A Parameterized Algorithm for Upward Planarity Testing of Biconnected Graphs

by

Hubert Yan-Chor Chan

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

Waterloo, Ontario, Canada, 2003

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS

I hereby declare that I am the sole author of this 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.

Abstract

We can visualize a graph by producing a geometric representation of the graph in which each node is represented by a single point on the plane, and each edge is represented by a curve that connects its two endpoints.

Directed graphs are often used to model hierarchical structures; in order to visualize the hierarchy represented by such a graph, it is desirable that a drawing of the graph reflects this hierarchy. This can be achieved by drawing all the edges in the graph such that they all point in an upwards direction. A graph that has a drawing in which all edges point in an upwards direction and in which no edges cross is known as an upward planar graph. Unfortunately, testing if a graph is upward planar is NP-complete.

Parameterized complexity is a technique used to find efficient algorithms for hard problems, and in particular, NP-complete problems. The main idea is that the complexity of an algorithm can be constrained, for the most part, to a parameter that describes some aspect of the problem. If the parameter is fixed, the algorithm will run in polynomial time.

In this thesis, we investigate contracting an edge in an upward planar graph that has a specified embedding, and show that we can determine whether or not the resulting embedding is upward planar given the orientation of the clockwise and counterclockwise neighbours of the given edge. Using this result, we then show that under certain conditions, we can join two upward planar graphs at a vertex and obtain a new upward planar graph. These two results expand on work done by Hutton and Lubiw [22].

Finally, we show that a biconnected graph has at most $k!8^{k-1}$ planar embeddings, where k is the number of triconnected components. By using an algorithm by Bertolazzi et al. [4] that tests whether a given embedding is upward planar, we obtain a parameterized algorithm, where the parameter is the number of triconnected components, for testing the upward planarity of a biconnected graph. This algorithm runs in $O(k!8^kn^3)$ time.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The area of graph drawing deals with geometric representations of abstract graphs, and has applications in many different areas such as software architecture, database design, project management, electronic circuits, and genealogy. These geometrical representations, known as graph drawings, represent each vertex as a point on the plane, and each edge as a curve connecting its two endpoints. A broader treatment of graph drawing can be found in Di Battista et al. [9] or Kaufmann and Wagner [24].

In order for a drawing to be useful, there are different criteria that we may want the drawing to satisfy. The set of criteria that are required for a specific drawing depends on the application area. Unfortunately, for many sets of drawing criteria, the problem of deciding whether a given graph has a drawing that satisfies these criteria is NP-complete. In our discussion of algorithmic results for graphs, we will use the number of vertices, n, as the input size, and we will use r to denote the total number of both sources (vertices that have no incoming edges) and sinks (vertices that have no outgoing edges).

One example of an NP-complete decision problem in graph drawing is the problem of deciding if a graph has a drawing in which no edges cross and in which all edges are drawn upward; this is the problem that we consider in this thesis. In order to try to solve this problem efficiently, we will apply techniques from parameterized complexity, an area in computational complexity developed by Downey and Fellows [15], which we will describe later in this chapter.

A common criterion for a drawing is that each edge is drawn as a straight line. A drawing that satisfies this criterion is called a *straight line* drawing. If each edge is drawn as either a horizontal or a vertical line, the drawing is called *rectilinear*. A *polyline drawing* is a drawing in which each edge is drawn as a polygonal line, that is, a curve that is made up of line segments. The points at which these line segments are connected are called *bends*. A polyline drawing in which vertices and bends are placed only at grid points is called a *grid drawing*.

Another important consideration for a graph drawing is its size. The *area* of a graph drawing is defined as the area of the smallest rectangle that encloses the drawing, and the *height* and *width* of a drawing are the height and width, respectively, of this same

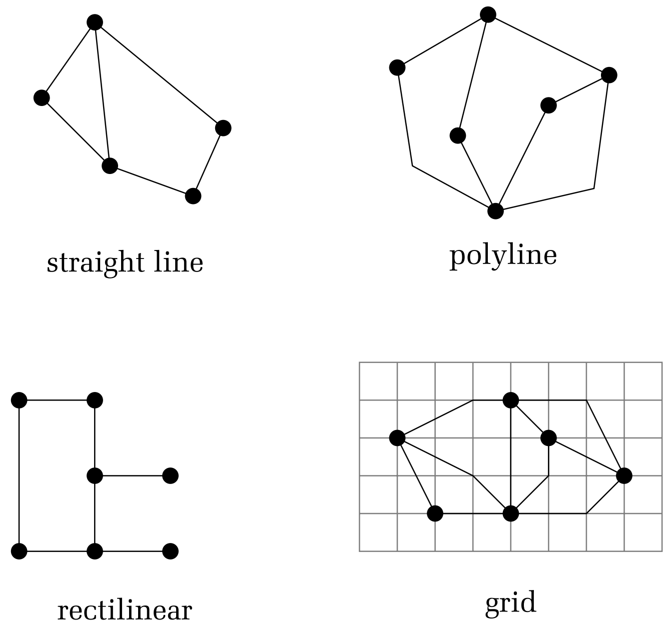


Figure 1.1: Examples of straight line, polyline, rectilinear, and grid drawings

rectangle. It is often desirable to obtain a drawing that has the minimum possible area, height, or width, while ensuring that vertices are not drawn too close to each other.

1.1 Upward planarity

A common criterion in graph drawing is that no edges cross, as crossing edges may render a drawing incomprehensible. A drawing that satisfies this criterion is known as *planar*, and a graph that has a planar drawing is a planar graph. One can determine whether or not a graph G is planar in linear time, as shown by Hopcroft and Tarjan [21].

We now informally define several terms related to planar drawings; formal definitions are given in Chapter 2. A planar drawing separates the plane into connected regions called *faces*. The unbounded face is called the *external face*.

For each vertex of a graph, a planar drawing of the graph defines an ordering of the edges around the vertex. The collection of these orderings is called an *embedding*; this allows us to compare the structures of two drawings. We say that two drawings are *equivalent* if they produce the same embedding, and we say that two drawings are *strongly equivalent* if they are equivalent and have the same outer face.

