path is called the ***distance*** between the two vertices and is denoted by $d(u, v)$. The maximum distance between any pair of vertices is called the ***diameter*** of the graph. That is, diameter $= \max\{d(v, u) : v, u \in V\}$. We say a graph is connected if there exists a path between every arbitrary pair of vertices.

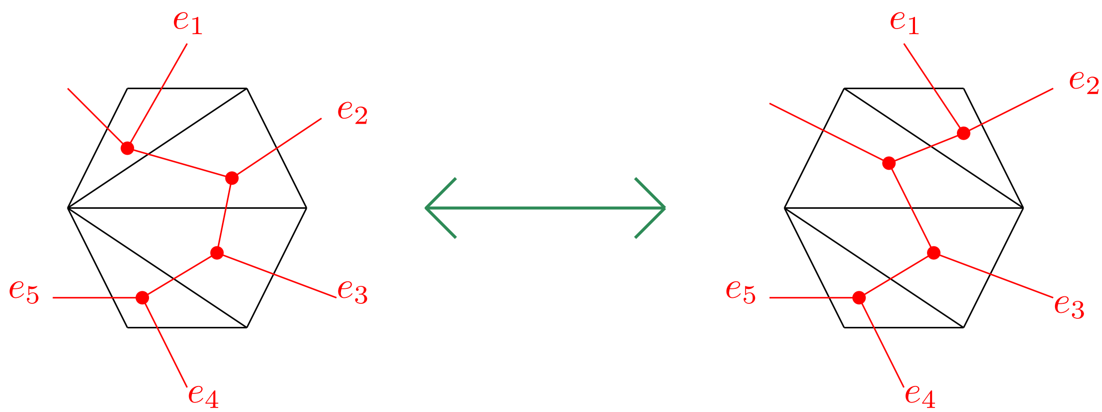
We can represent flips and triangulations using graphs. A ***flip graph*** has a vertex for every triangulation of P and an edge when two triangulations differ by a flip. In this way we can define triangulation flips as a *reconfiguration* problem. ***Reconfiguration*** is the study of the relationship between the feasible solutions of a problem [31]. Each feasible solution is considered as a valid configuration and reconfiguration consists of some steps to transform one feasible solution into another one in a way that each intermediary step is also a feasible solution to the problem (in other words a valid configuration). Some sensible examples of reconfiguration problems are *Sliding Puzzles*, *Rubik's Cube*, etc. In the ***reconfiguration graph***, each vertex represents a configuration and each edge shows the reconfiguration operation which is the flip in triangulations. As mentioned, Lawson proved that any two triangulations can be reconfigured into each other. This means there exists a path between any two vertices in the flip graph. Therefore, the flip graph is connected. Reconfiguring triangulations via flips is well studied, see the survey by Bose and Hurtado [6]. However, there are some very interesting open questions, and many properties of flip graphs remain to be discovered.

In addition to properties of the regular flip graph, properties of some specific flip graphs have been studied too. Especially, the flip graph of triangulations that must contain a specific set of edges has been studied widely. In this case, the flip graph would be a subgraph of the regular flip graph. One intriguing thing about flip graphs of triangulations is that many properties carry over when we restrict to triangulations containing some specified non-crossing edges—so-called *constrained triangulations*. The subgraph of the flip graph consisting of triangulations that contain all the constrained edges is connected, as proved by Chew [10]. This encourages people working on combinatorial reconfiguration to consider special cases of the reconfiguration graph to find more properties (especially connectivity properties) for different reconfiguration problems. As a matter of fact, that is one of the main reasons for our contribution to the subject of this thesis.

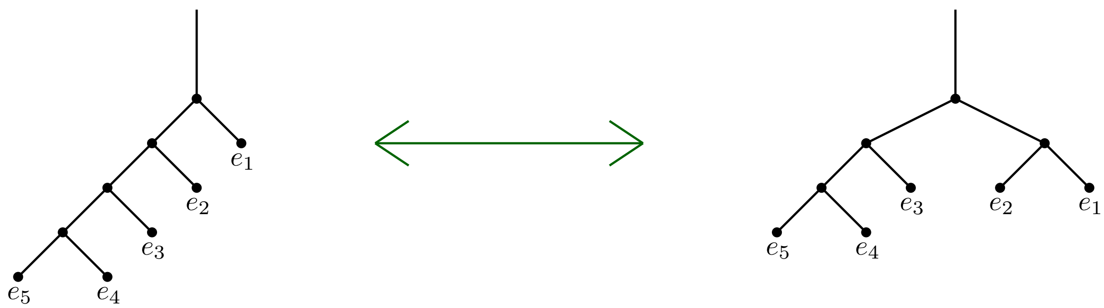
1.3 Points in Convex Position

A well-known and important class of points sets is the class of *convex point sets*. A point set P is a ***convex point set*** if all the points of P are extreme points of the convex hull of P, i.e., for every point $p \in P$ there is a line through p with all other points of P strictly to one side of the line.

The case of points in convex position is especially interesting because there is a bijection between flips in triangulations of a convex point set and rotations in binary trees [35], so that flip distance becomes rotation distance between binary trees. Figure 1.2 demonstrates the relationship between flips in convex point sets and rotation trees.



(a) Two triangulations that can be reconfigured into each other using a flip. Their corresponding tree and how it can be built has been shown in red.



(b) Two corresponding trees and how they can be reconfigured into each other using a rotation.

Figure 1.2: The relationship between triangulations of a convex point set and binary trees.

Finding the rotation distance between two binary trees is of great interest in biology for phylogenetic trees [11], and in data structures for splay trees [35]. Furthermore, the flip graph for n points in convex position is the 1-skeleton (the graph of vertices and edges) of an $(n-3)$-dimensional polytope called the *associahedron* [25]. See Figure 1.3.

1.4 Our results

In this thesis, we study connectivity properties of the flip graph when—instead of constraining certain edges between points to be present as in constrained triangulations—we *forbid* certain edges between points. To be precise, if a set X of edges between points is forbidden, we eliminate all triangulations that contain an edge of X, and examine whether

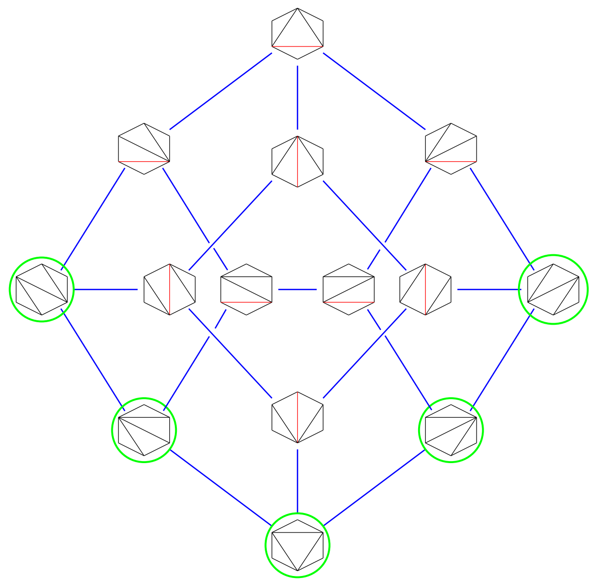


Figure 1.3: The flip graph of points of a convex hexagon is the 1-skeleton of an associahedron. If we forbid the two red edges, the resulting flip graph (with vertices circled in green) is connected.

the flip graph on the remaining triangulations is connected. We say that X is a ***flip cut set*** if the resulting flip graph is disconnected; in the special case where X is a single edge, we say that the edge is a ***flip cut edge***. For example the edge uv in Figure 1.4 is a flip cut edge, but the two red edges in Figure 1.3 do not form a flip cut set. We define the ***flip cut number*** of a set of points to be the minimum size of a flip cut set. This is analogous to the *connectivity* of a graph—the minimum number of vertices whose removal disconnects the graph.

Recently, Wagner and Welzl [38] have investigated some connectivity properties of the flip graph. They have shown that the flip graph is $\lceil \frac{n}{2} - 2 \rceil$-connected for P in general position. Since the structure of the flip graph depends on the edges between the points, it seems more natural to study connectivity of the flip graph after deleting some of these edges, rather than deleting some vertices of the flip graph, as standard graph connectivity does, and as the result of Wagner and Welzl [38] does.

As our main result, in Chapter 3.2 we characterize when an edge e is a flip cut edge in terms of connectivity (in the usual graph sense) of the edges that cross e. Observe that a triangulation that does not contain e must contain an edge that crosses e. We then use the characterization to give an $O(n \log n)$ time algorithm to test if a given edge e in a point

Figure 1.4: The smallest point set that has a flip cut edge. The edge $e = uv$ is a flip cut edge since forbidding e leaves two possible triangulations (as shown) and neither one allows a flip.

set of size n is a flip cut edge in Section 3.3. With that algorithm as preprocessing, we give a linear-time algorithm to test if two triangulations are still connected after we eliminate from the flip graph all triangulations containing edge e.

For the case of n points in convex position, there are no flip cut edges, and we show that the flip cut number is $n - 3$. For example, in Figure 1.3 the leftmost and rightmost triangulations become disconnected if we forbid one more edge, which yields a flip cut set of size 3 for $n = 6$. More details can be found in Chapter 4.

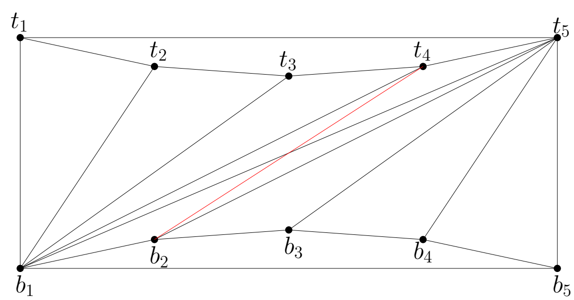


Figure 1.5: The "channel", and a triangulation that becomes ***frozen*** (an isolated vertex in the flip graph) if we forbid the edge b_2, t_{n-1} (in red). In fact, every edge $b_i t_j$, $i, j \notin \{1, 5\}$ is a flip cut edge.

In Section 3.4.2, we show that a point set of size n may have $\Theta(n^2)$ flip cut edges (see Figure 1.5), and in Section 3.4.4, we show that a flip cut edge may result in $\Theta(n)$ disconnected components in the flip graph. We also examine various special point sets whose flip graphs have been previously studied in Section 3.4, such as points on an integer grid [8] and, more generally, point sets without empty convex pentagons [14]. Our characterization of flip cut edges becomes simpler in the absence of empty convex pentagons. Point sets without empty convex pentagons must have collinear points; our results do not assume points in general position.

1.4.1 Examples

In this chapter we are going to provide some examples of flip cut edges and flip cut sets. However the details for each example are provided in later related chapters.

Figure 1.4 shows the smallest example of a point set that has a flip cut edge—in fact, by forbidding one edge, we get a triangulation from which no flips are possible. The triangulation associated with such an isolated vertex in the flip graph is called a ***frozen*** triangulation.

The point set shown in Figure 1.5, which is called a ***channel***, is created in the following way: first, put four points on four corners of an axis-aligned rectangle. Then, add n points from a flat convex curve connecting two points on the upper corners. These n points are called the upper chain. Finally, add n points of a concave flat curve that connects the two points on the lower corners. This is called the lower chain. The triangulation of this point set shown in Figure 1.5 becomes frozen (modulo triangulating the upper and lower convex subpolygons) when we forbid one flip cut edge. In fact, any edge joining an interior vertex of the top curve and an interior vertex of the bottom curve is a flip cut edge. So there are $\Theta(n^2)$ flip cut edges. We justify this more carefully in Section 3.4.2.

Grid points are another well-studied case for triangulation flips [8], in part because of physics applications. For n points lying on a $\sqrt{n} \times \sqrt{n}$ grid, there are again $\Theta(n^2)$ flip cut edges. In fact, for an infinite grid, every edge is a flip cut edge, though boundary effects interfere in finite grids. See Section 3.4.2. Points in a grid have the special property that there are no empty convex pentagons. Flips for point sets without empty convex pentagons were studied by Eppstein [14]. Such point sets must have collinear points [1]. See Section 3.4.1.

The "hourglass" shown in Figure 1.6 has a flip cut edge $e = uv$ such that forbidding e results in $\Theta(n)$ disconnected components in the flip graph, which is the most possible. See Section 3.4.4.

Some point sets have $\Theta(n^2)$ flip cut edges, but at the other extreme, some point sets have no flip cut edges. For example, there are no flip cut edges for points in convex position. The standard way to flip between two triangulations of a convex polygon is via a *star triangulation*. A ***star triangulation*** is a triangulation with all edges (except the edges of the convex hull) incident to one point. If there is a point not incident to any forbidden edge, we can still flip to a star centered on that point. See Figure 1.7. A more

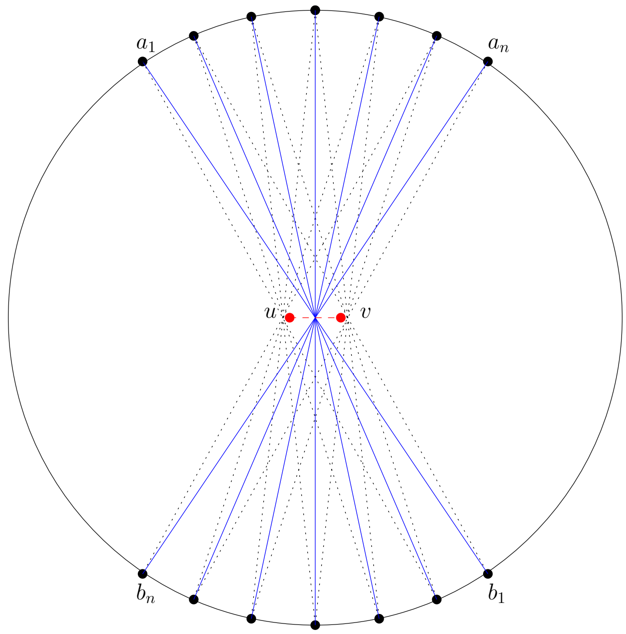


Figure 1.6: In this "hourglass" uv is a flip cut edge that creates n disconnected components in the flip graph, one for each a_ib_i.

detailed version of this argument is given in Lemma 24 when we investigate flip cut sets for points in convex position.

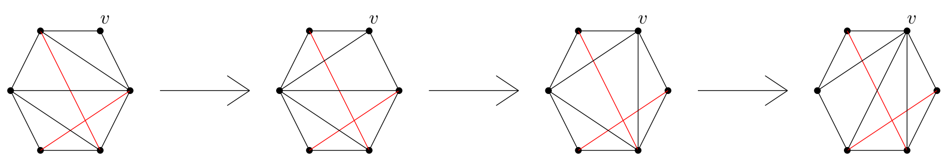


Figure 1.7: A convex hexagon with a set X of two forbidden edges (in red), and a flip sequence to connect one triangulation in $\mathcal{T}_{-X}$ to the triangulation that is a star at v, without using any forbidden edges.

Figure 1.8 shows some more examples of flip cut edges in point sets.

1.5 Notation and Definitions

We denote the flip graph of point set P by $\boldsymbol{\mathcal{F}(P)}$, or just $\boldsymbol{\mathcal{F}}$, when P is clear from context.