Given a directed graph, it is often also desirable that all edges point in the same direction, say upward, as this helps a viewer to visualize the precedence relationships among vertices. A drawing in which all edges point upward is called an *upward* drawing. An example of such a directed graph is a function call graph for a large software system: if the graph is drawn with all edges pointing upward, functions that are placed

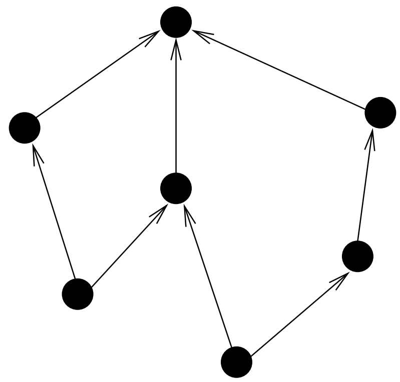


Figure 1.2: An example of an upward planar graph

with a smaller y coordinate could be interpreted as those providing higher-level functionality than those that are placed with a larger y coordinate.

The problem of determining whether or not a directed graph is upward, where we allow edges to cross, can also be solved in linear time. This can be decided by finding an ordering of the vertices such that for every directed edge (u, v), u comes before v in the ordering. We can then draw each vertex such that vertices that appear later in the ordering are drawn above those that appear earlier in the ordering. The problem of finding a suitable ordering of the vertices is called *topological sorting* and can be found, along with its solution, in a basic algorithms book [8, 7]. Both texts also show that one can obtain such an ordering if and only if the given graph does not have a directed cycle.

If a graph has a drawing that is both planar and upward, the graph is said to be *upward planar*. For example, the graph in Figure 1.2 is upward planar. Some graphs, such as the one in Figure 1.3, are not upward planar. Figure 1.3a shows a drawing that is upward but not planar, and Figure 1.3b shows a drawing that is planar but not upward. A given embedding is *upward planar* if there is a corresponding upward planar drawing.

Although testing whether a graph is planar can be done in linear time, and testing whether a graph is upward can be done in linear time, if we wish to determine whether a graph is upward planar, the problem becomes NP-complete [19]. This problem is known as *upward planarity testing*.

1.1.1 Related problems

In addition to upward planarity, there may be other desirable characteristics to have in our drawings. One possible type of drawing is the *layered drawing*, also called a *levelled drawing*, in which each vertex is drawn on a horizontal line, called a *layer*, and each edge is represented as a straight line between two different layers. A *proper layered drawing* is a layered drawing in which each edge has endpoints on consecutive layers when we order the layers according to their y coordinate. Proper layered upward planar

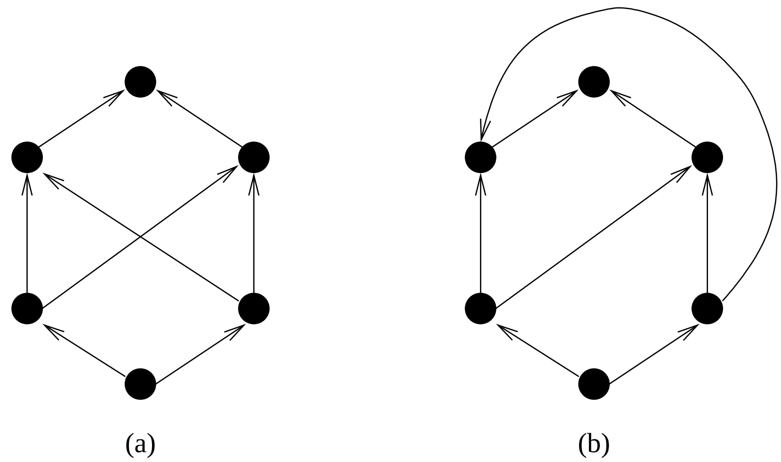


Figure 1.3: A graph that is not upward planar

drawings may be useful for visualizing software systems that have a layered architecture; in a layered architecture, components are assigned to layers, and a component in one layer may only communicate with components in the adjacent layers. Jünger et al. [23] investigate proper layered drawings and show that checking whether a graph has a proper layered upward planar drawing can be done in linear time: the proper layered and upward conditions force a layer assignment to the vertices, and Jünger et al. give a linear time algorithm that determines whether there exists a planar layered drawing that has the same layer assignment as a given layer assignment.

Another approach is to relax our requirements for a drawing, instead of adding new requirements. Bertolazzi et al. [3] relax slightly the upward planarity condition by introducing the concept of *quasi-upward planarity*. A drawing is quasi-upward planar if it is planar, and for each vertex v, there is a region containing v such that if we draw a horizontal line through v, all outgoing edges are drawn above the line, and all incoming edges are drawn below the line. An intuitive definition of a quasi-upward planar drawing is a drawing that is planar, and in which every edge is drawn upward as it leaves or enters a vertex. An example of a quasi-upward planar drawing is given in Figure 1.4. As can be seen from the example, quasi-upward planar graphs may have directed cycles, unlike upward planar graphs. The definition of quasi-upward planarity was motivated by the fact that upward planarity is a fairly strict criterion for drawings, and can be satisfied by few directed graphs, which limits its usefulness.

When creating a quasi-upward planar drawing, we also want to minimize the number of *bends* in the edges, where a bend for a quasi-upward planar drawing is defined as a point on a curve where it is tangent to the horizontal line. Intuitively, a bend is a point at which an edge changes its vertical direction. Bertolazzi et al. give a polynomial time algorithm for finding a quasi-upward planar drawing with minimum number of bends for a directed graph with a given embedding, and study the problem of finding a quasi-upward planar drawing for a digraph that does not have a specified embedding.

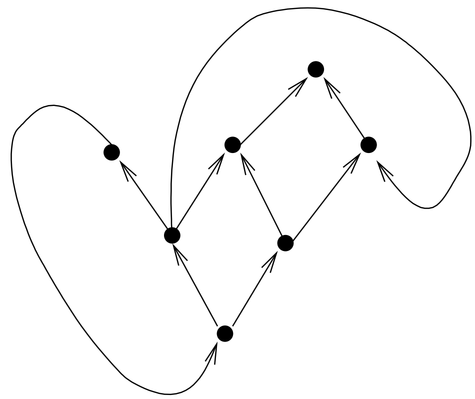


Figure 1.4: A quasi-upward planar drawing

Although we may know that a graph is upward planar, we may still be interested in other aspects of an upward planar drawing, such as its area. We may also want the drawing to display isomorphisms and symmetries in the graph, in order to help visualize subgraphs that are similar. Di Battista et al. [13] investigate the area requirements for upward drawings, and show that there exist graphs that require exponential area for straight line drawings. For a special class of graphs, they are able to give an algorithm that produces an upward planar drawing with small area. An *st-graph* is an acyclic digraph that has exactly one source s, exactly one sink t, and the edge (s,t). Di Battista et al. present a linear time algorithm that produces a planar polyline drawing for any st-graph that has $O(n^2)$ area, displays the symmetries and isomorphisms in the graph, and has a small number of bends in the drawing of the edges. Their algorithm works by first producing a drawing, known as a dominance drawing, in which every edge is drawn either as a vertical line pointing upward, a horizontal line pointing towards the right, or a diagonal line pointing upward towards the right, and then rotating the drawing by 45°. Their dominance drawing is produced by performing two topological sorts based on edge orderings in a given embedding.

Bertolazzi et al. [4] investigate the problem of testing whether or not a given embedding has a corresponding drawing that is upward planar. They introduce a new problem of assigning sources and sinks to each face in a given embedding, and define consistency properties for a set of assignments. They then show that an embedding is upward planar if and only if it has a consistent assignment of sources and sinks to faces, and give an $O(n+r^2)$ algorithm that determines whether or not an embedding is upward planar by finding a consistent assignment, if one exists. This is used in the same paper to give an algorithm for checking whether a triconnected graph is upward planar, and is used by Papakostas [25] for checking whether an outerplanar graph is biconnected.

If we are given a drawing instead of an embedding, we can improve our time; Di Battista and Liotta [10] give a linear time algorithm that checks whether a given drawing is upward planar.

1.1.2 Algorithms on special classes of graphs

Since upward planarity testing is NP-complete, a common approach to solving the problem is to look at special classes of graphs. In this section, we survey the known algorithms for upward planarity testing on special classes of graphs.

Di Battista and Tamassia [12] give a linear time algorithm that gives an upward planar drawing for an st-graph. Their algorithm operates by finding a planar embedding of the graph, sorting the edges and vertices based on the embedding, and determining the locations of the vertices and edges based on their sorted order. If a straight line drawing is required, the time complexity is $O(n \log n)$.

Di Battista et al. [11] show that a bipartite graph is upward planar if and only if it is planar. Since planarity testing can be done in linear time, it follows that upward planarity testing can also be done in linear time on bipartite graphs. Note as well that since trees are always both bipartite and planar, trees are always upward planar.

Bertolazzi and Di Battista [2] give a $O(n + r^3 \log r)$ algorithm that determines if a triconnected graph, a graph in which there are at least three vertex-disjoint paths between any pair of vertices, is upward planar. Bertolazzi et al. [4] improve this to $O(n + r^2)$ time. Since r is no greater than the number of vertices, one can write this as $O(n^2)$. Both results are based on the following classic theorem.

Theorem 1.1. *[32] A triconnected graph has a unique planar embedding, up to reversal of all the edge orderings in the embedding.*

An *outerplanar* graph is a graph that can be drawn such that all vertices are on the outer face. Papakostas [25] gives polynomial time algorithms for outerplanar graphs: one that tests whether a graph has an upward planar drawing that is also outerplanar in $O(n)$ time, and one that tests whether it has an upward planar drawing that may not be outerplanar in $O(n^2)$ time. His algorithms operate by using tree search on the dual of the graph, and are based on the fact that an outerplanar graph has a unique outerplanar embedding.

Hutton and Lubiw [22] give a quadratic time algorithm that operates on single-source graphs. To find an upward planar drawing, their algorithm first separates a graph into biconnected components, and then the biconnected components are separated into triconnected components. They then take advantage of the fact that a triconnected component has a unique planar embedding, and that the entire graph has only one source, which means that every source s in a triconnected component is either the unique source of the graph or is a shared vertex with other components. Thus each triconnected component can only take on a limited number of "shapes," and can be represented by a smaller marker graph that captures the "shape" of the component. The algorithm then finds an upward planar drawing of each triconnected component together with the marker graphs of the adjoining triconnected components, and then combines the upward planar drawings of the triconnected components to form an upward planar drawing of the entire graph.

Class	Complexity	Reference
st-graph	$O(n)$	[12]
bipartite	$O(n)$	[11]
triconnected	$O(n + r^2)$	[4]
outerplanar	$O(n^2)$	[25]
single source	$O(n)$	[5]

Table 1.1: Algorithms on special classes of graphs

Bertolazzi et al. [5] improve on Hutton and Lubiw's result by giving a linear time algorithm. Given a planar embedding, they create a new graph, called a face-vertex graph, by creating a new vertex for each face, and connecting the original vertices to the vertices that represent the incident faces. Based on a subgraph of this face-vertex graph, they give a new characterization for upward planar graphs, and show that this yields a linear-time algorithm for single-source graphs. Bertolazzi et al. also give a parallel algorithm that takes $O(\log n)$ time using $n \log \log n / \log n$ processors.

1.1.3 Combinatorial characterizations

Given a directed graph, we may be able to tell whether it is upward planar by testing some other properties. The characterizations mentioned in this section may not translate well to efficient algorithms, but they are useful for proving that given graphs are upward planar.

Tamassia and Tollis [29] show that in any upward planar embedding, the outgoing edges of any vertex v must appear consecutively around v, as must the incoming edges. This is easily shown by taking an upward planar drawing that corresponds to the planar embedding, and drawing a horizontal line through v. Since the drawing is upward planar, all outgoing edges must appear above the line, and all incoming edges must appear below the line.

Theorem 1.2. *[29] All outgoing (incoming) arcs of any vertex v of an upward planar embedding Γ appear consecutively around v.*

Di Battista and Tamassia [12] show the equivalence among several types of upward drawings.