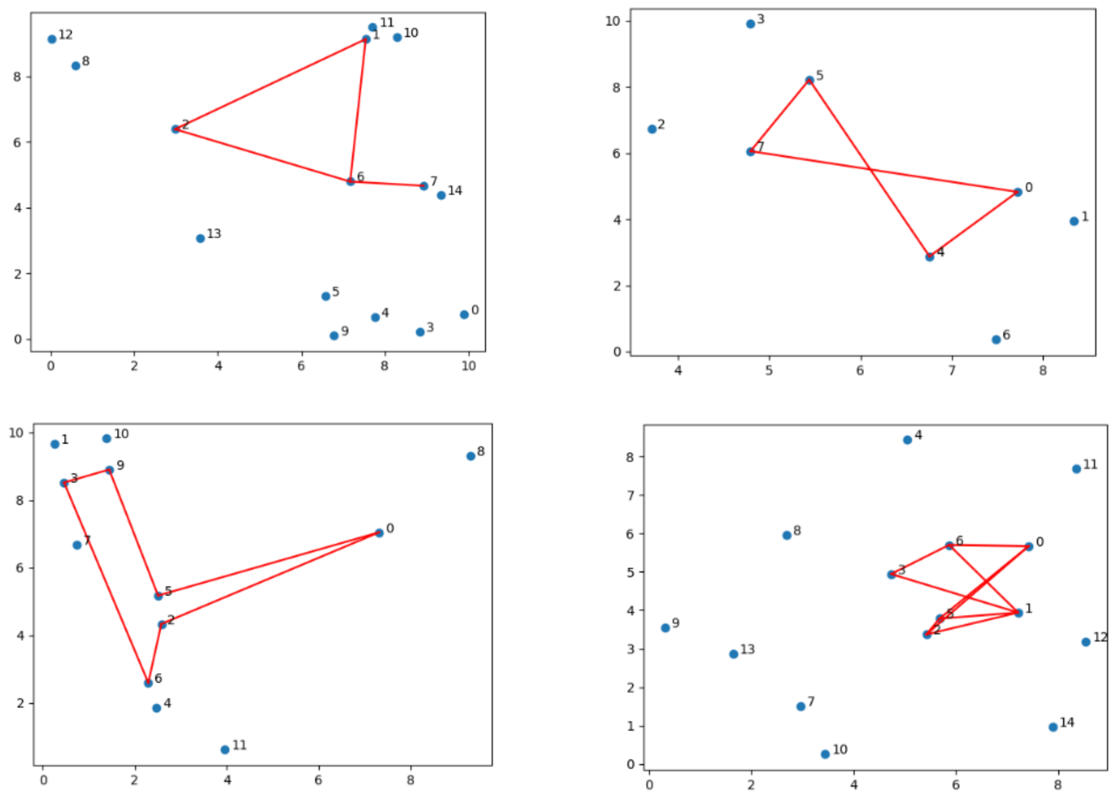


Figure 1.8: Some point sets and their flip cut edges (in red).

For a subset E of the edges of P, let $\boldsymbol{\mathcal{T}_{+E}(P)}$ be the set of triangulations of P that include all the edges of E. These are known as ***constrained triangulations***. Let $\boldsymbol{\mathcal{F}_{+E}(P)}$ be the subgraph of the flip graph induced on the vertex set $\mathcal{T}_{+E}(P)$. It is known that $\mathcal{F}_{+E}(P)$ is connected [10].

Let $\boldsymbol{\mathcal{T}_{-E}(P)}$ be the set of triangulations of P that include *none* of the edges of E, and let $\boldsymbol{\mathcal{F}_{-E}(P)}$ be the subgraph of the flip graph induced on $\mathcal{T}_{-E}(P)$. When E consists of a single edge e, we will write $\mathcal{T}_{-e}(P)$, and so on. Also, we will omit P when the point set is clear from the context.

A subset E of edges of P is a ***flip cut set*** if the flip graph $\mathcal{F}_{-E}(P)$ is disconnected. The smallest size of a flip cut set is called the ***flip cut number*** of P. If $\{e\}$ is a flip cut set of size one, we call e a ***flip cut edge***.

An ***empty convex quadrilateral*** or ***EC4*** is a set of 4 points in P that form a convex quadrilateral with no other points of P inside or on the boundary. We also use ***EC3*** for empty triangles, ***EC5*** for empty convex pentagons, and so on.

Point set P is a ***convex point set*** if all the points of P are extreme points of the convex hull of P, i.e., for every point $p \in P$, there is a line through p with all other points of P strictly to one side of the line. Point set P is in ***general position*** if no three points are collinear.

Chapter 2

Related Work

In this section, we briefly discuss related work on general reconfiguration problems, flip graphs, and questions related to these topics. Then, we give some background on flip graphs of triangulations, some variations, and the problems that arise in this specific setting of reconfiguration.

Reconfiguration and Flip Graphs

As described in the previous section, *reconfiguration* is the study of the relationship between the feasible solutions of a problem. The reconfiguration, or “flip” graph has a vertex for each feasible solution of the problem, and an edge between two solutions if they can be transformed into each other via an operation called a flip. There are many famous reconfiguration problems including “15-puzzle”, “independent set reconfiguration”, “vertex cover reconfiguration”, “k-coloring reconfiguration”, “dominating set reconfiguration”, “sliding puzzle”, etc. People interested in reconfiguration problems study three important question for each problem: (1) Is the flip graph for the reconfiguration problem connected? (2) What is the asymptotic (or exact) diameter of the flip graph? (3) What is the complexity of finding the shortest path between two configuration in the flip graph? These are the three well-known questions that people studying reconfiguration try to answer. Further information on these questions and different reconfiguration problems can be found in two surveys about reconfiguration by Nishimura [31] and van den Heuvel [36].

Reconfiguring Triangulations

Connectivity of the flip graph. In the previous chapter we discussed that Lawson proved that any two arbitrary triangulations T_1, T_2 of a point set P can be transformed into each other using a sequence of flips. Lawson proved this using two approaches. Both of the proofs use a canonical triangulation and prove that any triangulation can be transformed into this canonical triangulation. In the first approach [23], the canonical triangulation can be constructed in the following way: Sort the points by their x-coordinates. First, create a triangle with the first three points. Then, add other points in order one by one and add the edge between the added point and each of the previous ones if it does not cross the previous edges. Lawson [23] proved that any triangulation can be reconfigured into this canonical triangulation which shows that any two triangulations can be transformed into each other via a sequence of flips. The second approach [24] used *Delaunay triangulations* (which were introduced by Lee and Lin [26]) as canonical triangulations. A triangulation of a point set P is a ***Delaunay triangulation*** if for any triangle Δabc, we have that the circle through a, b and c does not contain any other point of P inside it. Again, Lawson showed that any triangulation can be reconfigured to the Delaunay triangulation, hence any two triangulations can be transformed into each other by a sequence of flips. Note that in both of these proofs, in the sequence of the flips, first we reconfigure one triangulation to the canonical one and then reconfigure the canonical triangulation to the second one.

Diameter of the Flip Graph. There is considerable work on distances and diameter of flip graphs. Lawson [24], for example, showed that for both canonical triangulations, $O(n^2)$ flips are sufficient to reconfigure any triangulation into the canonical one. This means $O(n^2)$ is an upper bound for the number of flips needed for transforming any triangulation into another one. Hurtado et al. [21] proved that this upper bound is tight. In other words, they found a point set and two specific triangulations of that point set which need $\Omega(n^2)$ flips in order to reconfigure one into the other. This point set is known as a "channel" and the two triangulations are shown in Figure 2.1. Note that the upper and lower polygons need to be triangulated in order to complete the triangulations. Also, it can be seen that the only possible flip in the middle for each of the two triangulations, are b_1t_5 for the left channel and t_1b_5 for the right channel. In order to prove the $\Omega(n^2)$ bound we can consider a binary encoding of the triangulations. We encode triangles with two vertices from the upper chain and one vertex from the lower chain by 1 and triangles with two vertices from the lower chain and one vertex from the upper chain by 0. Then the right triangulation is encoded by 11110000 and the left one is encoded by 00001111. Each flip here is equal to swapping a neighbouring 0 and 1. So, it can be shown that $\Omega(n^2)$ swaps or flips are needed to transform one encoding (triangulation) to another one.

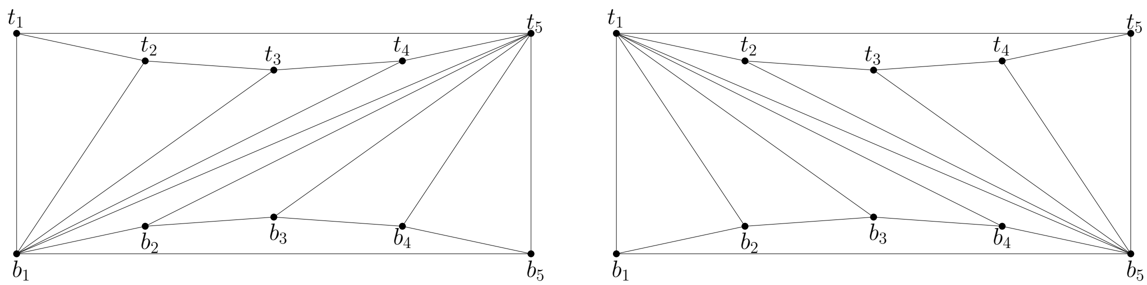


Figure 2.1: Two triangulaions of a channel that require $\Omega(n^2)$ flips in order to transform one into the other.

Hurtado et al. [21] showed that for a point set P with k *convex layers* (the ***convex layers*** of a point set are the sequence of convex hulls of the point set, each obtained after removing the vertices of the previous layer), each triangulation of P can be transformed to another one with at most $O(nk)$ flips. They also showed that at most $O(n + k^2)$ flips are needed to reconfigure a triangulation of an n-gon with k *reflex vertices* to another one. A ***reflex vertex*** is a vertex that has a concave internal angle inside the polygon.

Bounds on Flip Distance Between Triangulations. As can be seen from the above discussion, most of the proofs for connectivity of the flip graph use some canonical triangulation to reconfigure two triangulations. However, usually the shortest way to transform one configuration into another does not include a canonical configuration in the intermediary steps. This observation led to finding some bounds on the distance between two triangulations T_1 and T_2. Hanke et al. [18] provided an upper bound on the minimum number of flips needed to transform one triangulation into another one. They showed that the number of intersections between the edges of the two triangulations is an upper bound on the distance between those two triangulations. Eppstein [14] provided a lower bound on the flip distance between two triangulations of the point set using *quadrilateral graphs*. In a ***quadrilateral graph*** of point set P, there is a vertex for each edge between two points of the point set and there exists a graph edge between those two vertices if the four endpoints of the corresponding edges create a convex empty quadrilateral (EC4). We denote the quadrilateral graph by QG. Eppstein showed for two triangulations T_1 and T_2 on point set P if we define a complete bipartite graph between the edges of T_1 and T_2 and assign the weight of an edge to be the distance between the corresponding edges in QG, then the weight of the minimum perfect matching in this bipartite graph is a lower bound on the flip distance between T_1 and T_2.

Complexity of the Flip Distance Between Two Triangulations. Kanj et al. [22] showed that the problem of finding the flip distance between two triangulations is fixed-parameter tractable, that is, there is an $O(n + kc^k)$ time algorithm that identifies whether two triangulations can be reconfigured into each other using at most k flips. For a general point set, finding the distance in the flip graph between two triangulations (the "flip distance") is known to be NP-hard [28], and even APX-hard [32]. NP-hardness of the flip distance problem is proved using channels as the building block for the reduction.

Variations: Constrained Triangulations. As mentioned in the previous chapter, sometimes some special families and types of triangulations (and flip graphs as a result) are studied for their properties and ***constrained triangulations*** is the most famous one among them. In 1987, Chew [10] introduced the notion of constrained triangulations and constrained Delaunay triangulations. Given a set X of constrained edges, the ***constrained triangulations*** are the ones that contain all the edges of X, i.e., $\mathcal{T}_{+X}$ in our notation. The ***constrained Delaunay triangulation (CDT)*** is a triangulation in $\mathcal{T}_{+X}$ that is as close as possible to the Delaunay triangulation. That is, CDT contains all the edges of X and for each edge $e = uv$ of the CDT there exists a circle through u and v, and for any point $p \in P$ inside the circle there is an edge $f \in X$ that crosses one of the segments pv or pu. Chew [10] showed that any constrained triangulation can be flipped to a CDT while keeping all the constrained edges, which implies connectivity of the flip graph for constrained triangulations (which we denote by $\mathcal{F}_{+X}$). Houle et al. [20] introduced another family of triangulations and another subgraph of the flip graph. This new family, which consists of only triangulations that admit a perfect matching, was also proven to have connected flip graphs [20].

Variations: Simultaneous Flips The next variation, introduced by Hurtado et al. [21], is the notion of *simultaneous flips* in triangulations. There are some flips in triangulations that can be done at the same time. A set of flips can be done simultaneously if no pair of the flipping edges are sides of the same triangle. In that sense, these are independent from each other. Based on this observation, for a triangulation T, if we flip a set E of triangulation edges in T and obtain a new triangulation T', this would be called a ***simultaneous flip***. Hurato et al. also showed that there are always $\lceil \frac{n-4}{2} \rceil$ edges that can be flipped. Galtier et al. [16] showed that $O(n)$ simultaneous flips is sufficient to reconfigure one triangulation into another one. They provided an example of two triangulations that needed at least $\Omega(n)$ simultaneous flips in order to transform one into the other one. They also showed that there are always $\lceil \frac{n-4}{6} \rceil$ edges that can be flipped simultaneously, and provided an example where the maximum number of edges that can be flipped simultaneously is $\lceil \frac{n-4}{5} \rceil$.

Variations: Edge-Labeled Flips Another variation on flips in triangulations is to consider edge-labelled triangulations, as introduced by Bose et al. [7]. Here the edges of a triangulation have distinct labels and a flip transfers the label from the edge that is removed to the edge that is added. Bose et al. proved a connectivity and diameter result for points in convex position—any labelled triangulation can be flipped to any other with $O(n \log n)$ flips. For general point sets, the labelled flip graph is not connected, but Lubiw et al. [27] characterized when one labelled triangulation can be flipped to another. The proof uses the *flip complex*, which is an abstract complex (i.e., a complex which is defined combinatorially rather than geometrically) that generalizes the associahedron. Some properties of the associahedron carry over to the flip complex, e.g., the two dimensional faces have size 4 or 5.

Connectivity and Expander Properties Recently, Wagner and Welzl [38] showed that for n points in general position in the plane, the flip graph is $\lceil \frac{n}{2} - 2 \rceil$-connected. More generally, researchers who study connectivity properties of flip graphs are interested in mixing rate results. Wagner and Welzl also proposed the question of whether their connectivity results are going to be helpful for rapid mixing of triangulations and expander properties of the triangulation flip graph. An open frontier in the study of flip graphs and triangulation mixing rate has to do with expander properties. The reason for these studies is that expander properties would potentially be helpful for rapid mixing via random flips. Mixing time of triangulations has been studied in [30, 29]. Molloy et al. [30] gave a lower bound of $\Omega(n^{3/2})$ and an upper bound of $O(n^{23} \log(n/\epsilon))$. McShine and Tetali [29] improved the upper bound to $O(n^5 \log(n/\epsilon))$. Eppstein and Frishberg [15] showed that mixing time for the flip walk on a convex point set is $O(n^{4.75})$. Caputo et al. [8] studied rapid mixing of lattice triangulations, which has important applications in quantum physics. Lattice triangulations are triangulations on a *grid point set*. It has recently been shown that the reconfiguration graph of bases of a matroid is an expander [2], and it would be interesting to know if similar results hold for triangulation flip graphs.

Convex Point Sets and Associahedra

As mentioned in the previous chapter, the flip graph for n points in convex position can be realized geometrically as the 1-skeleton of an $(n-3)$-dimensional polytope called the associahedron. This means that each vertex of the associahedron represents one triangulation of the point set and each edge of the associahedron represents a flip. The $(n-3)$-dimensional associahedron is denoted $\mathcal{A}_{n-1}$. Using the fact that the flip graph of convex point sets is the

1-skeleton of associahedra, and applying Balinski's theorem [3] to this 1-skeleton Wagner and Welzl [38] showed that for points in convex position, the flip graph is $(n-3)$-connected. Another interesting fact about associahedra is that each vertex has $n-3$ adjacent edges, which follows from the fact that each triangulation of a convex point set has $n-3$ possible flips.

We showed that there exists a bijection between binary trees and triangulations in Section 1.3, which means there exists a bijection between rotations in trees and flips in triangulations. As a result, associahedra are the realizations of the reconfiguration graphs for rotations in binary trees. Also, it is known that there is a correspondence between each vertex of the associahedra and each different way of inserting parentheses in an expression of length n; then, each edge represents applying the associativity rule once. Many other realizations of associahedra have been discussed in [9].

The 3-dimensional associahedron $\mathcal{A}_5$ is the one that can be understood better than the higher dimensional ones because of its simplicity. As shown in Figure 1.3 it has 14 vertices, 21 edges, and 9 faces (2D facets). Also, it is interesting that each face corresponds to one of the inner diagonals of the convex hexagon, in the sense that the face consists of all the triangulations that contain that diagonal. More generally, each $(n-4)$-dimensional face of the $(n-3)$-dimensional associahedron consists of the triangulations that contain one specific inner diagonal of the convex point set of size n.

The diameter of associahedra has been studied widely in the past. Sleator et al. [35] showed that the diameter of $\mathcal{A}_{n-1}$ is $2n-6$ for large enough n. Equivalently, maximum rotation distance for trees of size n is $2n-6$ for large n's. Later, Pournin [33] proved this for any n larger than 9. This means that for a convex point set with n points, the diameter of the flip graph is $2n-10$ for $n > 12$. However, a main open question is the complexity of the flip distance problem for convex point sets (is it NP-hard or in P?).

Chapter 3

Flip Cut Edges

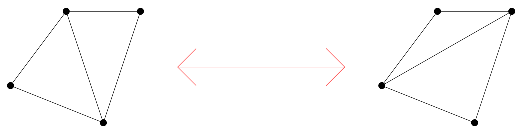
In this chapter, we explain the initial steps in identifying flip cut edges (Section 3.1), then we characterize flip cut edges (Section 3.2) and give an $O(n \log n)$ time algorithm to test if a given edge is a flip cut edge (Section 3.3). In Section 3.4 we examine some special point sets and establish bounds on the number of flip cut edges and the number of connected components.

3.1 Initial Steps

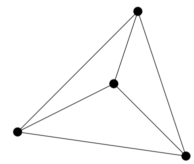
This section contains an analysis of flip cut edges in small point sets, and a discussion of the role of EC4's (empty convex quadrilaterals) and EC5's in characterizing flip cut edges. This material adds intuition but is not necessary for the following sections.

3.1.1 The Simplest Example of a Flip Cut Edge

The main problem at the beginning is whether it is possible to disconnect the flip graph by forbidding exactly one of the edges between a pair of points (which we call a ***flip cut edge***). In order to solve this problem, one should begin with some simple examples and test whether an edge with the described property can be found. Any point set containing three points only has one valid triangulation, which can be constructed by creating a triangle that contains the points as its vertices. Therefore, the flip graph contains only one vertex and cannot become disconnected in any way. Remember that if there exists a valid flip for a triangulation, we need a convex empty quadrilateral where the flip is exchanging its



(a) The flip graph of a convex point set of size four



(b) The flip graph of a point set with size four whose convex hull is a triangle contains only one frozen triangulation

Figure 3.1: Flip graphs of point sets with size four

diagonals. So, a point set containing four points that create an EC4 is a point set with two valid triangulations (Figure 3.1a). First, note that we cannot forbid any edge of the convex hull since every valid triangulation contains all of them. Thus, in the previous case, we only can forbid one of the diagonals of the convex quadrilateral. This leads to a new flip graph with exactly one vertex which is connected. In a point set P, $|P| = 4$ where the convex hull is a triangle and there exists a point inside this triangle, no flips are possible, so we have a frozen triangulation and an isolated vertex in the flip graph which again is not possible to become disconnected (Figure 3.1b). This means no point set of size less than five can have a flip cut edge. So, we are going to consider point sets of size five.

The first option is a convex point set of size five. Remember that a point set is in convex position if all the points are vertices of the convex hull of that point set. The flip graph for this point set contains a cycle of size five (Figure 3.2). By forbidding any single edge of this point set, two triangulations become invalid and a path of size three will be the new flip graph. So, again, it is not possible to disconnect the flip graph. Now, consider in our point set P, $|P| = 5$, the convex hull is a convex quadrilateral, and the other point is inside this quadrilateral. In, this scenario, if you have colinear points, then again, the flip graph only contains an isolated vertex, and if it does not, there exists only one valid flip which means disconnecting it is not possible (Figure 3.3a). The last possibility for $|P| = 5$ is a point set with a triangle convex hull and two points inside the hull. In this case, again, there exists only one valid flip (Figure 3.3b) and disconnecting the flip graph is impossible. As a result, we move on to point sets of size six.

In the previous chapters, we saw that the flip graph for a convex point set of size six is a 3D associahedron (Figure 1.3). We also saw that by forbidding one edge of this point set, we remove triangulations on one of the 2D faces of this associahedron. Thus, forbidding one edge does not disconnect the flip graph. The second possibility is having a pentagon convex hull and a point inside. For this case, the flip graph is shown in Figure 3.4. Again, there is no edge such that forbidding the edge disconnects the flip graph. Finally, consider

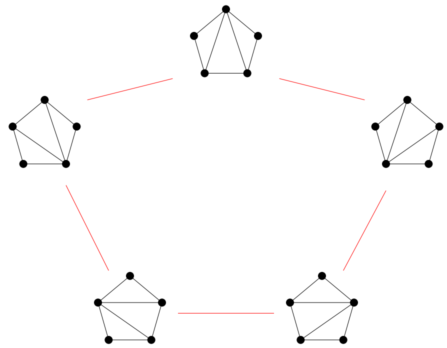
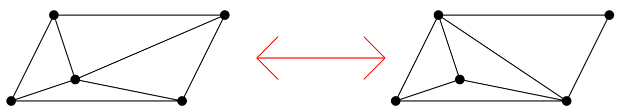
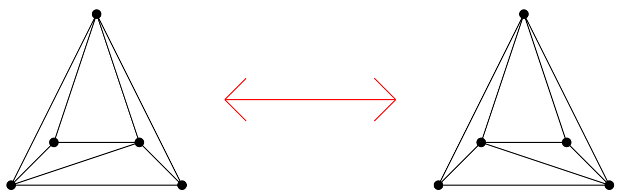


Figure 3.2: The flip graph for a pentagon point set



(a) Flip graph of a point set with size five whose convex hull is a quadrilateral



(b) Flip graph of a point set with size five whose convex hull is a triangle

Figure 3.3: Flip graph of point sets with size five

a point set with a quadrilateral as its convex hull which contains two points inside the hull. The flip graph is shown in Figure 3.5; by forbidding the edge between the two points inside the hull, the middle triangulations in the flip graph become invalid and we are left with two isolated vertices in the resulting flip graph. This means that we have a flip cut edge in this example, and since we have shown there exist no flip cut edges in simpler examples, this is the smallest possible example.

3.1.2 The role of EC4's and EC5's

In this section we investigate the relationship between flip cut edges and EC4's (empty convex quadrilaterals) and EC5's. We make a conjecture about EC4's which we disprove, and a conjecture about EC5's which we later show is correct.

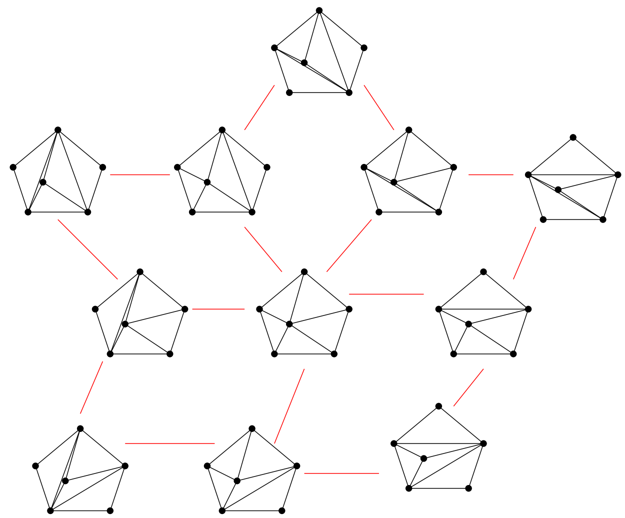


Figure 3.4: The flip graph for a point set with size six whose convex hull is a pentagon

Observation 1. *In Figure 3.5, each of the two disconnected triangulations contains one of the possible convex quadrilaterals that includes the forbidden edge as its diagonal.*

This observation leads us to the thinking that maybe the ***quadrilateral graph*** introduced by Eppstein [14] can help us in identifying flip cut edges. Recall that in a ***quadrilateral graph (QG)*** of point set P, there is a vertex for each edge between two points of the point set and there exists a graph edge between those two vertices if the four endpoints of the corresponding edges create a convex empty quadrilateral (EC4). In other words, suppose by removing the forbidden edge in QG we disconnect two edges in QG that cross the forbidden edge. Then, does it mean that forbidding that edge disconnects the flip graph? The point set in Figure 3.5 is an example that gives a positive answer to this question. The QG for this point set is shown in Figure 3.6. After removing vertex uv, a_1b_1 and a_2b_2 become disconnected in the QG.

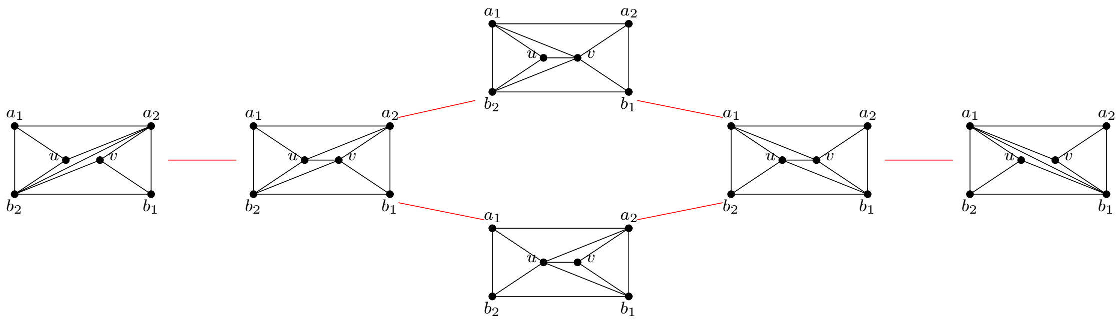


Figure 3.5: The flip graph for a point set which gives us the smallest example that contains a flip cut edge: by forbidding the edge between two points inside the quadrilateral, the resulting flip graph only contains the leftmost and rightmost triangulations, which are disconnected from each other.

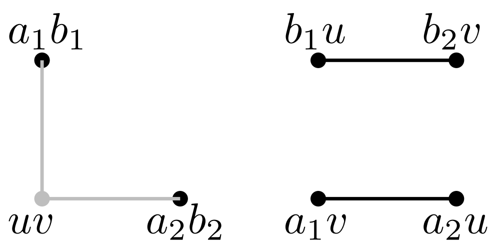
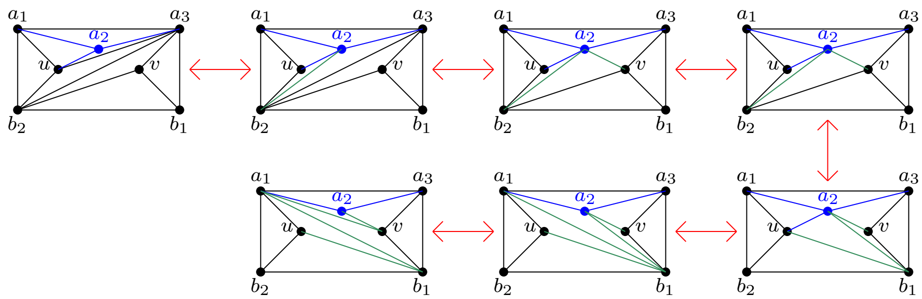


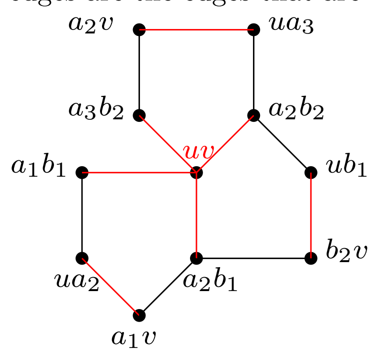
Figure 3.6: The QG for the smallest example in Figure 3.5: forbidding uv disconnects a_1b_1 and a_2b_2

Conjecture 2. *Suppose that the forbidden edge $e = uv$ is contained in two EC4's, Q_1 and Q_2 with diagonals ab and cd respectively. Suppose that removing uv from the QG disconnects ab from cd. Then, in the flip graph $\mathcal{F}_{-e}$, every triangulation containing $Q_1 \cup ab$ is disconnected from every triangulation containing $Q_2 \cup cd$.*

Unfortunately, this conjecture is FALSE. The point set shown in Figure 3.7a is a counter-example for the conjecture. The QG of this point set becomes disconnected after removing edge uv (After forbidding uv, we need to remove vertex uv and red edges in the QG that corresponds to invalid flips as shown in Figure 3.7b). But, the flip graph is not disconnected as we can transform one of the triangulations containing one of the quadrilaterals to the triangulation that contains the other quadrilateral. In more detail, forbidding uv in the QG disconnects a_1b_1 from a_3b_2, so the intuition was that we might be unable to flip from a triangulation containing a_1b_1 to a triangulation containing a_3b_2. However, although we cannot flip directly between those two edges, the example flip sequence shows that a_3b_2 flips to position va_2, and a_1b_1 flips from position ua_2.



(a) The flip graph of the simplest point set with flip cut edge after adding point a_2. Blue edges are added after adding a_2. Green edges are the edges that are created during the flip operations.



(b) The QG of the point set in Figure 3.7a. Red edges correspond to invalid flips after forbidding uv.

Figure 3.7: An example where removing uv disconnects a_1b_1 and a_3b_2 in the QG (b) but triangulations containing those diagonals are still connected in the flip graph (a).

We need a more subtle way to analyze how one EC4 is transformed to another one. To do this, we will examine EC5's. We first revisit the flip sequence of Figure 3.7a and then formulate several refined conjectures. Figure 3.7a shows that in the path from transforming the first quadrilateral to the second one, we use flips inside three intermediary pentagons: $ua_2a_3vb_2, ua_2vb_1b_2, ua_1a_2vb_1$. So, it shows adding a_2 creates some EC5's inside of which we can flip in a way that enables us to transform EC4 ua_3vb_2 in Figure 1.4 to EC4 ua_1vb_1. So, we conjectured that EC4's containing the forbidden edge are fundamental elements for identifying whether forbidding that edge disconnects the flip graph and EC5's are the bridges that help us transform each of the EC4's to the other ones using a sequence of flips occurring inside the EC5's. The formal description of the conjecture is as follows:

Conjecture 3. *Edge $e = uv$ is a flip cut edge iff there exist two EC4's Q_1 and Q_2 containing e in a way that there does not exist any other point p where $p \cup Q_1$ and $p \cup Q_2$ are both EC5's.*

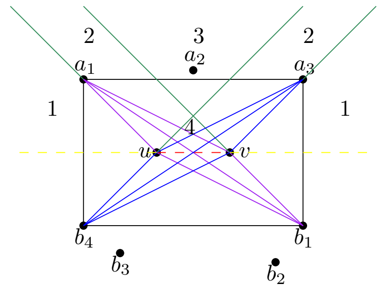
We are going to DISPROVE this conjecture, though it seems promising based on Figures 3.5 and 3.8. Adding a new point to the point set in Figure 3.5 in different areas acknowledges our conjecture; we draw the extension of the edges of the two EC4's (Figure 3.8a). After adding one point to each of areas 1, 2, 3, and 4, we see that only adding a point to area 4, prevents edge uv from being a flip cut edge. And, that is because only a point in area 4 can create EC5's with the vertices of both ua_1vb_1 and ua_3vb_2. Note that since the picture is symmetric, the same areas are created on the bottom half of the point set. However, checking only area 4 for an additional point is not enough for identifying whether uv is a flip cut edge.

In Figure 3.8a some points are added but not to the area 4. So, this point set satisfies the condition in Conjecture 3. However, Figure 3.8b shows how we can transform EC4 ua_1vb_1 to EC4 ua_3vb_4 using flips inside the pentagons shown step by step. More specifically, first we use EC5 $ua_1vb_1b_2$ to transform EC4 ua_1vb_1 to EC4 ua_1vb_2; then, use flips inside $ua_1a_2vb_2$ to transform ua_1vb_2 to ua_2vb_2. We continue by using $ua_2vb_2b_3$ to transform ua_2vb_2 to ua_2vb_3. $ua_1a_2vb_2$ is used in the same way to transform ua_1vb_2 to ua_2vb_2. Finally, we flip inside $ua_3vb_3b_4$ to transform ua_3vb_3 to ua_3vb_4. Figure 3.9 shows how to use an EC5 to flip one EC4 to another one even with having a forbidden edge. A careful analysis of the flip graph of the point set in Figure 3.8a shows that it is connected. So, instead of the previous conjecture, we propose a new one that supports our observation on Figure 3.8 to characterize flip cut edges. In order to do that, first, we introduce a new graph called *pentagon graph*.