Theorem 1.3. *[12] Let G be a digraph. The following statements are equivalent*

1. *G is a subgraph of a planar st-graph;*

2. *G is upward planar;*

3. *G has an upward planar grid drawing; and*

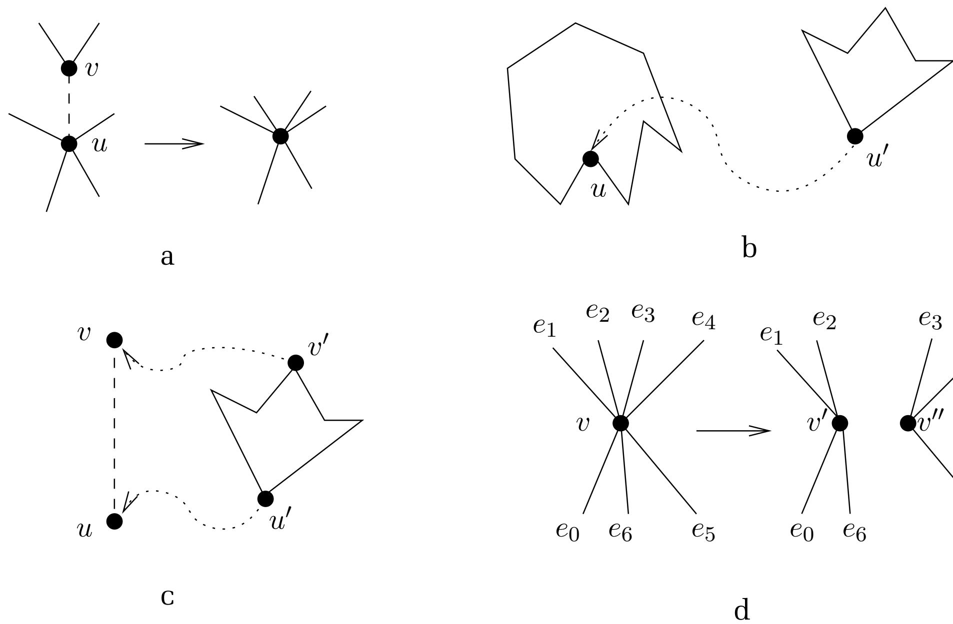


Figure 1.5: Operations on upward planar graphs.

4. G has an upward planar straight line drawing.

Thomassen [30] characterizes graphs that have upward planar drawings in which all faces are convex polygons, and presents the following characterization for single source graphs:

Theorem 1.4. *[30] Let φ be the planar drawing of an acyclic digraph G that has only one source, s. Then there is an upward planar drawing strongly equivalent to φ if and only if s is on the outer face and, for every cycle Σ in φ, Σ has a vertex that does not have any outgoing edges inside or on Σ.*

Hutton and Lubiw [22] give another characterization for single source graphs, based on Thomassen's result:

Theorem 1.5. *[22] Given a single source directed acyclic graph G, and a planar drawing φ of G with a specified outer face and source s on the outer face, G has an upward planar drawing strongly equivalent to φ if and only if the following condition holds: For each vertex $v \in V$, v is a sink on the outer face of the planar drawing φ_v induced on $P_G(v)$, where $P_G(v)$, the predecessor set of v, is the set of vertices from which there is a directed path to v.*

They also consider four operations that can be performed on upward planar graphs, and show that after performing these operations, the resulting graphs will be upward planar if certain conditions hold. The operations are illustrated in Figure 1.5.

The first operation contracts an edge in an upward planar graph. Contracting the edge (u, v) means that we remove (u, v) and identify its endpoints so that the edges incident to the new vertex are those that were incident to u or v. This is defined more formally in Chapter 2. The idea behind the proof of this lemma is that we can construct a new drawing of the graph obtained by contracting (u, v) from a drawing of G by "pulling" v and all its incident vertices along a corridor around (u, v) until u and v meet.

Lemma 1.1. *[22] (Figure 1.5a) Let G be a directed acyclic graph with the edge (u, v) where v has no other incoming edges. Then the graph obtained by contracting (u, v) is upward planar if G is upward planar.*

The second operation joins together two upward planar graphs at a single vertex.

Lemma 1.2. *[22] (Figure 1.5b) Let G be an upward planar digraph with a vertex u, and let H be an upward planar digraph with single source u'. Let G' be the digraph formed by identifying u and u' in $G \cup H$. Then G' is upward planar.*

The third operation replaces an edge with an upward planar graph.

Lemma 1.3. *[22] (Figure 1.5c) Let G be an upward planar digraph with an edge (u, v), and H be an upward planar digraph with single source u' and a sink v' both on the outer face. Let G' be the digraph formed by removing the (u, v) edge of G and adding H, identifying vertex u with u' and vertex v with v'. Then G' is upward planar.*

The last operation splits a vertex into two vertices.

Lemma 1.4. *[22] (Figure 1.5d) Let G be a directed acyclic graph that has an upward planar drawing where the edge ordering defined by the drawing about vertex v is $e_0, \ldots, e_{k-1}$. Let G' be the directed acyclic graph formed by splitting v into two vertices: v' incident with edges $e_i, \ldots, e_j$, and v'' with edges $e_{j+1}, \ldots, e_{i-1}$ ($i \neq j \mod k$). Then G' is upward planar. If G had a single source, and i and j are such that each of v' and v'' retain at least one incoming edge, then the resulting G' is also a single source digraph.*

The proofs of these lemmas assume that the graphs involved are all single source graphs. Hutton and Lubiw, therefore, show that given an upward planar graph G, we can construct a single source upward planar graph G' that has G as a subgraph, and such that the conditions necessary to apply the lemmas still hold. The idea behind this construction is that we "resolve" each source by adding an incoming edge that connects it to a vertex below it.

Lemma 1.5. *[22] Let G be a connected upward planar digraph. Then G is a subgraph of some single-source upward planar H such that all non-source $v \in V(G)$ have the same indegree in G as in H.*

1.2 Parameterized complexity

One technique for dealing with NP-complete problems is called parameterized complexity. Parameterized complexity is a relatively new technique, introduced by Downey and Fellows [15], motivated by the desire to find efficient algorithms for hard problems. Since many graph drawing problems are NP-complete, we may want to apply parameterized complexity techniques to solve these problems.

All known techniques to solve NP-complete problems suffer from a combinatorial explosion, resulting in running times that are exponential in the size of the input. The idea behind parameterized complexity is that we wish to constrain this combinatorial explosion to a limited aspect of the problem that is described by a parameter k that is polynomial in the input size. If we can then keep this parameter small, as is the case in many practical situations, we may be able to solve the problem efficiently for instances whose size is larger than what we would normally be able to solve using an algorithm that runs in time exponential in the size of the problem.

Downey and Fellows define a parameterized problem as a subset L of $\Sigma^* \times \Sigma^*$, where the first item of the pair is an instance of the problem and the second item is the parameter. L is the language of all "yes" instances of the problem. They then define fixed-parameter tractability as follows:

Definition 1.1. A parameterized problem $L \subseteq \Sigma^* \times \Sigma^*$ is *fixed-parameter tractable* if there is an algorithm that correctly decides, for input $(x, y) \in \Sigma^* \times \Sigma^*$, whether $(x, y) \in L$ in time $f(k)n^\alpha$ where n is the size of the main part of the input x, i.e. $|x| = n$, k is the parameter which we can take to be the length of y, i.e. $k = |y|$, α is a constant (independent of k), and f is an arbitrary function.

The classical example for parameterized complexity is the VERTEX COVER problem, which is NP-complete. However, if we wish to find a vertex cover of size k, where k is a fixed parameter, this can be done in $O(kn + \frac{4}{3}^k k^2)$ time, as shown by Balasubramanian et al. [1].

Two parameters that can be used in graph problems are treewidth and pathwidth. Given a graph, the concepts of *treewidth* and *pathwidth* describe how similar the graph is to a tree or a path. Zhou [33] investigates applying the concept of treewidth and pathwidth to graph drawing. Since many graph drawing problems are easily solvable on trees and paths, Zhou investigates adapting algorithms for trees and paths to draw graphs of bounded treewidth or pathwidth, using the treewidth and pathwidth of the graph as the parameter.

Dujmović et al. [17, 16] investigate parameterized complexity in layered graph drawings. In the first paper, Dujmović et al. give a linear time algorithm that decides if a graph has a drawing with h layers in which none of the edges cross. The fixed parameter used in their algorithm is h, the number of layers, and the algorithm is based on the fact that a graph that has an h-layered planar drawing has a pathwidth that is a function of h. They then show that this algorithm can be modified to solve related problems, such as determining whether, for a fixed k, a given graph can be drawn with at most k crossings (known as the k-crossing problem), or determining whether a graph can be drawn such that deleting at most r edges, for fixed r, removes all crossings (known as the r-planarization problem).

The second paper by Dujmović et al. investigates the problem of drawing a graph on two layers, and uses the technique of reduction to a problem kernel: given an instance I of the problem with parameter k, we construct a new problem J whose size is no larger than $f(k)$ for some function f. We can then solve J and use its solution to solve I. Dujmović et al. use this technique to obtain a linear time algorithm for the 2-planarization problem on two-layer graphs, as well as for the 1-layer planarization problem, in which the permutation of the vertices on one layer is fixed.

1.3 Thesis outline

The organization of this thesis is as follows. In Chapter 2, we define the terminology that we will be using in the remainder of the thesis. In Chapter 3 we prove technical lemmas and introduce notation that we will use in our proofs of our main results.

We then extend the results of Hutton and Lubiw given in Lemmas 1.1 and 1.2. In Chapter 4, we investigate the conditions under which contracting an edge ϵ in an upward planar graph with a given upward planar embedding results in an embedding that is also upward planar. Our characterization is based on the orientations of the clockwise and counterclockwise neighbours of ϵ in the given embedding. Using our edge contraction results, we then investigate in Chapter 5 the conditions under which joining together two upward planar graphs G_1 and G_2 form a graph that is upward planar.

In Chapter 6, we bound the number of possible embeddings of a biconnected graph by a function of the number of triconnected components, and give a parameterized algorithm that decides if a biconnected graph is upward planar in $O(k!8^{k-1}n^3)$ time, where the parameter k is the number of triconnected components. This result is an example of using parameterized complexity techniques to solve a graph drawing problem.

Finally, in Chapter 7, we give our conclusions and identify areas for further research.

Chapter 2

Definitions

In this chapter, we formally define the terminology that we use to describe graphs, drawings, and embeddings. In order to describe a drawing formally, we will also need to define some terms from topology.

2.1 Graphs

We assume the reader is familiar with the definitions of graphs and directed graphs. These can be found in a basic graph theory book such as Diestel [14] or Bondy and Murty [6]. Unless otherwise specified, the definitions from this section are taken from these books.

In this thesis, we are concerned primarily with upward planar drawings, which are drawings of directed graphs. Thus we assume, unless otherwise specified, that all graphs are simple (i.e. multiple edges and loops are not allowed) and are directed.

Given a graph G, we will denote its set of vertices $V(G)$ and its set of edges $E(G)$. Where there is no ambiguity as to the graph to which we refer, we may use simply V and E. When discussing the complexity of algorithms, we will use the number of vertices n of G as the input size.

In Chapters 5 and 6, we will be combining two smaller graphs to create a larger graph. We define the *union* $G_1 \cup G_2$ of two graphs G_1 and G_2 as the graph whose set of vertices $V(G_1 \cup G_2)$ is $V(G_1) \cup V(G_2)$ and whose set of edges $E(G_1 \cup G_2)$ is $E(G_1) \cup E(G_2)$.

If we are given a graph, we may wish to describe how well connected it is, that is, how many ways we can reach a given vertex from any other given vertex. The connectivity of a graph affects the number of possible planar drawings that it can have. An undirected graph is *connected* if there is a path from any vertex to any other vertex. A *connected component* is a maximal connected subgraph.

An undirected graph is *k-connected* if there are at least k vertex-disjoint undirected paths between any pair of vertices, and a *k-connected component* is a maximal k-connected subgraph. A graph that is 2-connected is also called *biconnected*, and a graph that is 3-connected is also called *triconnected*.

An equivalent definition of k-connectivity is that the removal of any set of k vertices, along with their incident edges, does not disconnect the graph. A set of vertices whose removal disconnects a graph is called a *cutset*, and a single vertex whose removal disconnects a graph is called a *cutvertex*.

We define the connectivity of a directed graph as the connectivity of its underlying undirected graph.