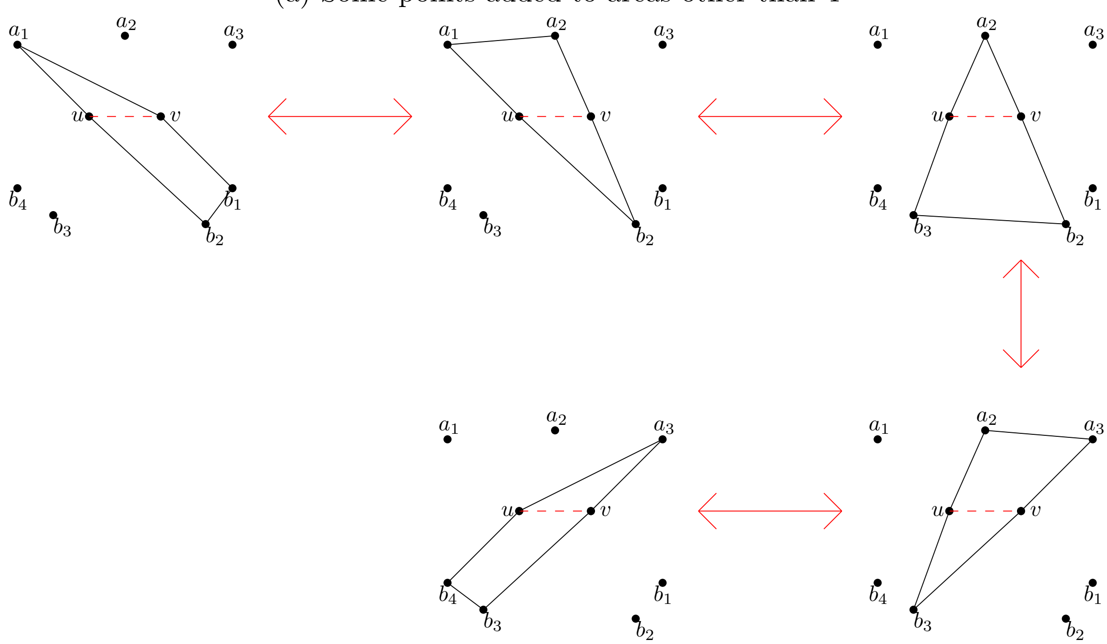
Definition 4. *For an edge $e = uv$ in the point set P, the* ***pentagon graph (PG)*** *of e can be constructed in the following way: every EC4 containing e as a diagonal is a vertex. There exists an edge between two vertices if there exists an EC5 that contains both corresponding EC4's inside it.*

The next conjecture relates flip cut edges to the PG. Later, in Section 3.3.2, we will prove this conjecture to be CORRECT (Lemma 13).

Conjecture 5. *Edge e is a flip cut edge iff its pentagon graph is disconnected.*



(a) Some points added to areas other than 4



(b) How to use EC5's to transform EC4 ua_1vb_1 to EC4 ua_3vb_4

Figure 3.8: Different areas for adding an additional point to point set in Figure 3.5 to see the effect on the forbidden edge remaining a flip cut edge

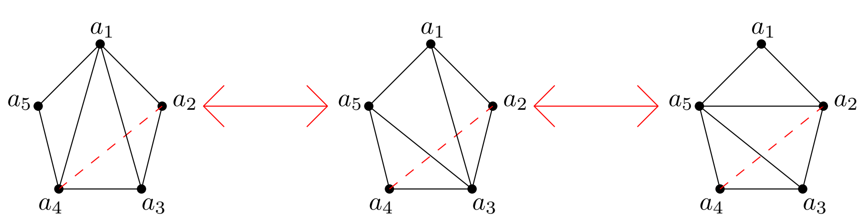


Figure 3.9: Transforming EC4 $a_1a_2a_3a_4$ to $a_2a_3a_4a_5$ using EC5 $a_1a_2a_3a_4a_5$

3.2 Characterizing Flip Cut Edges

In this section we give three characterizations of flip cut edges. The characterizations are in terms of connectivity of associated graphs that we call G_Y, G_Z, and PG. The pentagon graph (PG) was introduced in Section 3.1.2, and the other two will be defined below. The first characterization helps identifying connected components better, the second one is helpful for having an efficient algorithm, and the characterization based on PG more precisely demonstrates the transition between triangulations using flips.

3.2.1 Main Characterization, G_Y

Consider an edge $e = uv$ of P. Let us orient e horizontally so we can use the terms "above" and "below" to refer to the two sides of e. In the horizontal orientation, suppose that u lies to the left of v.

Let Y be the set of edges f of P such that f crosses e. Let G_Y be the line graph of Y, i.e., the vertex set of G_Y is Y, and two edges of Y are adjacent in G_Y if they meet at a point. See Figure 3.10.

We will prove that e is a flip cut edge if and only if G_Y is disconnected. In fact, we will be able to identify the connected components of $\mathcal{F}_{-e}$ from G_Y.

Observation 6. *$\mathcal{T}_{-e}$ is the union of the sets of triangulations $\mathcal{T}_{+f}$ for $f \in Y$. Each flip graph $\mathcal{F}_{+f}$ is connected.*

Proof. For any edge f, $\mathcal{F}_{+f}$ is connected by properties of constrained triangulations [10]. For $f \in Y$, $\mathcal{T}_{+f}$ is a subset of $\mathcal{T}_{-e}$ because triangulations that contain f cannot contain e. Finally, since every triangulation of $\mathcal{T}_{-e}$ contains some edge of Y, $\mathcal{T}_{-e}$ is the union of $\mathcal{T}_{+f}$ for $f \in Y$. □

The observation says that every vertex of the graph $\mathcal{F}_{-e}$ appears in some $\mathcal{F}_{+f}$ for $f \in Y$. In fact, every edge of the graph $\mathcal{F}_{-e}$ appears in some $\mathcal{F}_{+f}$ for $f \in Y$, as we will prove as part of Theorem 8.

Based on Observation 6, in order to identify connected components of the flip graph $\mathcal{F}_{-e}$, it suffices to figure out which subgraphs $\mathcal{F}_{+f}$ are connected to which other ones in $\mathcal{F}_{-e}$, i.e., to know when there is a path in $\mathcal{F}_{-e}$ from an element of $\mathcal{T}_{+f}$ to an element of $\mathcal{T}_{+g}$.

Before giving the main theorem, we give one more observation.

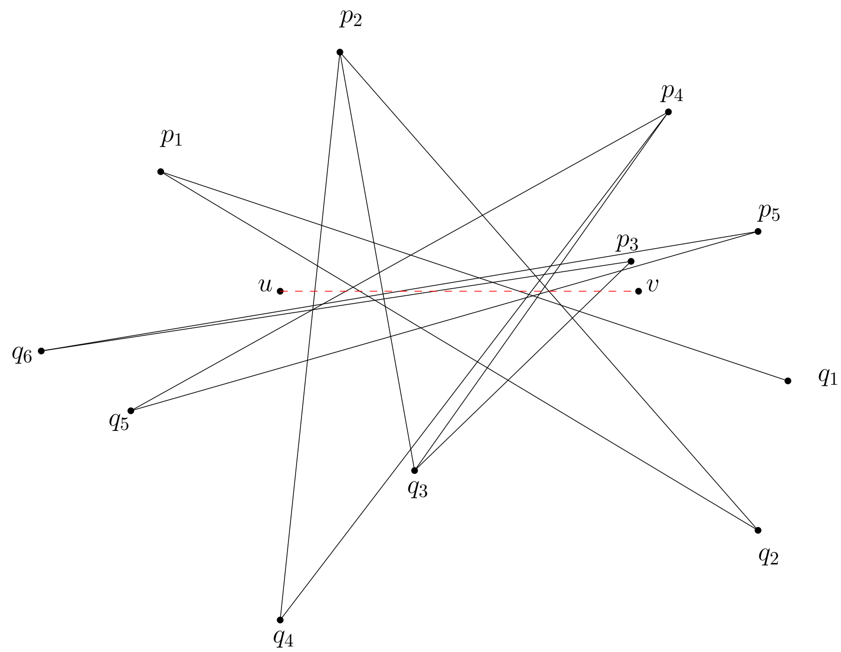


Figure 3.10: The edges Y that cross $e = uv$. Here, G_Y is connected, so e is not a flip cut edge.

Observation 7. *For any triangulation T in $\mathcal{T}_{-e}$, the edges of Y in T, which we denote $Y \cap T$, are connected in G_Y.*

Proof. The triangles of T that cross the segment uv form a path in the planar dual of the triangulation, and so the edges of $Y \cap T$ which are the side edges of these triangles form a connected subgraph of G_Y. See Figure 3.11. □

Theorem 8. *Edge e is a flip cut edge if and only if G_Y is disconnected. More specifically, edges f and g in Y are connected in G_Y if and only if $\mathcal{T}_{+f}$ and $\mathcal{T}_{+g}$ are connected in $\mathcal{F}_{-e}$.*

Proof. If fg is an edge of G_Y, then, in the point set, the edges f and g are incident at a common endpoint so they do not cross, which implies that there is a triangulation containing f and g, so $\mathcal{F}_{+f}$ and $\mathcal{F}_{+g}$ are connected (they have a triangulation in common). Thus, by transitivity, if f and g are connected in G_Y then $\mathcal{F}_{+f}$ and $\mathcal{F}_{+g}$ are connected.

For the converse, if $\mathcal{F}_{+f}$ and $\mathcal{F}_{+g}$ are connected, this means that there exists a sequence of flips that transforms triangulation $T_1 \in \mathcal{T}_{+f}$ to $T_2 \in \mathcal{T}_{+g}$ where intermediary triangulations are also in $\mathcal{T}_{-e}$. So, it suffices to show that if we flip from T_1 in $\mathcal{T}_{-e}$ to T_2 in $\mathcal{T}_{-e}$,

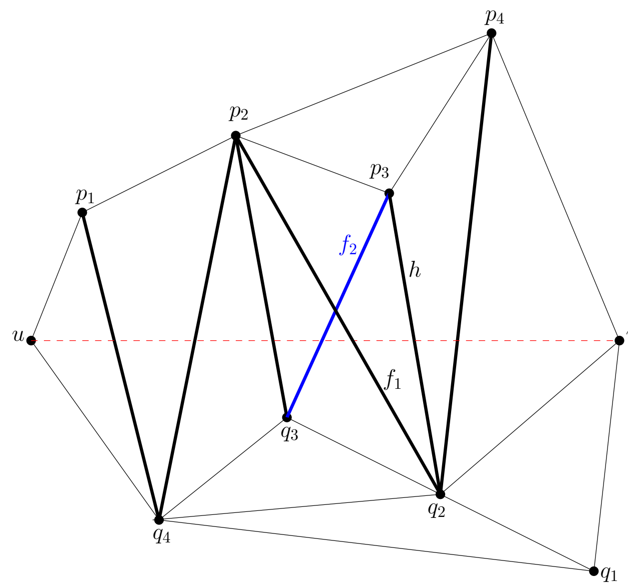


Figure 3.11: The edges of $Y \cap T$ (in thick black) are connected in G_Y. A flip of f_1 to f_2 requires another edge $h \in Y \cap T_1 \cap T_2$.

then $Y \cap T_1$ and $Y \cap T_2$ are connected in G_Y. This implies that if there is a sequence of flips transforming T_1 in $\mathcal{T}_{-e}$ to T_2 in $\mathcal{T}_{-e}$, then there is a path connecting f to g in G_Y. First note that by Observation 7, $Y \cap T_1$ is a connected set in G_Y and the same is true for $Y \cap T_2$. We will show that $Y \cap T_1$ and $Y \cap T_2$ have an element in common. Let $f_i \in T_i \cap Y$, $i = 1, 2$. If either edge is in the other triangulation, we are done. Otherwise, the flip from T_1 to T_2 must flip f_1 to f_2. Then the EC4 formed by the endpoints of f_1 and f_2 has a side edge h that is an edge of Y in both T_1 and T_2. See Figure 3.11. (Note that this argument shows that every edge of the flip graph $\mathcal{F}_{-e}$ lies in $\mathcal{F}_{+h}$ for some $h \in Y$.) $\square$

Algorithmic implications. The theorem gives an immediate polynomial time algorithm to test if e is a flip cut edge: construct G_Y (a graph on $O(n^2)$ vertices) and test connectivity. The details of this algorithm can be found in Algorithm 1 in Section 3.3.1. In Section 3.3.3 we give a faster $O(n \log n)$ time algorithm.

Our other algorithmic goal is to "identify" the connected components of $\mathcal{F}_{-e}$. Although the flip graph is exponentially large, we can identify connected components by grouping them into $\mathcal{T}_{+f}$'s for $f \in Y$, and identify which groups of triangulations are connected to which other ones. That being said, we focus on the problem of testing whether two triangulations T_1 and T_2 in $\mathcal{T}_{-e}$ are in the same connected component of $\mathcal{F}_{-e}$. Using

Theorem 8, we can do that as follows. Pick f_1 in $Y \cap T_1$ and f_2 in $Y \cap T_2$ (note that such edges exist). Then $T_1 \in \mathcal{T}_{+f_1}$ and $T_2 \in \mathcal{T}_{+f_2}$. So T_1 and T_2 are connected in $\mathcal{F}_{-e}$ iff $\mathcal{T}_{+f_1}$ and $\mathcal{T}_{+f_2}$ are connected in $\mathcal{F}_{-e}$ iff (by Theorem 8) f_1 and f_2 are connected in G_Y, which we can test in polynomial time. In Subsection 3.3 we give a faster $O(n)$ time algorithm.

3.2.2 Second Characterization, G_Z.

For the efficient algorithms in Section 3.3.3, we need an alternative characterization of flip cut edges in terms of a subgraph of G_Y.

For any $f \in Y$, let $Q(f)$ be the convex quadrilateral formed by the endpoints of f and $e = uv$. Let Z be the set of edges $f \in Y$ such that $Q(f)$ is an EC4. Let G_Z be the line graph of Z. Then G_Z is an induced subgraph of G_Y with a smaller vertex set.

Note that any $f \in Z$ has one point above e and one point below e, and these points make empty triangles with e. Let A be the set of points a above e such that auv is an empty triangle (an EC3). Let B be the set of points b below e such that bvu is an empty triangle. Thus Z consist of the edges from A to B that cross $e = uv$, i.e., $Z = Y \cap (A \times B)$. See Figure 3.13.

We will prove an analogue of Theorem 8:

Theorem 9. *Edge e is a flip cut edge if and only if G_Z is disconnected.*

We will also be able to characterize connected components of $\mathcal{F}_{-e}$ in terms of G_Z, but this cannot be analogous to Theorem 8, because not every triangulation contains an edge of Z. We begin with some preliminary results.

Lemma 10. *Edges $f, g \in Z$ are connected in G_Z if and only if they are connected in G_Y.*

Proof. The forward direction is clear since G_Z is an induced subraph of G_Y.

For the other direction, suppose f and g are connected in G_Y. Suppose, $f = f_1, f_2, ...,$ $f_m = g$ is a shortest path in G_Y between f and g. If all the f_i's are in Z, then f and g are connected in G_Z. Otherwise, we will modify the path to replace edges of Y by edges of Z. Suppose $f_i = p_i q_i$ with p_i above uv and q_i below uv. Because the path is shortest, the common points between successive edges are alternately above and below the line L through uv. In particular, suppose without loss of generality that $p_1 = p_2, q_2 = q_3, \ldots, p_{2i-1} = p_{2i}, q_{2i} = q_{2i+1}, \ldots$

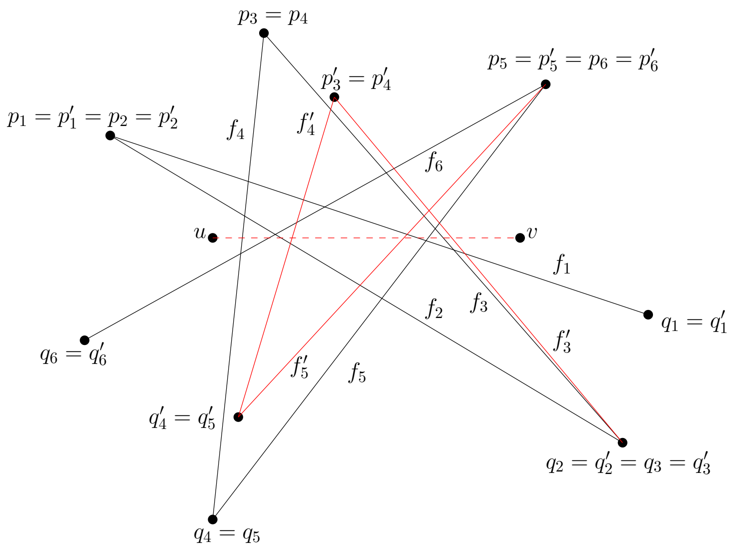


Figure 3.12: Proof of Lemma 10. The original path in G_Y from $f_1 \in Z$ to $f_6 \in Z$ is $f_1, \ldots, f_6$ (shown in black), and the modified path replaces the edges of $Y - Z$, which are f_3, f_4, f_5 by the edges $f_3', f_4', f_5' \in Z$ (shown in red).