In Chapter 4, we give conditions under which contracting an edge in an upward planar graph results in a graph that is still upward planar. Given a directed or undirected graph G, we *contract* an edge $\epsilon = (v, w)$, which can be a directed or undirected edge, by removing v, w, and the edges incident to either v or w. We then add a new vertex v_ϵ, and for each edge (u, v) or (u, w) in G, we add the edge (u, v_ϵ). If G contains both the edges (u, v) and (u, w), we only add (u, v_ϵ) once. We say that (u, v) or (u, w) is the edge *corresponding* to (u, v_ϵ). If G is a directed graph, we also add the edge (v_ϵ, u) for each edge (v, u) or (w, u) in G. The resulting graph is denoted G/ϵ [14].

Given a directed graph, we may wish to describe the relationships among vertices, and between vertices and the edges incident to them. Given a directed edge $\epsilon = (v, w)$, v is the *tail* of ϵ, w is the *head*, and we say that w *dominates* v [30]. The edge ϵ is an *outgoing* edge of v, and an *incoming* edge of w. If there is a directed path from v to w, we say that v is a *predecessor* of w, and that w is a *successor* of v.

The number of outgoing and incoming edges incident to a given vertex can affect properties of a graph. A vertex has *outdegree* k if it has exactly k outgoing edges, and *indegree* ℓ if it has exactly ℓ incoming edges [6]. The outdegree and indegree of v are denoted $\deg^+(v)$ and $\deg^-(v)$, respectively. A vertex that has no incoming edges is called a *source*, while a vertex that has no outgoing edges is called a *sink* [30]. In other words, v is a source if $\deg^-(v) = 0$, and a sink if $\deg^+(v) = 0$.

2.2 Topology

We now define various topology terms that will be used to describe graph drawings and related concepts. We assume that the reader is familiar with the definitions of functions and of continuity.

Since our graphs are drawn on the plane, many functions that we will use have values in $\mathbb{R}^2$. For the function $f : S \to \mathbb{R}^2$, where S can be any arbitrary set, we will denote the x component of $f(s)$ by $f_x(s)$, and the y component by $f_y(s)$.

A drawing, which we will define formally below, represents each edge by a curve. A *Jordan curve* J is a continuous function from the interval $[0, 1]$ to $\mathbb{R}^2$. The *endpoints* of J are $J(0)$ and $J(1)$ [26]. A Jordan curve is *simple* if it does not intersect itself. In other words, J is simple if $J(p) = J(q)$ implies $p = q$. We may sometimes find it convenient to work with a set of points instead of a function. Given a curve J, we define the set $\hat{J} = \{p \mid J(x) = p \text{ for some } x \in [0, 1]\}$.

Since we desire drawings in which edges do not cross, we must define what it means for two curves to cross. We say that the curves J_1 and J_2 *cross* if $\hat{J}_1 \cap \hat{J}_2$ is nonempty. We say that the curve J *crosses* the point p if $p \in \hat{J}$. We use the term *intersects* as a synonym for "crosses".

In the drawings that we will be considering, the direction of each edges is important, thus we introduce terms to describe the direction of a curve. A curve is *monotone* if every horizontal line intersects it at most once. A curve J is *monotone increasing* if for every pair $a, b \in [0, 1]$, $a > b$ implies $J_y(a) > J_y(b)$. We may also say that J is *upward*, or that it *points upward*. If, instead, $a > b$ implies $J_y(a) < J_y(b)$, we say that the curve is *monotone decreasing*, is *downward*, or *points downward*.

In our proofs, we will want to describe different properties of subsets of the plane. An *open ball* centred at the point p and with radius $r > 0$ is the set $\{q | \operatorname{dist}(p, q) < r\}$, where dist is the Euclidean distance function. A set S is *open* if for every point p in S, there is an open ball centred at p that is a subset of S [27]. The *boundary* of a set S is the set of points p such that every open ball centred at p contains points from S as well as points that are not in S. Given a set S, if we remove its boundary from S, the resulting set is an open set. A set S is *connected* if, for every pair of points p and q, one can draw a curve J between p and q such that $\hat{J}$ is a subset of S [27]. We may refer to a connected subset of $\mathbb{R}^2$ as a *region*.

2.3 Graph drawing

Given a graph, we wish to produce a geometric representation of the graph. This representation is called a drawing.

A *drawing* φ of a directed or undirected graph G is a function that maps each vertex v of G to a distinct point $\varphi(v)$ on the plane, and each edge (v, w) to a simple Jordan curve $J = \varphi((v, w))$ with endpoints $\varphi(v)$ and $\varphi(w)$ [9]. If G is a directed graph, we use the convention that $\varphi(v) = J(0)$ and $\varphi(w) = J(1)$.

We will sometimes refer to the drawing of an edge or a vertex by the edge or the vertex itself. For example, when we say that the edges ϵ_1 and ϵ_2 cross, we mean that their associated curves $\varphi(\epsilon_1)$ and $\varphi(\epsilon_2)$ cross.

Given a drawing φ of a directed or undirected graph G, we may want to consider only a smaller part of the drawing. If H is a subgraph of G, we define φ_H as the drawing φ *induced* on H. That is, $\varphi_H(v) = \varphi(v)$ for all $v \in V(H)$, and $\varphi_H(\epsilon) = \varphi(\epsilon)$ for all $\epsilon \in V(H)$.

An undirected graph G is *planar* if there is a drawing of G that does not have crossing edges. Such a drawing is called a *planar drawing* [9].

A directed graph is called *upward planar* if it has a planar drawing in which all edges are drawn as curves that are monotone increasing. Such a drawing is called *upward planar* [9].

A planar drawing partitions the plane into regions called *faces* [9]. The *boundary* of each face is made up of edges and vertices; this definition for the boundary of a face gives the same set of points as the definition of the boundary of a region. Since the plane $\mathbb{R}^2$ is infinite, one of the faces must have infinite size; this face is called the *external face* [9] or *outer face* [14]. We say that a vertex or edge is *on* a face if it is part of the boundary

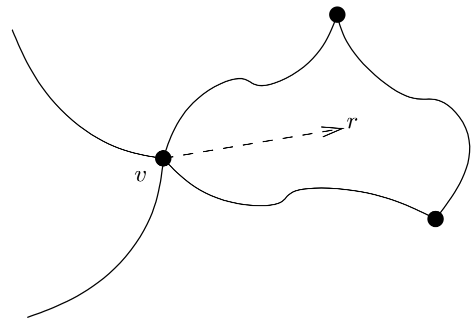


Figure 2.1: The face defined by the vertex v and the ray r

for that face. We say that a point p is *inside*, or *within*, a face F if p is an element of the region F. Similarly, we say that a curve J is *inside*, or *within*, a face F is $\hat{J}$ is a subset of F.

Remark 2.1. Given an upward planar drawing of a graph G and a vertex v of G, if we draw a ray r in any direction starting from v, then the part of r closest to v is either contained within a face or is co-linear with an edge. Hence in every direction, other than those that are co-linear with an edge, we can find a face that v is on. This is illustrated in Figure 2.1. When r is a vertical ray going downward from v and is not co-linear with an edge, we call the face that contains the initial portion of r the face *below* v. If r is a vertical ray going upward from v, we call this face the face *above* v.

Since two different drawings may have similar structures, we want to be able to describe the structure of a drawing in a combinatorial fashion, which will allow us to compare the structures of different drawings. For a vertex v in a directed or undirected graph, a planar drawing φ defines a clockwise circular order of the edges around v. The collection Γ of these orderings for each vertex is called a *(planar) embedding* [9], and we say that φ *corresponds to* Γ. The embedding associated with a drawing defines an equivalence relation between drawings. Two drawings are *equivalent* [9] if they have the same embedding. Two drawings are *strongly equivalent* [30] if, in addition to having the same embedding, they also have the same outer face.

Given an embedding Γ of a graph G and two edges ϵ_1 and ϵ_2 of G that have common endpoint v, we introduce terminology to describe the relationship between ϵ_1 and ϵ_2 in the clockwise edge ordering around v defined by Γ. If ϵ_2 comes immediately after ϵ_1 in the clockwise ordering, we say that ϵ_1 and ϵ_2 are *edge-ordering neighbours*, that ϵ_2 is the *clockwise neighbour* of ϵ_1, and that ϵ_1 is the *counterclockwise neighbour* if ϵ_2.

The embedding of a graph also defines an ordering of the edges that form its boundary. To formally define this ordering, we must first define the dual of a planar graph. Given a planar graph G with a specified drawing, we build the *dual graph* [9] G^* as follows: first, we create a vertex v_F for each face F of G. For each edge ϵ of G, ϵ is on two faces F_1 and F_2. We then create edge $\epsilon^* = \{v_{F_1}, v_{F_2}\}$ of G. The edge ϵ^* is called the *dual edge* of ϵ.

We can obtain a planar drawing of the dual graph G^* from a planar drawing of G:

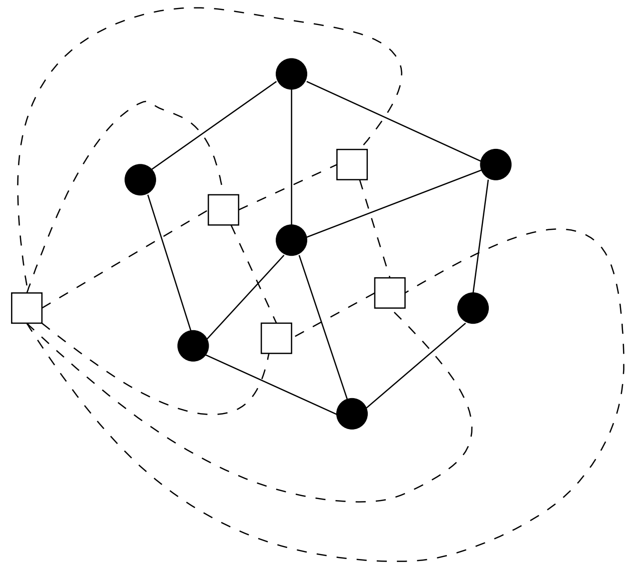


Figure 2.2: Drawing a dual graph. The edges of the dual graph are drawn as dashed lines, and the vertices are drawn as squares.

draw the vertex v_F of G^* in the face F of G, and draw the edge ϵ^* of G^* such that it crosses the edge ϵ of G. Thus, for every face F of G, we have a vertex v_F in G^*, which has an associated clockwise ordering of the edges around it. From this clockwise ordering of edges incident to v_F, we obtain a clockwise ordering of the edges that form the boundary of F.

Remark 2.2. Note that if π_1 and π_2 are consecutive edges in the edge ordering around F and have the common endpoint v, π_2 will be the counterclockwise neighbour of π_1 in the edge ordering around v. As well, if π_2 is the counterclockwise neighbour of π_1 in the edge ordering around v, π_1 and π_2 will be consecutive edges in an ordering around a face that is uniquely determined from the pair (π_1, π_2) as follows. Let $\pi_2 = (v, v_2)$, and let $\pi_3 = (v_2, v_3)$ be the counterclockwise neighbour of π_2 around v_2; π_3 will be our third edge in the face. Next, let $\pi_4 = (v_3, v_4)$ be the counterclockwise neighbour of π_3 around v_3; π_4 will be our fourth edge in the face. We continue this process until we reach π_1. The resulting list of edges forms the boundary of a face F, as shown in Figure 2.3.

Given a vertex and a face, we wish to be able to specify their relationship in a drawing φ. We say that a cycle Σ *surrounds* a vertex v in a drawing φ if any ray drawn from v, going in any direction, intersects the drawing of Σ. We say that a face F *surrounds* v if the cycle that makes up the boundary of F surrounds v.

In this thesis, we adopt the convention of using lowercase Roman letters to represent vertices, lowercase Greek letters to represent edges, and uppercase Roman letters to represent larger collections of vertices or edges, such as graphs, cycles, or faces. We will also use lowercase Roman letters to represent points on the plane, although we will typically only use the letters p, q, and r for this.

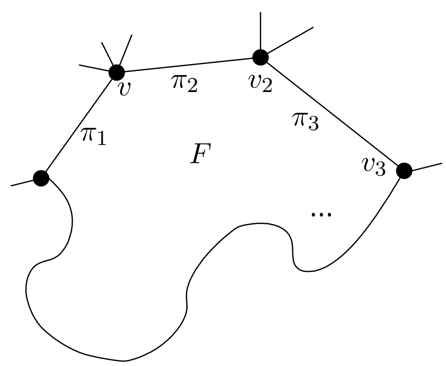


Figure 2.3: The face defined by two consecutive edges π_1 and π_2.