The plan is to replace p_i by some point p_i' and replace q_i by some point q_i' so that the segments $f_i' = p_i'q_i'$ are edges in Z and form a path connecting f to g.

If triangle p_iuv is empty, then $p_i' = p_i$; otherwise p_i' is a point inside triangle p_iuv that is closest to line L. Define q_i' similarly with respect to triangle q_iuv. Note that $f_1' = f_1$ and $f_m' = f_m$.

We claim that each f_i' is in Z. We must show that the line segment $p_i'q_i'$ is an edge, that it crosses $e = uv$, and that the quadrilateral $Q(f_i')$ is empty.

First note that triangles $p_i'uv$ and $q_i'uv$ are empty because we picked p_i' and q_i' closest to line L. Next we claim that these two empty triangles together form a convex quadrilateral. This is because f_i crosses uv, so $Q(f_i)$ is convex, and when we move p_i to p_i' and q_i to q_i', convexity is preserved. Thus f_i' is an edge of Z.

Finally, note that $p_{2i-1}' = p_{2i}'$ and $q_{2i}' = q_{2i+1}'$ because that was true of the original points. Thus the edges f_i' form a path in G_Z connecting f and g. □

Lemma 11. *Every connected component of G_Y contains an edge of Z.*

This lemma can be proved in several different ways. One possibility is to take an edge $f \in Y$ and construct the edge $f' \in Z$ inside $Q(f)$ as in the above proof. However, showing that f and f' are connected in G_Y runs into some complications due to the possibility of collinear points.

Instead we take an edge $f \in Y$ and consider a triangulation T containing f and show that T can flip in $\mathcal{F}_{-e}$ to a triangulation containing an edge $g \in Z$. Below we give a short proof based on this idea, and in Section 3.3 we give an alternative algorithmic proof that efficiently finds an edge of Z connected to f in G_Y.

Proof. Let f be an edge of Y and let T be a triangulation that contains f. We will prove there is an edge $g \in Z$ that is connected to f in G_Y. There is a flip sequence from T to a triangulation that contains $e = uv$. The last flip in this sequence must involve an edge g of Z flipping to e. Until the last flip, we are in one connected component of $\mathcal{F}_{-e}$, so, by Theorem 8, all the edges of Y in all the trianglulations in the sequence are in the same connected component of G_Y. Thus f and g are connected in G_Y. $\square$

Proof of Theorem 9. If e is a flip cut edge, then by Theorem 8, G_Y is disconnected. By Lemma 11, every connected component of G_Y contains an edge of G_Z. Thus G_Z is disconnected.

In the other direction, if G_Z is disconnected, then by Lemma 10, G_Y is disconnected, so e is a flip cut edge by Theorem 8. $\square$

In Section 3.3.3 we give efficient algorithm that uses G_Z to test if edge e is a flip cut edge, and, if so, to identify when two given triangulations are in different components of $\mathcal{F}_{-e}$.

3.2.3 Third Characterization, PG.

The third characterization says that e is a flip cut edge if and only if PG is disconnected. This was stated as Conjecture 5 and is proved below as Lemma 13 using Theorem 9.

Recall from Definition 4 that PG has a vertex for every EC4 that has e as a diagonal and an edge when two EC4's are contained in an EC5.

Claim 12. PG is a subgraph of G_Z with the same vertex set.

Proof. A vertex of PG corresponds to an EC4 that has e as a diagonal, and the other diagonal of such an EC4 is an edge of Z. This gives a one-to-one correspondence between the vertices of PG and the set Z.

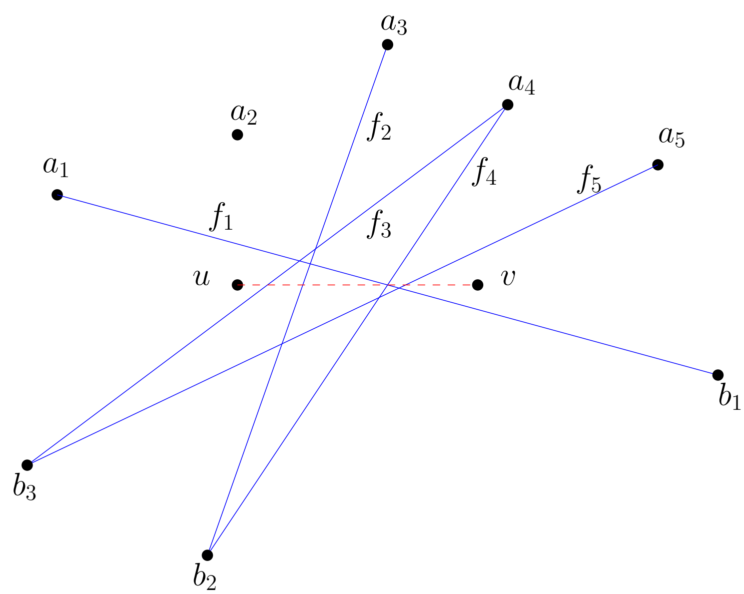


Figure 3.13: The points that form empty triangles with uv are $A = a_1, \ldots, a_5$ and $B = b_1, b_2, b_3$ ordered cyclically around u. The edges of Z are $f_1, \ldots, f_5$ and they form two connected components in G_Z, $\{f_1\}$ and $\{f_2, \ldots, f_5\}$. For any point $b \in B$, the points $a \in A$ such that $ab \in Z$ form a subinterval of A.

To show that any edge of PG is an edge of G_Z, consider two vertices of PG corresponding to edges $f, g \in Z$. Then $Q(f)$ and $Q(g)$ are EC4's that are contained in a common EC5. This implies that f and g share a common endpoint, so they are joined by an edge in G_Z. □

Lemma 13. *Edge $e = uv$ is a flip cut edge iff its pentagon graph PG is disconnected.*

Proof. Based on Theorem 9, e is a flip cut edge iff G_Z is disconnected. So, we need to prove PG is disconnected iff G_Z is disconnected, i.e., PG is connected iff G_Z is connected.

Based on the above claim, PG is a subgraph of G_Z on the same vertex set. Thus, the forward direction is trivial.

For the other direction, it suffices to show that if f and g are joined by an edge in G_Z then their corresponding vertices ($Q(f)$ and $Q(g)$) are connected in PG. The existence of an edge between f and g means they have a common endpoint. Suppose, without loss of generality, this endpoint is b below $e = uv$, and $f = ab$ and $g = a'b$. Suppose $a = a_0, a_1, ..., a' = a_l$ are the points above uv that create an empty triangle with u and

v in cyclic order around u. Note that a and a' create an empty triangle with u and v because $f = ab$ and $g = a'b$ are in Z. Because of the conditions about order and having empty triangles, we can conclude that $a_i ubv$, $a_{i+1}ubv$, and $ua_ia_{i+1}vb$ are all empty for $i = 0, ..., l-1$. Hence, there is an edge between every a_iubv, $a_{i+1}ubv$ in PG. Thus, $Q(f)$ and $Q(g)$ are connected in PG. $\square$

3.3 Algorithms for Flip Cut Edges

In this section we give three algorithms to test if an edge is a flip cut edge using the three characterizations from the previous section. One is easy to describe, one led to our implementation, and one is efficient. In more detail:

1. Algorithm 1 is straight-forward. It tests if G_Y is connected, as justified by Theorem 8. The running time is $O(n^4)$.
2. Algorithm 2 uses the simplest graph which is PG (it contains fewer vertices and edges in comparison with G_Y and G_Z) and checks if it is connected. It also is a better representation of flips for transforming triangulations containing different quadrilaterals. Algorithm 2 is the one we used for the examples in Figure 1.8. The reason we implemented this one at first was that Conjecture 5 was our first idea of characterizing flip cut edges and it also is a better demonstration of the way we can transform between triangulations containing EC4s created with u and v. The running time is $O(n^4)$.
3. Algorithm 3 tests if G_Z is connected (justified by Theorem 9) with an efficient $O(n \log n)$ time implementation.

Algorithms 1, 2, and 3 are in the first three subsections. In Section 3.3.4, we show how to find all flip cut edges in time $O(n^3)$ (note that there may be $\Theta(n^2)$ flip cut edges). In Section 3.3.5, with Algorithm 3 as a preprocessing step, we give an $O(n)$ time algorithm to test if two triangulations in $\mathcal{T}_{-e}$ are connected in $\mathcal{F}_{-e}$.

3.3.1 Testing for a Flip Cut Edge Using G_Y

Recall that Theorem 8 states:

Theorem 8. *Edge e is a flip cut edge if and only if G_Y is disconnected. More specifically, edges f and g in Y are connected in G_Y if and only if $\mathcal{T}_{+f}$ and $\mathcal{T}_{+g}$ are connected in $\mathcal{F}_{-e}$.*

In order to use this characteristic in implementing an algorithm, we build the graph G_Y and check whether it is connected or not. Note that the vertices of G_Y are edges in point set P that cross the edge $e = uv$ (the one we are testing). So, we can have $\Theta(n^2)$ vertices and $\Theta(n^4)$ edges. So, running DFS or BFS to check whether G_Y is connected takes $O(n^4)$ time. Also, note that this algorithm implicitly identifies the connected components of the flip graph. Triangulations containing edges represented by vertices in connected components of G_Y are in the same connected component in the flip graph $\mathcal{F}_{-e}$. Algorithm 1 describes the detailed implementation of the explained algorithm.

Algorithm 1 Connected components of G_Y

Input: $u, v, p_1, \ldots, p_n$
Output: Connected components of G_Y in the form of $C_1, \ldots, C_k$

procedure TestFlipCutEdge1
  `flipCutEdge` $\leftarrow$ False
  $k \leftarrow 0$
  // Find Y
  $Y \leftarrow \emptyset$
  for $i \leftarrow 1$ to n **do**
    for $j \leftarrow i + 1$ to n **do**
      if $p_i p_j$ crosses uv **then** add $p_i p_j$ to Y
    end for
  end for
  // Find the edges of G_Y
  for all v in Y, `adjacencyList`$[v] = \emptyset$, `visited`$[v] \leftarrow$ False
  for v_1 in Y **do**
    for v_2 in Y **do**
      if $v_1 \neq v_2$ and v_1, v_2 share a common endpoint **then**
        insert v_2 into `adjacencyList`$[v_1]$
      end if
    end for
  end for
  // Run DFS on G_Y to find connected components
  for $i \leftarrow 1$ to $Y.size()$ **do**
    if not `visited`$[v_i]$ **then**
      $k \leftarrow k + 1$
      $C_k \leftarrow \emptyset$
      run DFS on graph with Y as its set of vertices and `adjacencyList` representing the edges
      for every v visited during DFS, insert v into C_k
      for every v visited during DFS, `visited`$[v] \leftarrow$ True
    end if
  end for
  if $k > 1$ **then** `FlipCutEdge` $\leftarrow$ True
end procedure

3.3.2 Testing for a Flip Cut Edge Using PG

Since, in the beginning, we were looking for characterization based on Conjecture 5 and PG this was the first algorithm that we devised, thus this was the one we implemented. Also, note that the graph built based on this algorithm better represents the sequence of flips that transforms one triangulation into other ones. Algorithm 2 builds PG for e by finding EC5's and EC4's that contain diagonal e. Then, based on the connectivity of this graph, it decides whether e is a flip cut edge. We check every two points with u and v to see if they create a convex quadrilateral (this takes $O(n^2)$ time) and then we need a linear-time check to see whether that quadrilateral is empty. Similarly, for every set of three points, we check if they can create an empty convex pentagon with u and v, which takes $O(n^4)$ time. This algorithm works correctly as a result of Lemma 13. However, G_Z can be used for finding flip cut edges much more efficiently. This will be shown in the following section.

Algorithm 2 Connected components of PG

Input: $u, v, p_1, \ldots, p_n$

Output: Connected components of PG in the form of $C_1, \ldots, C_k$

```
procedure TestFlipCutEdge2
    flipCutEdge ← False
    k ← 0
    // Find V, the vertex set of PG
    V ← ∅
    for i ← 1 to n do
        for j ← i + 1 to n do
            if u, v, p_i, p_j create a convex quadrilateral containing uv as a diagonal then
                isempty ← True
                for l ← 1 to n do
                    if p_l is inside quadrilateral created by u, v, p_i, p_j then
                        isEmpty ← False
                        break
                    end if
                end for
                if isEmpty then
                    insert EC4 up_i vp_j to V
                    adjacencyList[up_i vp_j] = ∅, visited[up_i vp_j] ← False
                end if
            end if
        end for
    end for
    // Find the edges of PG
    for i ← 1 to n do
        for j ← i + 1 to n do
            if up_i vp_j ∈ V then
                for l ← j + 1 to n do
                    if u, v, p_i, p_j, p_l create a convex pentagon containing uv as a diagonal then
                        isempty ← True
```

```
            for t ← 1 to n do
              if p_t is inside pentagon created by u, v, p_i, p_j, p_l then
                isEmpty ← False
                break
              end if
            end for
            if isEmpty then
              if u p_i v p_l ∈ V then
                insert u p_i v p_j to adjacencyList[u p_i v p_l]
                insert u p_i v p_l to adjacencyList[u p_i v p_j]
              end if
              if u p_j v p_l ∈ V then
                insert u p_i v p_j to adjacencyList[u p_j v p_l]
                insert u p_j v p_l to adjacencyList[u p_i v p_j]
              end if
            end if
          end if
        end for
      end if
    end for
  end for
  // Run DFS to find connected components of PG
  for all v in V, visited[v] ← False
  for i ← 1 to V.size() do
    if not visited[V_i] then
      k ← k + 1
      C_k ← ∅
      run DFS on graph with V as its set of vertices and adjacencyList representing the edges
      for every v visited during DFS, insert v into C_k
      for every v visited during DFS, visited[v] ← True
    end if
  end for
  if k > 1 then
    FlipCutEdge ← True
  end if
end procedure
```

3.3.3 Efficiently Testing for a Flip Cut Edge Using G_Z

To test if edge e is a flip cut edge, by Theorem 9 it suffices to test if G_Z is connected. The algorithm also identifies the connected components of G_Z (though without explicitly listing the elements of Z, since there can be $\Theta(n^2)$ of them). In particular, we find disjoint subsets $A_1, \ldots, A_c$ of A, and $B_1, \ldots, B_c$ of B such that the ith connected component of G_Z consists of the edges of Z between A_i and B_i. Note that Z will not in general contain all pairs from $A_i \times B_i$.

We first need some more properties of the sets A and B and the graph G_Z. Because no empty triangle is contained in another, the ordering of $a \in A$ by decreasing (convex) angle $\angle auv$ is the same as the ordering by increasing (convex) angle $\angle avu$. Let $a_1, \ldots, a_k$ be this ordering of A. Similarly, let $b_1, \ldots, b_l$ be the ordering of B by decreasing (convex) angle $\angle bvu$, or equivalently, by increasing angle $\angle buv$. Thus the cyclic order of $A \cup B$ around u is $a_1, \ldots, a_k, b_1, \ldots, b_k$. See Figure 3.13.

Observation 15. *For any point b_j, the set of points a_i such that the edge $a_i b_j$ is in $Z(e)$ form a subinterval of the ordering of A, and similarly for a_i.*

Later on, we will find it useful to have an even stronger property:

Observation 16. *Let L be the line through uv.*

1. *If $a_i b_j$ crosses L to the right of v, then the same is true for all $a_{i'} b_{j'}$, $i' \geq i, j' \leq j$.*

2. *If $a_i b_j$ crosses L to the left of u, then the same is true for all $a_{i'} b_{j'}$, $i' \leq i, j' \geq j$.*

3. *If $a_{i_1} b_{j_2}$ and $a_{i_2} b_{j_1}$ are in Z for some $i_i \leq i_2$ and $j_1 \leq j_2$, then $a_i b_j$ is in Z for all $i_1 \leq i \leq i_2$ and all $j_1 \leq j \leq j_2$.*

Proof. Since (2) is symmetric to (1), it suffices to prove (1). To prove (1), suppose $a_i b_j$ crosses L to the right of v. Then the angle $\angle a_i v b_j$ is convex to the right of v, so (by the ordering) the same is true of $\angle a_{i+1} v b_j$ and $\angle a_i b_{j-1}$. The result follows by induction.

Also, note that (3) follows from (1) and (2). Suppose that for some $i, i_1 \leq i \leq i_2$ and $j, j_1 \leq j \leq j_2$, $a_i b_j$ is not in Z, then it either crosses L to the right of v or to the left of u. If $a_i b_j$ crosses L to the right of v, then based on (1), $a_{i_2} b_{j_1}$ is not in Z either, which is a contradiction. If $a_i b_j$ crosses L to the left of u, then based on (2) $a_{i_1} b_{j_2}$ is not in Z, which is a contradiction again. So, for any $i, i_1 \leq i \leq i_2$ and $j, j_1 \leq j \leq j_2$, $a_i b_j$ is in Z. $\square$

Finding the ordered sets A and B. We show how to find the ordered set A in $O(n \log n)$ time. Sort all the points $p \in P$ that lie above e by decreasing angle $\angle puv$, breaking ties by distance from u. Call this the *u-order*, $<_u$. Also, sort the same points by increasing angle $\angle pvu$, and call this the *v-order*, $<_v$. Define a_1 to be the first point in the v-order. Observe that a_1 is a member of A (the first member of A), and that any points p with $p <_u a_1$ do NOT belong to A and can be discarded. In general, we proceed through the v-order. Define a_i to be the next un-discarded element of the v-order, and discard all points p with $p <_u a_i$.

To prove that this is correct, observe that when we discard p, we have $p <_u a_i$ and $p >_v a_i$. Thus, the triangle puv contains a_i, so discarding p is correct. Observe that when we choose a_i, we have $a_{i-1} <_v a_i$ because we follow the v-order, and we have $a_{i-1} <_u a_i$ because we discarded all points $p <_u a_{i-1}$ in the previous step.

We can find the ordered set B similarly. The time to find the ordered sets A and B is $O(n \log n)$, and in fact, it is $O(n)$ after the preliminary sorting steps.

Finding the connected components of G_Z. We do not explicitly list the vertices of each connected component (there are too many). Rather, we find a partition of A into $A_1, \ldots, A_c$ and a partition of B into $B_1, \ldots, B_c$ such that the ith connected component of G_Z consists of the edges of G_Z contained in $A_i \times B_i$. Given the ordered lists A and B, we find the connected components of G_Z in linear time as follows. The algorithm has two phases. In Phase 1, we find the first edge of the next component, and in Phase 2, we complete the component. Phase 1 initially begins with a_1 and b_1, but more generally, we will find the first edge of the next component A_c, B_c among the "active" points $a_i, \ldots, a_k$ and $b_j, \ldots, b_\ell$, maintaining the invariant that there are no edges of Z from an inactive point to an active point. We search for an edge that crosses uv. Let L be the line through uv. If $a_i b_j$ crosses L to the right of v, then by Observation 16(1), there are no edges of Z from b_j to any active point. So we increment j and the invariant is maintained. If $a_i b_j$ crosses L to the left of u, then, with similar justification, we increment i. Continue until $a_i b_j$ crosses uv. This completes Phase 1—we have found the first edge of the connected component. Add a_i to A_c and add b_j to B_c.

For Phase 2, we complete the connected component by alternately "pivoting" on a_i and b_j. To pivot on a_i, update j to the maximum index such that $a_i b_j$ crosses uv, putting all the b points we find into B_c. To pivot on b_j, update i to the maximum index such that $a_i b_j$ crosses uv, putting all the a points we find into A_c. When no more pivots are possible, we are done with this connected component and done with Phase 2. We increment i and j by 1, and go back to Phase 1 to find the next connected component.

More details are given as Algorithm 3. The algorithm runs in linear time since it only performs a linear scan through each of the ordered lists A and B, doing constant work for each point.

We now justify correctness of the algorithm. We already argued that during Phase 1 we maintain the invariant that there are no edges of Z between an inactive point and an active point, i.e., there are no edges of Z of the form $a_{i'}b_{j'}$ with $i' \leq i$ and $j' > j$ or with $i' > i$ and $j' \leq j$. Now consider Phase 2. Observe that when we add points to A_c and B_c, they are part of the same connected component of G_Z. This is because when we pivot on a_i, all the b points that we add to B_c are adjacent to a_i in Z, and similarly, when we pivot on b_j, all the a points that we add to A_c are adjacent to b_j in Z. It remains to show that when we declare a connected component "done" because we can no longer pivot at a_i or b_j, then the invariant holds for active points $a_{i+1}, \ldots, a_k$ and $b_{j+1}, \ldots, b_\ell$, i.e., there are no edges of Z of the form $a_{i'}b_{j'}$ with $i' \leq i$ and $j' > j$ or with $i' > i$ and $j' \leq j$. First suppose $a_{i'}b_{j'} \in Z$ for some $i' \leq i$ and $j' > j$. Then by Observation 16(3), $a_i b_{j+1}$ is in Z, so we would not have been done pivoting at a_i. The case of $i' > i$ and $j' \leq j$ follows by symmetry.

3.3.4 Finding All Flip Cut Edges

To find all the flip cut edges, we run the above test on each of the $O(n^2)$ edges. We preprocess by sorting the points cyclically around each point p in a total of $O(n^2 \log n)$ time. Then, to test a particular edge $e = uv$, we find the ordered sets A and B around uv, and apply Algorithm 3, which takes linear time apart from the sorting. Thus the total time to find all flip cut edges is $O(n^3)$.

3.3.5 Testing Connectivity of Two Triangulations

We now show how to test if two triangulations T_1 and T_2 in $\mathcal{T}_{-e}$ are connected in $\mathcal{F}_{-e}$. We assume that we have the output of the above algorithm, i.e., the sets $A_1, \ldots, A_c$ and $B_1, \ldots, B_c$ such that the ith connected component of G_Z consists of the edges of Z between A_i and B_i. We can then find the component i of a given edge $e \in Z$ in constant time.

As mentioned in Section 3.2 one approach is to pick one edge f_1 from $Y \cap T_1$, and one edge f_2 from $Y \cap T_2$, and test if f_1 and f_2 are in the same connected component of G_Y. However, we only have G_Z available to us, and there are triangulations in $\mathcal{T}_{-e}$ that contain no edges of Z.

Algorithm 3 Connected components of G_Z

Input: $u, v, a_1, \ldots, a_k, b_1, \ldots, b_\ell$

Output: Connected components of G_Z in the form of $A_1, \ldots, A_c, B_1, \ldots, B_c$

procedure CONNECTEDCOMPONENTS
 for all i, `done[`a_i`]` $\leftarrow$ False; for all j, `done[`b_j`]` $\leftarrow$ False
 $i \leftarrow 1$; $j \leftarrow 1$; $c \leftarrow 0$
 while $i \leq k$ and $j \leq \ell$ **do**
  // Phase 1. Find the first edge of component c
  while $a_i b_j$ does not cross uv **do**
   if $a_i b_j$ crosses L to the right of v **then**
    $j \leftarrow j + 1$; **if** $j > \ell$ **then Halt**
   end if
   if $a_i b_j$ crosses L to the left of u **then**
    $i \leftarrow i + 1$; **if** $i > k$ **then Halt**
   end if
  end while
  $c \leftarrow c + 1$; Insert a_i into A_c, and b_j into B_c
  // Phase 2. Find all edges in component c by alternately pivoting on a_i, b_j until no further pivot is possible
  repeat
   // Pivot on a_i
   while $a_i b_j$ crosses uv and $j \leq \ell$ **do**
    insert b_j into B_c; $j \leftarrow j + 1$
   end while
   $j \leftarrow j - 1$; `done[`a_i`]` $\leftarrow$ `True`
   // Pivot on b_j
   while $a_i b_j$ crosses uv and $i \leq k$ **do**
    insert a_i into A_c; $i \leftarrow i + 1$
   end while
   $i \leftarrow i - 1$; `done[`b_j`]` $\leftarrow$ `True`
  until `done[`a_i`]` and `done[`b_j`]`
  $i \leftarrow i + 1$; $j \leftarrow j + 1$
 end while
end procedure

Instead, we give an algorithmic version of Lemma 11 that finds an edge $g_1 \in Z$ in the same component of G_Y as the edges $Y \cap T_1$, and an edge $g_2 \in Z$ in the same component of G_Y as the edges $Y \cap T_2$. Then we simply test if g_1 and g_2 are in the same set $A_i \times B_i$.

We first establish correctness and then give the details of finding g_1 and g_2 in linear time. We must prove that g_1 and g_2 are connected in G_Z if and only if T_1 and T_2 are connected in $\mathcal{F}_{-e}$. Let f_i be an edge of $Y \cap T_i$. By Observation 7, the edges of $Y \cap T_i$ are all connected in G_Y, so the choice of f_i is arbitrary. Now T_1 and T_2 are connected in $\mathcal{F}_{-e}$ iff (by Theorem 8) f_1 and f_2 are connected in G_Y iff (by choice of g_1, g_2) g_1 and g_2 are connected in G_Y iff (by Lemma 10) g_1 and g_2 are connected in G_Z.

We now give the details of finding g_1 (finding g_2 is similar). Triangulation T_1 has a sequence C of triangles that intersect uv. See Figure 3.14. Each triangle in C shares an edge of $Y \cap T_1$ with the previous triangle in C, and all the edges of $Y \cap T_1$ are in one connected component of G_Y (by Observation 7). Among the vertices of triangles of C, let a be a vertex above the line L through uv that is closest to L and let b be a vertex below the line L that is closest to L. Then ab is an edge of Z and is in the same connected component as $Y \cap T_1$. This can be done in linear time.

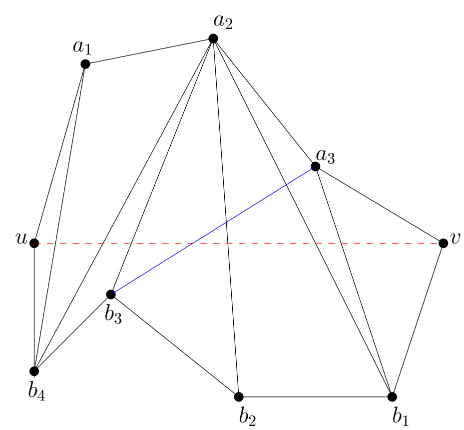


Figure 3.14: The triangles C of T_1 that cross edge $e = uv$, and an edge a_3b_3 of Z (in blue) that is connected in G_Y to the edges of $Y \cap T_1$.

3.4 Further Results on Flip Cut Edges

In this section we establish some bounds on the number of flip cut edges, and on the number of disconnected components caused by forbidding one flip cut edge. To do this, we fill in the details of the claims made in Section 1.4.1 (Examples) about some special point sets: channels, hourglasses, and grid point sets. We begin by relating flip cut edges to empty convex pentagons. Throughout the section we refer to a forbidden edge e, and to Z, G_Z, A, B, and so on, as defined above.

3.4.1 Empty Convex Pentagons (EC5's)

Empty convex pentagons (EC5's) play a significant role in the study of flip graphs. In Section 3.1.2 we showed how EC5's are used to flip between two EC4's without using the forbidden edge. Later, we proved how PG illustrates the role of EC5's in transforming different triangulations of $\mathcal{F}_{-e}$. EC5's have important roles in other works too. Eppstein [14] studied the flip graph of points without EC5's, and gave a polynomial time algorithm to find the minimum number of flips between two given triangulations. Points on an $n \times m$ grid have no EC5's, and in the other direction, a point set without EC5's must have collinear points. In particular, any $n \geq 10$ points in general position contain an EC5 [19], and sufficiently large point sets with no EC5 have arbitrarily large subsets of collinear points [1]. Empty convex pentagons also play a significant role in analyzing flips for coloured edges and proving the Orbit Theorem [27].

We show that the absence of EC5's leads to a simple condition for flip cut edges. In particular, if a point set has no EC5's, then edge e is a flip cut edge if and only if e is in (i.e., is a diagonal of) at least two EC4's. In fact, it suffices to exclude local EC5's:

Lemma 17. *An edge e that is not a diagonal of an EC5 is a flip cut edge if and only if it is a diagonal of at least two EC4's, i.e., $|Z| \geq 2$.*

Proof. If e is a flip cut edge, then by Theorem 9, G_Z is disconnected, so we must have $|Z| \geq 2$. (For this direction we did not use the assumption about EC5's.)

For the other direction, suppose e is not a flip cut edge and $|Z| \geq 2$. We will show that e is a diagonal of an EC5. By Theorem 9, G_Z is connected. Because neighbours in G_Z are consecutive in the orderings of A and B (by Observation 15), there must be two edges f, g in Z that are incident at one end and consecutive in the ordering of A (or B) at the other end.

Without loss of generality, suppose $f = a_i b_j$ and $g = a_{i+1} b_j$. See Figure 3.15. The five points a_i, a_{i+1}, b_j, u, v form a convex pentagon. (Observe that the angle at u is convex because of the EC4 $Q(f)$, and the angle at a_i is convex because a_i and a_{i+1} make empty triangles with uv and a_i precedes a_{i+1} in A, and similar arguments show that the other

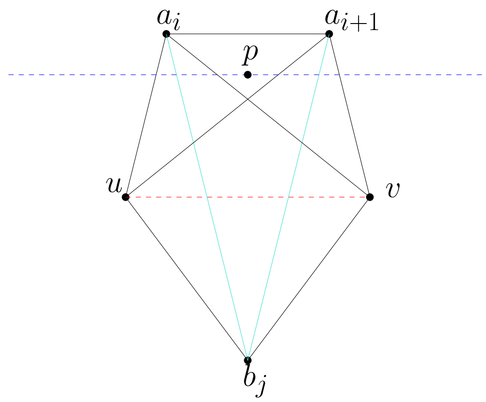


Figure 3.15: The endpoints of the edges a_ib_j and $a_{i+1}b_j$ form an EC5 with u and v.

angles are convex.) If this pentagon is not empty, then it contains a point p outside $Q(f) \cup Q(g)$ and such a point of minimum y coordinate is a point of A between a_i and a_{i+1}, a contradiction. Thus the pentagon is empty, so e is a diagonal of an EC5. □

3.4.2 Grid Point Sets

The points of a $k \times \ell$ grid have no empty convex pentagons, so by Lemma 17, an edge is a flip cut edge if and only if $|Z| \geq 2$. Edges "near" the boundary of the grid may fail to be flip cut edges, but we show that edges farther from the boundary are flip cut edges, and we show that in an infinite grid, all edges are flip cut edges. We first deal with horizontal/vertical edges.

Claim 18. Let e be a horizontal/vertical edge of a grid point set. Then e is a flip cut edge if and only if it does not lie on the boundary of the grid.

Proof. Note that e must have unit length since no edge goes through intermediate points. We may assume without loss of generality that e goes from (x, y) to $(x + 1, y)$. If e lies on the boundary, then Z is empty. For the converse, suppose e is not on the boundary. Then the points just above e, $a_1 = (x, y + 1), a_2 = (x + 1, y + 1)$, and the points just below e, $b_1 = (x + 1, y - 1), b_2 = (x, y - 1)$, lie in the grid, and the two edges a_1, b_1 and a_2, b_2 lie in Z, so e is a flip cut edge by Lemma 17. □

Next consider an edge e that is not horizontal or vertical. Reflect so that e goes from the point (x, y) to the point $(x + \Delta x, y + \Delta y)$ with $\Delta x \geq \Delta y > 1$. Since an edge cannot go through intermediate points, $\gcd(\Delta x, \Delta y) = 1$. Let L be the line through e. On the infinite grid, translate L upward and parallel to itself until it hits grid points and call the resulting line L_U. Similarly, translate L downward and parallel to itself until it hits grid points, and call the resulting line L_D.

Claim 19. A grid point makes an empty triangle with e iff it lies on L_U or L_D.

Proof. The "if" direction is clear. For the other direction, consider a triangle T determined by e and an apex grid point strictly above L_U. The area of T is strictly larger than the area of a triangle with apex on L_U. By Pick's theorem [17] (the area of a triangle on the grid is the number of grid points strictly inside the triangle plus half the number on the boundary of the triangle), triangle T cannot be empty. $\square$

By the easy direction of this claim, a grid point $a \in L_U$ and a grid point $b \in L_D$ provide an edge $ab \in Z$ iff ab crosses e. The other direction of the claim is used in Figure 3.16b to demonstrate that some edges are not flip cut edges.

We now analyze the points on L_U and L_D. We claim that there is a point $a = (a_x, a_y)$ on L_U with $a_x \in [x, x + \Delta x)$ and $a_y \in (y, y + \Delta y]$. To justify this, note that if $a = (a_x, a_y)$ is a point on L_U, then so is $(a_x + \Delta x, a_y + \Delta y)$ and $(a_x - \Delta x, a_y - \Delta y)$. This implies that there is a point $a = (a_x, a_y)$ on L_U with $a_x \in [x, x + \Delta x)$. Then we must have $a_y \in (y, y + \Delta y]$, as otherwise $(a_x, a_y - 1)$ would be between L and L_U. By symmetry, the point $b = (b_x, b_y) := ((x+\Delta x)-(a_x-x), (y+\Delta y)-(a_y-y)) = (2x+\Delta x-a_x, 2y+\Delta y-a_y)$ lies on L_D and has $b_x \in (x, x + \Delta x]$ and $b_y \in [y, y + \Delta y)$. Then, the segment ab crosses e at their midpoints, so $ab \in Z$. The same is true for the segment from $(a_x + i\Delta x, a_y + i\Delta y)$ to $(b_x - i\Delta x, b_y - i\Delta y)$ for any $i = 0, 1, \ldots$. This implies that on an infinite grid, Z has infinite size. Thus we have proved:

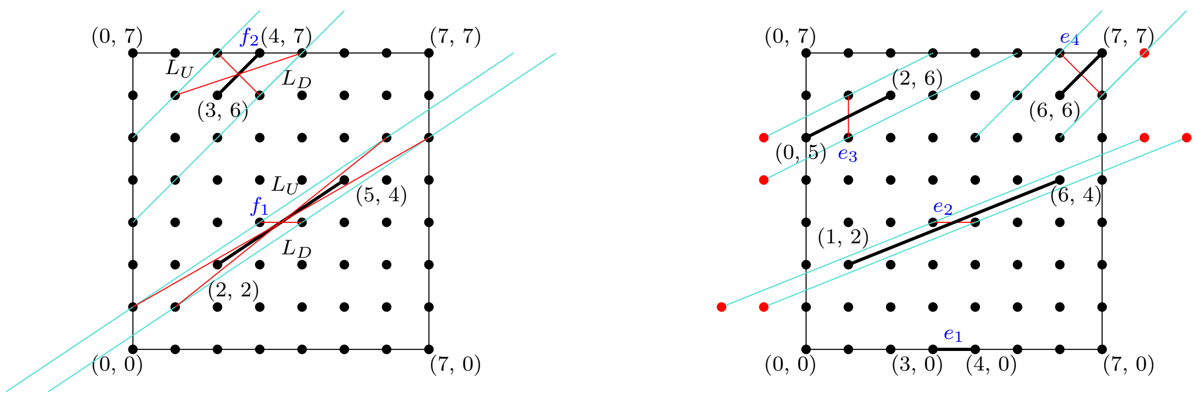
Lemma 20. *For the points of the infinite integer grid, every edge is a flip cut edge.*

In a finite grid, the boundary edges are not flip cut edges, but they are not the only exceptions, see Figure 3.16. It is possible to characterize flip cut edge in terms of integer solutions to equations, but for now we simply note that a large enough grid of n points has $\Theta(n^2)$ flip cut edges, as justified by the following.

Proposition 21. *Let G be a $k \times \ell$ grid and let G' be the middle one-third grid (the grid which contains points that have x-coordinates between $\frac{k}{3}$ and $\frac{2k}{3}$ and y-coordinates between $\frac{l}{3}$ and $\frac{2l}{3}$). Then every edge of G' is a flip cut edge of G.*

Proof. Let e be an edge of G'. We will show that Z has at least two elements. As above, we may assume that e goes from (x, y) to $(x + \Delta x, y + \Delta y)$ with $\Delta x \geq \Delta y > 1$ and $\gcd(\Delta x, \Delta y) = 1$. Since e lies in G' we have $\Delta x \leq k/3$, $\Delta y \leq \ell/3$.

As noted above, there is a point $a = (a_x, a_y)$ on L_U with $a_x \in [x, x + \Delta x)$ and $a_y \in (y, y+\Delta y]$ and a point $b = (b_x, b_y) = (2x+\Delta x-a_x, 2y+\Delta y-a_y)$ on L_D with $b_x \in (x, x+\Delta x]$ and $b_y \in [y, y + \Delta y)$. Both a and b lie in the grid G and $ab \in Z$. For our second element of Z, we use the edge from $(a_x + \Delta x, a_y + \Delta y)$ on L_U to $(b_x - \Delta x, b_y - \Delta y)$ on L_D. Note that these points lie in G. $\square$



(a) f_1 and f_2 are flip cut edges since they are in at least two EC4's, as shown by the diagonals (in red) with endpoints on L_U and L_D (in cyan).

(b) e_1, e_2, e_3, e_4 are not flip cut edges since they are in 0 or 1 EC4's.

Figure 3.16: Example edges in a 7×7 grid.

3.4.3 The Number of Flip Cut Edges

In Section 1.4.1, we claimed that channels, as shown in Figure 1.5, are point sets with $\Theta(n^2)$ flip cut edges. Here we justify that claim.

Proposition 22. *For a channel, every edge between an interior point on the upper reflex chain and an interior point on the lower reflex chain is a flip cut edge.*

Proof. Let the points of the upper reflex chain be $t_1, \ldots, t_n$ and the points of the lower reflex chain be $b_1, \ldots, b_n$, as shown in Figure 1.5. Consider the edge $b_i t_j$ where $i, j \notin \{1, n\}$. We use Lemma 17 to show that b_it_j is a flip cut edge. There is no EC5 with b_it_j as a diagonal because such an EC5 would have to have at least 3 points of one reflex chain, say the upper one, and those points cannot be part of a convex polygon with b_i. However, both the edges $b_{i-1}t_{j+1}$ and $b_{i+1}t_{j-1}$ form EC4's together with b_it_j. Thus by Lemma 17, b_it_j is a flip cut edge. $\square$

3.4.4 The Number of Components from a Flip Cut Edge

In Section 1.4.1, we claimed that one flip cut edge e can cause $\Theta(n)$ disconnected components in $\mathcal{F}_{-e}$. Here we justify that claim.

Theorem 23. *A flip cut edge can create $O(n)$ disconnected components in the flip graph, and this is the most possible.*

Proof. First of all, connected components in the flip graph $\mathcal{F}_{-e}$ correspond to connected components in G_Y (by Theorem 8) and therefore correspond to disjoint sets of points, so there are at most n of them.

The hourglass in Figure 1.6 is an example of a point set of size $2n+2$ with a flip cut edge e that results in n components in $\mathcal{F}_{-e}$. To construct the hourglass, place points $a_1, \ldots, a_n$ along the top half of a circle, and let $b_1, \ldots, b_n$ be the diametrically opposite points. For each i construct the wedge from a_i to b_{i-1} and b_{i+1}. Place points u and v on the horizontal diameter of the circle, so close to the center of the circle that they are *inside* all the wedges. Then G_Y consists of the n edges a_ib_i, and no two of these are connected in G_Y. Thus (by Theorem 8) $\mathcal{F}_{-e}$ has n disconnected components. $\square$

Chapter 4

Flip Cut Set for Points in Convex Position

In this chapter, we are going to give some examples of flip cut sets, and then prove that flip cut number for points in convex position is $n - 3$.

4.1 Some Examples of Flip Cut Sets

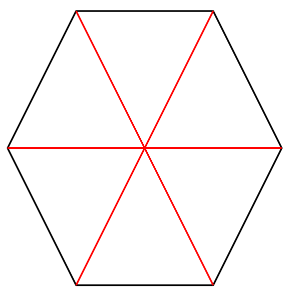
As mentioned in the introduction, a point set in convex position has no flip cut edge, i.e., its flip cut number is greater than 1. We are going to prove this in the next section of this chapter. In this section, we will show all the possible flip cut sets for convex point sets with sizes six, seven, and eight. We have not yet been able to characterize flip cut sets for convex point sets. That being said, examining these examples should give some clue on where to start.

First, note that for a convex point set of size five, there exists no flip cut set, since forbidding two edges fails to disconnect the flip graph and forbidding more edges than that leaves us with no valid triangulation. For a convex point set of size six, there are two minimal flip cut sets with size three, which are shown in Figures 4.1 and 4.2.

The connected components of the flip cut sets in Figures 4.1 and 4.2 contain only frozen triangulations. In Figure 4.3, a flip cut set for a convex point set of size seven is shown. Each of the connected components of the flip graph for this set of forbidden edges contains 3 vertices (triangulations). Hence, both components are non-frozen components.

Figure 4.4 shows all the other possible minimal flip cut sets of a point set with size seven. We can see that all of these flip cut sets have size 4. Figure 4.5a shows some minimal flip cut sets for a convex point set of size eight. The size of all these sets is five. However, Figure 4.5b shows that not all sets of size five are flip cut sets.

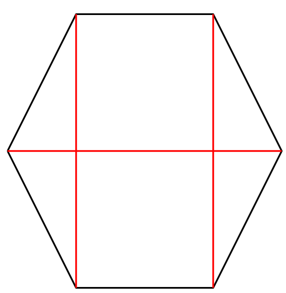
Although we have not been able to characterize flip cut sets for points in convex position, in the next section of this chapter, we show that the flip cut number for convex point sets is $n-3$ where n is the size of the point set.



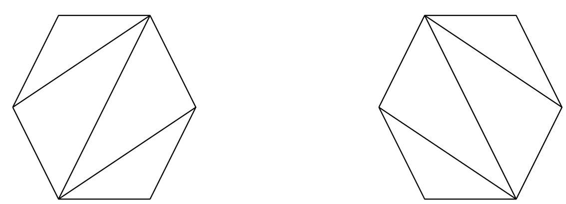
(a) A flip cut set X for a convex point set of size six

(b) The components of $\mathcal{F}_{-X}$ are frozen

Figure 4.1: An example of flip cut set for $n = 6$

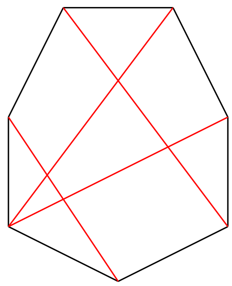


(a) Another flip cut set X for a convex point set of size six

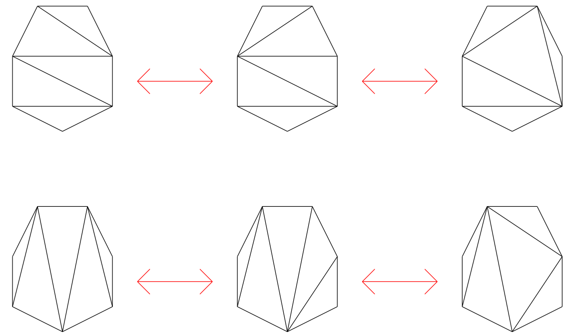


(b) The frozen components of $\mathcal{F}_{-X}$

Figure 4.2: Another example of flip cut set for $n = 6$ whose components are frozen



(a) A flip cut set X for a convex point set of size seven



(b) The components of $\mathcal{F}_{-X}$ are not frozen.

Figure 4.3: An example of a flip cut set for $n = 7$ whose components are not frozen

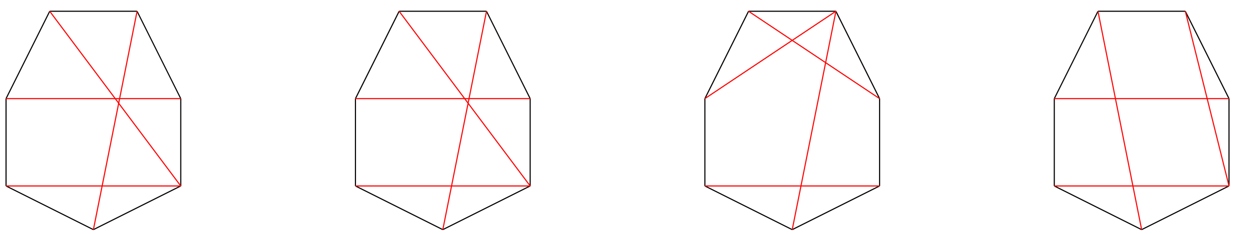
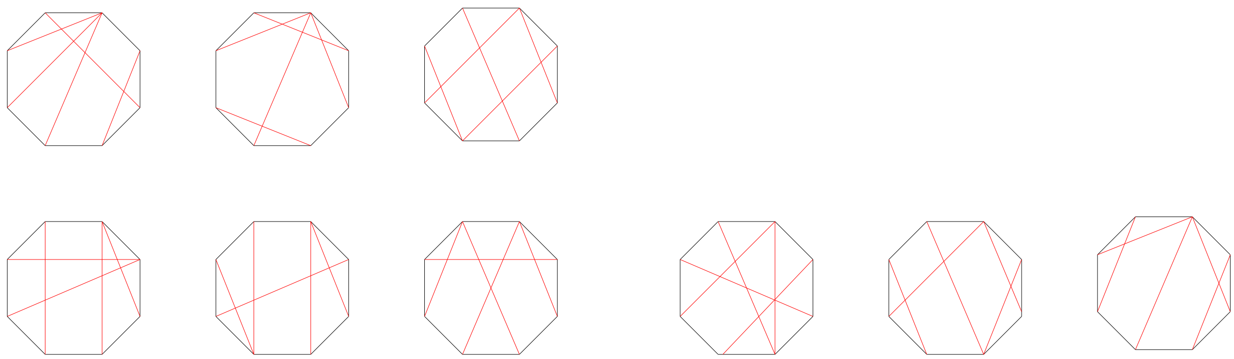


Figure 4.4: Other flip cut sets for a convex point set of size seven

4.2 Flip Cut Number for Points in Convex Position

As mentioned in the introduction, a point set in convex position has no flip cut edge, i.e., its flip cut number is greater than 1. In this section we show that the flip cut number of n points in convex position is $n-3$.

We give a direct proof, but first we discuss what the result means in terms of associahedra. The flip graph of n points in convex position is the 1-skeleton of the $(n-3)$-dimensional associahedron $\mathcal{A}_{n-1}$ (see Figure 1.3), so by Balinski's theorem [3], the 1-skeleton is $(n-3)$-connected. The dual polytope $\bar{\mathcal{A}}_{n-1}$ is also an $(n-3)$-dimensional polytope with an $(n-3)$-connected 1-skeleton. The ***face lattice*** of an n-dimensional convex polytope is defined as a lattice with $n+1$ layers; the i'th layer of the lattice contains vertices corresponding to each of the $(i-1)$-dimensional faces of the polytope. There exists an edge between two vertices (faces) of two consecutive layers, if the associated face with a higher dimension contains the face with a lower dimension. The face lattice of $\bar{\mathcal{A}}_{n-1}$ is the inverted face lattice (in which the order of layers has been reversed) of $\mathcal{A}_{n-1}$, so the vertices of $\bar{\mathcal{A}}_{n-1}$



(a) Some flip cut sets for a point set of size eight

(b) Some forbidden sets of size five that are not flip cut sets

Figure 4.5: Some sets of forbidden edges. Ones on the left are flip cut sets, but sets on the right are not.

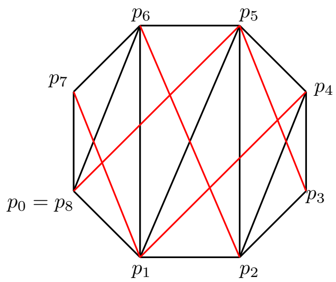
correspond to the $(n-4)$-dimensional faces (the facets) of $\mathcal{A}_{n-1}$ and the edges of $\bar{\mathcal{A}}_{n-1}$ correspond to $(n-5)$-dimensional faces of $\mathcal{A}_{n-1}$. Forbidding a chord of the original polygon corresponds to deleting a facet of $\mathcal{A}_{n-1}$, i.e., a vertex of $\bar{\mathcal{A}}_{n-1}$. Deleting fewer than $n-3$ facets of $\mathcal{A}_{n-1}$ leaves the remaining facets connected via $(n-5)$-dimensional faces. What our result shows is that the remaining vertices (0-dimensional faces) of $\mathcal{A}_{n-1}$ are connected via the remaining edges (1-dimensional faces) of $\mathcal{A}_{n-1}$. We do not see how to prove our result using this polyhedral interpretation. Instead, we give a direct proof that the flip cut number of n points in convex position is $n-3$.

For points in convex position, we number the points $p_1, \ldots, p_n$ in cyclic order around the convex hull. The edges $p_i p_{i+1}$ (addition modulo n) must be present in every triangulation. A ***chord*** is a line segment joining points that are not consecutive in the cyclic ordering. An ***ear*** is a chord of the form $p_{i-1}p_{i+1}$ (addition modulo n), and we say that this ear ***cuts off*** point p_i. Every triangulation of a convex point set on $n \geq 4$ points has at least two ears (because the dual tree has at least two leaves). A ***star triangulation*** is a triangulation all of whose chords are incident to the same point.

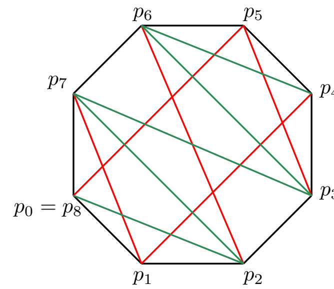
For a set X of chords, we study connectivity of $\mathcal{F}_{-X}$. We define the ***degree*** of a point p in X to be the number of chords of X incident to p.

We begin with a necessary condition for a flip cut set. This provides a rigourous proof that the flip cut number is greater than 1, and will also be needed for our main result.

Lemma 24. *If X is a flip cut set for a set of points in convex position, then every point is incident to an edge of X, i.e., the degree in X of every point is at least 1.*



(a) A zigzag triangulation T (in black) and a set of forbidden edges X (in red) that makes T a frozen triangulation.



(b) Another zigzag triangulation T' in $\mathcal{F}_{-X}$.

Figure 4.6: Construction of a flip cut set of size $n-3$

Proof. We prove the contrapositive: if there is a point $p \in P$ with no incident edges of X, then X is not a flip cut set. Specifically, we prove that any triangulation T in $\mathcal{T}_{-X}$ is connected to the star triangulation centered at p. This is in fact the standard way to show connectivity of the (full) flip graph of a convex point set. In our situation, we must show that the required flips do not use forbidden edges.

So, suppose triangulation T contains a triangle incident to p whose other two points, q and r, are not consecutive on the convex hull. Now T contains a triangle, say qrs, on the other side of qr. Then $pqsr$ forms a convex quadrilateral, and we can flip the chord qr to the chord ps (which is not forbidden). This increases the number of chords incident to p, so, by induction, T can be flipped to the star triangulation centered at p without using forbidden edges. □

Corollary 25. *There is no flip cut edge in a convex point set.*

In the rest of this section we prove that the flip cut number for n points in convex position is $n-3$. We begin with the following two lemmas.

Lemma 26. *A convex point set of size n has a flip cut set of size $n-3$.*

Proof. Consider a "zigzag" triangulation T of a convex n-gon as shown in Figure 4.6a. This triangulation has $n-3$ chords that form a path $p_0, p_{n-2}, p_1, p_{n-3}, ..., p_{\lfloor \frac{n}{2} \rfloor}$. Each chord e of T can flip to a unique other chord $f(e)$. If we forbid the $n-3$ chords $X := \{f(e) : e \text{ a chord of } T\}$ then T becomes a frozen triangulation, i.e., it becomes an isolated node in the flip graph $\mathcal{F}_{-X}$. In order to complete the proof, we only need to show that there exists another triangulation in $\mathcal{F}_{-X}$ different from T. (This is where we use the fact that T is a zig-zag triangulation—for example, a star triangulation would not work.) We create another zigzag triangulation T' in the following way. The path of chords for T' starts at the same vertex $p_0 = p_n$, but the first chord from p_0 goes in the other direction, i.e., to p_2. See Figure 4.6b. The new zigzag path is $p_0, p_2, p_{n-1}, p_3, p_{n-2}, ..., p_{\lceil \frac{n}{2} \rceil}$. We prove that T' does not use any forbidden edges. Note that each forbidden edge crosses only one chord of T. However, each chord of T', crosses at least two chords of T. This is because each chord in T' of the form $p_{i+2}p_{n-i}$, for $i = 0, .., \lfloor \frac{n}{2} \rfloor - 2$, crosses $p_{i+1}p_{n-i-2}$ and $p_{i+1}p_{n-i-3}$ from T and any chord of the form $p_{i+1}p_{n-i}$, for $i = 1, .., \lfloor \frac{n}{2} \rfloor - 2$, in T' crosses $p_i p_{n-i-1}$ and $p_{i-1}p_{n-i-1}$ from T. So, since each forbidden edge $e \in X$ crosses exactly one chord of T, but each chord of T' crosses at least two chords of T; therefore T' does not contain any forbidden edges. □

Lemma 27. *Consider a convex point set P with n points and a set X of forbidden edges with $|X| \leq n - 3$. Then there is a triangulation of P that uses no forbidden edges, i.e., $\mathcal{T}_{-X}$ is non-empty.*

Proof. The proof is by induction on n with base case $n = 3$. If there is a point p that is incident to no forbidden edges, then take the star triangulation at p. Otherwise, suppose every point is incident to at least one forbidden edge. There are n chords of the form $p_{i-1}p_{i+1}$ (addition modulo n). So, there exists at least one of them that is not forbidden. Suppose it is $p_{i-1}p_{i+1}$. We build a triangulation containing the ear $p_{i-1}p_{i+1}$. Let P' be $P - \{p_i\}$ of size $n' = n - 1$. Since p_i is incident to at least one forbidden edge, there are at most $n - 4 = n' - 3$ forbidden edges of P'. Thus, by induction, there is a triangulation of P' that uses no forbidden edges. Together with chord $p_{i-1}p_{i+1}$ this provides the desired triangulation of P. Thus $\mathcal{T}_{-X}$ is non-empty. □

Theorem 28. *The flip cut number of a convex point set P with n points is $n - 3$.*

Proof. By Lemma 26, we can disconnect P's flip graph by forbidding $n-3$ edges. So, now we only need to show that if X is a set of forbidden edges and $|X| \leq n - 4$, then $\mathcal{F}_{-X}(P)$ is connected. By Lemma 24, $\mathcal{F}_{-X}(P)$ is connected if there is a point $p \in P$ not incident to any edge of X. Thus we may assume that every point $p \in P$ is incident to some edge of X. Let S and T be two triangulations of P that do not contain any edges of X. We will prove that S is connected to T in $\mathcal{F}_{-X}(P)$. We will prove this by induction on n, with the base case $n = 3$ where the statement is vacuously true.

We know that each triangulation contains at least two ears. We consider two cases.

Case 1. S and T have a shared ear that cuts off point p_i. Consider sub-polygon P' obtained by removing point p_i. By assumption there is a forbidden edge incident to p_i. Let X' be the forbidden edges of X not incident to p_i. Then $|X'| \leq |X|-1 \leq |P|-5 = |P'|-4$, so by induction, the triangulations of P' induced from S and T are connected in $\mathcal{F}_{-X'}(P')$. Thus S and T are connected in $\mathcal{F}_{-X}(P)$.

Case 2. S and T have no shared ear. We claim that there is an ear e_1 in S and an ear e_2 in T such that e_1 and e_2 do not cross, i.e., the points that are cut off by e_1 and by e_2 are not adjacent on the convex hull. Suppose S has an ear e_1 that cuts off point p_i. T has at least two ears, and if they cross e_1, they must cut off points p_{i-1} and p_{i+1}. But then the second ear of S cannot cross both these ears of T unless $n = 4$ (and $X = \emptyset$), in which case, a single flip converts S to T.

Thus we may assume a pair of non-crossing ears e_1 of S and e_2 of T. Suppose e_1 cuts off point q_1 and e_2 cuts off point q_2. Let P_1 be $P - \{q_1\}$. P_1 has $n_1 = n-1$ points. Let X_1 be the forbidden edges of X induced on P_1. By assumption there is at least one edge of X incident to q_1, so $|X_1| \leq |X| - 1 \leq n-5 = n_1 - 4$. By induction, the flip graph $\mathcal{F}_{-X_1}(P_1)$ is connected. Let S_1 be the triangulation of P_1 formed by cutting the ear e_1 off S. The plan is to apply induction on P_1 to connect triangulation S_1 to a new triangulation R_1 of P_1 that includes the chord e_2.

We construct R_1 as follows. Let P_2 be $P - \{q_1, q_2\}$. Then P_2 has size $n_2 = n-2$. Let X_2 be the forbidden edges of X induced on P_2. Then $|X_2| \leq |X_1| \leq n_1 - 4 = n_2 - 3$. By Lemma 27, P_2 has a triangulation that uses no chords of X_2. Adding chord e_2 yields the triangulation R_1.

By induction on P_1 and X_1 the triangulations S_1 and R_1 are connected in $\mathcal{F}_{-X_1}(P_1)$. Finally, R_1 and T share the ear e_2, so by Case 1, they are connected in $\mathcal{F}_{-X}(P)$. Altogether, we have connected S to T in $\mathcal{F}_{-X}(P)$, as required. $\square$

Chapter 5

Conclusions and Open Problems

We examined connectivity of the flip graph of triangulations when some edges between points are forbidden, and introduced the concepts of flip cut edges, flip cut sets, and the flip cut number. We gave an $O(n \log n)$ time algorithm to identify flip cut edges and test connectivity after forbidding a flip cut edge, and we proved that the flip cut number of a convex n-gon is $n-3$. We conclude with some open questions.

1. Is there a polynomial-time algorithm to test if a set of edges is a flip cut set? To compute the flip cut number?

2. What happens if we specify both a set of forbidden edges and a set of forced (constrained) edges. Can we characterize when the resulting flip graph is disconnected? A better understanding of this situation might be helpful for solving the previous open question.

3. The asymptotic diameter of the flip graph of a convex n-gon is $2n-10$—a famous result of Sleater, Tarjan and Turston [35], improved to all $n > 12$ by Pournin [33]. For a convex point set and a set X of forbidden edges with $|X| < n-3$ what is the diameter of $\mathcal{F}_{-X}$ in terms of n and $|X|$?

4. It is open whether there is a polynomial-time algorithm for the flip distance problem for a convex polygon. Is the following generalization NP-complete: Given n points in convex position, a set X of forbidden chords, two triangulations T_1 and T_2, and a number k, is the flip distance from T_1 to T_2 in $\mathcal{F}_{-X}$ less than or equal to k?

5. A flip cut edge is a bottleneck to connecting triangulations via flips. One might guess that point sets with no flip cut edges provide better mixing properties. More generally, how does the flip cut number affect mixing properties?

References

[1] Zachary Abel, Brad Ballinger, Prosenjit Bose, Sébastien Collette, Vida Dujmović, Ferran Hurtado, Scott Duke Kominers, Stefan Langerman, Attila Pór, and David R Wood. Every large point set contains many collinear points or an empty pentagon. *Graphs and Combinatorics*, 27(1):47–60, 2011.

[2] Nima Anari, Shayan Oveis Gharan, and Cynthia Vinzant. Log-concave polynomials, entropy, and a deterministic approximation algorithm for counting bases of matroids. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 35–46. IEEE, 2018.

[3] Michel L Balinski. On the graph structure of convex polyhedra in n-space. *Pacific Journal of Mathematics*, 11(2):431–434, 1961.

[4] Marshall Bern and David Eppstein. Mesh generation and optimal triangulation. *Computing in Euclidean Geometry*, pages 47–123, 1995.

[5] Reza Bigdeli and Anna Lubiw. Forbidding edges between points in the plane to disconnect the triangulation flip graph. *arXiv e-prints arXiv:2206.02700*, 2022.

[6] Prosenjit Bose and Ferran Hurtado. Flips in planar graphs. *Computational Geometry*, 42(1):60–80, 2009.

[7] Prosenjit Bose, Anna Lubiw, Vinayak Pathak, and Sander Verdonschot. Flipping edge-labelled triangulations. *Computational Geometry: Theory and Applications*, 68:309–326, 2018.

[8] Pietro Caputo, Fabio Martinelli, Alistair Sinclair, and Alexandre Stauffer. Random lattice triangulations: Structure and algorithms. *The Annals of Applied Probability*, 25(3):1650–1685, 2015.

[9] Cesar Ceballos, Francisco Santos, and Günter M Ziegler. Many non-equivalent realizations of the associahedron. *Combinatorica*, 35(5):513–551, 2015.

[10] L. Paul Chew. Constrained Delaunay triangulations. *Algorithmica*, 4(1):97–108, 1989.

[11] Bhaskar DasGupta, Xin He, Tao Jiang, Ming Li, John Tromp, and Louxin Zhang. On distances between phylogenetic trees. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, volume 97, pages 427–436. SIAM, 1997.

[12] Jesús De Loera, Jörg Rambau, and Francisco Santos. *Triangulations: Structures for Algorithms and Applications*, volume 25. Springer Science & Business Media, 2010.

[13] Herbert Edelsbrunner. *Geometry and Topology for Mesh Generation.* Cambridge University Press, 2001.

[14] David Eppstein. Happy endings for flip graphs. *Journal of Computational Geometry*, 1(1):3–28, 2010.

[15] David Eppstein and Daniel Frishberg. Improved mixing for the convex polygon triangulation flip walk. *arXiv e-prints arXiv:2207.09972*, July 2022.

[16] Jerome Galtier, Ferran Hurtado, Marc Noy, Stephane Perennes, and Jorge Urrutia. Simultaneous edge flipping in triangulations. *Int. J. Comput. Geometry Appl.*, 13:113–133, 04 2003.

[17] Branko Grunbaum and G. C. Shephard. Pick's theorem. *The American Mathematical Monthly*, 100(2):150–161, 1993.

[18] Sabine Hanke, Thomas Ottmann, and Sven Schuierer. The edge-flipping distance of triangulations. *Journal of Universal Computer Science*, 2:570–579, 1996.

[19] Heiko Harborth. Konvexe fünfecke in ebenen punktmengen. *Elemente der Mathematik*, 33:116–118, 1978.

[20] Michael Houle, F. Hurtado, M. Noy, and E. Rivera-Campo. Graphs of triangulations and perfect matchings. *Graphs and Combinatorics*, 21:325–331, 2005.

[21] Ferran Hurtado, Marc Noy, and Jorge Urrutia. Flipping edges in triangulations. *Discrete & Computational Geometry*, 22(3):333–346, 1999.

[22] Iyad Kanj, Eric Sedgwick, and Ge Xia. Computing the flip distance between triangulations. *Discrete & Computational Geometry*, 58(2):313–344, 2017.

[23] C. L. Lawson. Generation of a triangular grid with application to contour plotting. Technical report, Memo 299, Jet Propulsion Laboratory, 1972.

[24] C. L. Lawson. Transforming triangulations. *Discrete Mathematics*, 3(4):365–372, 1972.

[25] Carl W. Lee. The associahedron and triangulations of the n-gon. *European Journal of Combinatorics*, 10(6):551–560, 1989.

[26] D. T. Lee and Arthur K. Lin. Generalized Delaunay triangulation for planar graphs. *Discrete & Computational Geometry*, 1:201–217, 1986.

[27] Anna Lubiw, Zuzana Masárová, and Uli Wagner. A proof of the orbit conjecture for flipping edge-labelled triangulations. *Discrete & Computational Geometry*, 61(4):880–898, 2019.

[28] Anna Lubiw and Vinayak Pathak. Flip distance between two triangulations of a point set is NP-complete. *Computational Geometry*, 49:17–23, 2015.

[29] Lisa McShine and Prasad Tetali. On the mixing time of the triangulation walk and other Catalan structures. In *Randomization Methods in Algorithm Design*, volume 43 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 147–160. American Mathematical Society, 1999.

[30] Michael Molloy, Bruce A. Reed, and William L. Steiger. On the mixing rate of the triangulation walk. In *Randomization Methods in Algorithm Design*, volume 43 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 179–190. American Mathematical Society, 1999.

[31] Naomi Nishimura. Introduction to reconfiguration. *Algorithms*, 11(4):52, 2018.

[32] Alexander Pilz. Flip distance between triangulations of a planar point set is APX-hard. *Computational Geometry*, 47(5):589–604, 2014.

[33] Lionel Pournin. The diameter of associahedra. *Advances in Mathematics*, 259:13–42, 2014.

[34] Dana Randall and Prasad Tetali. Analyzing Glauber dynamics by comparison of Markov chains. *Journal of Mathematical Physics*, 41(3):1598–1615, 2000.

[35] Daniel D Sleator, Robert E Tarjan, and William P Thurston. Rotation distance, triangulations, and hyperbolic geometry. *Journal of the American Mathematical Society*, 1(3):647–681, 1988.

[36] Jan van den Heuvel. The complexity of change. In S. Blackburn, S. Gerke, and M. Wildon, editors, *Surveys in Combinatorics*, volume 409 of *London Mathematical Society Lecture Note Series*, pages 127–160. Cambridge University Press, 2013.

[37] Uli Wagner and Emo Welzl. Connectivity of triangulation flip graphs in the plane. *CoRR*, abs/2003.13557, 2020.

[38] Uli Wagner and Emo Welzl. Connectivity of triangulation flip graphs in the plane (Part I: Edge flips). In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2823–2841, 2020.