Chapter 3

Technical lemmas

In this chapter, we introduce lemmas and definitions that we will use to prove our main results. First, we will introduce the notion of transformations that take one graph drawing and produce another drawing. This allows us to produce a drawing in which we can ensure that extra criteria are satisfied. Then we show how to contract an edge in a graph and create an embedding of the new graph based on an embedding of the original graph. This is used in our edge contraction result. Finally, we discuss reversing the orientation of edges in a directed graph. This is used in our edge contraction results in which we show that the embedding that we obtain after contracting the edge ϵ is upward planar if and only if the graph that we obtain by instead reversing the direction of ϵ is upward planar.

3.1 Transformations

In our proofs later on, it may be easier to reason about drawings if, in addition to being upward planar, they satisfy other given criteria. For example, in Chapter 4, we want to obtain a drawing in which a given edge is drawn vertically. One way to ensure that these criteria are satisfied is to take a drawing and apply a transformation to it in order to obtain a new drawing that satisfies the criteria. However, we will want to ensure that after applying such a transformation, the drawing is still upward planar.

Definition 3.1. A function $\psi : \mathbb{R}^2 \to \mathbb{R}^2$ is called a *transformation* if, given any drawing φ of an arbitrary graph G, $\psi \circ \varphi$ is a drawing of G. In other words, $\psi \circ \varphi$ maps each vertex v to a distinct point $\psi \circ \varphi(v)$, and each edge (v, w) to a simple Jordan curve $\psi \circ \varphi((v, w))$ with endpoints $\psi \circ \varphi(v)$ and $\psi \circ \varphi(w)$. The graph G may be directed or undirected.

We now show that any function that is one-to-one and continuous is a transformation. Since many functions are known to be one-to-one and continuous, we can then conclude that they are transformations, without having to prove this separately for each function.

Lemma 3.1. *If ψ is a continuous one-to-one function from $\mathbb{R}^2$ to $\mathbb{R}^2$, then ψ is a transformation.*

Proof. Given a drawing φ of a graph G, we must show that $\psi \circ \varphi$ maps each vertex v to a distinct point $\psi \circ \varphi(v)$, and each edge (v, w) to a simple Jordan curve $\psi \circ \varphi((v, w))$ with endpoints $\psi \circ \varphi(v)$ and $\psi \circ \varphi(w)$.

To prove that $\psi \circ \varphi$ maps each vertex v to a distinct point, we show that if it maps v and w to the same point, then v and w must be the same vertex. If $\psi \circ \varphi(v) = \psi \circ \varphi(w)$, this means that $\varphi(v) = \varphi(w)$, since ψ is a one-to-one function. But as φ is a drawing, it maps each vertex to a distinct point. Therefore we conclude that $v = w$, which implies that $\psi \circ \varphi$ maps each vertex to a distinct point.

Now, we show that $\psi \circ \varphi$ maps each edge (v, w) to a Jordan curve. Since φ is a drawing, $\varphi((v, w)) = J$ is a Jordan curve. Now, since ψ is a continuous function from $\mathbb{R}^2$ to $\mathbb{R}^2$, and J is a continuous function from $[0, 1]$ to $\mathbb{R}^2$, $\psi \circ J$ is also a continuous function from $[0, 1]$ to $\mathbb{R}^2$. Thus $\psi \circ \varphi(v, w)$ is a Jordan curve.

We must show that $\psi \circ J = \psi \circ \varphi(v, w)$ is a simple curve. To show this, we show that if $\psi \circ J$ maps real numbers a and b to the same point, a and b must be the same number. Suppose that $\psi \circ J(a) = \psi \circ J(b)$. Since ψ is one-to-one, we have $J(a) = J(b)$. But J is a simple curve, which means that we must have $a = b$. So we can conclude that $\psi \circ J = \psi \circ \varphi((v, w))$ is a simple curve.

Finally, we show that the endpoints of $\psi \circ \varphi((v, w))$ are $\psi \circ \varphi(v)$ and $\psi \circ \varphi(w)$. Since φ is a drawing, the endpoints of J, $J(0)$ and $J(1)$, are $\varphi(v)$ and $\varphi(w)$. Without loss of generality, we can assume that $J(0) = \varphi(v)$, and $J(1) = \varphi(w)$. Now, the endpoints of the curve $\psi \circ J$ are $\psi \circ J(0)$ and $\psi \circ J(1)$. Since $J(0) = \varphi(v)$, and $J(1) = \varphi(w)$, we have $\psi \circ J(0) = \psi \circ \varphi(v)$ and $\psi \circ J(1) = \psi \circ \varphi(w)$, hence $\psi \circ \varphi(v)$ and $\psi \circ \varphi(w)$ are the endpoints of $\psi \circ \varphi((v, w))$, as required. $\square$

We can also show that any function that is a transformation must be one-to-one. This fact is used in the proof of Lemma 3.3.

Lemma 3.2. *If ψ is a transformation, then ψ must be a one-to-one function.*

Proof. Suppose that ψ is not one-to-one. Then there are points p and q on the plane such that $p \neq q$, but $\psi(p) = \psi(q)$. We now construct a graph G and a drawing φ of G such that $\psi \circ \varphi$ is not a drawing of G. We define G to be the graph with two vertices v and w and no edges, and set $\varphi(v) = p$ and $\varphi(w) = q$. The function φ maps each vertex to a distinct point, and G has no edges hence φ trivially satisfies the conditions for drawing edges. Therefore φ is a drawing of G. However, $\psi \circ \varphi(v) = \psi(p) = \psi(q) = \psi \circ \varphi(w)$, and hence $\psi \circ \varphi$ does not map the vertices of G to distinct points, which means that ψ is not a transformation. $\square$

Since we deal with upward planar drawings, we want to ensure that after we apply a transformation to a drawing, the resulting drawing is upward planar.

Definition 3.2. We say that a transformation $\psi : \mathbb{R}^2 \to \mathbb{R}^2$ *maintains upward planarity* if, given an upward planar drawing φ of a graph G, $\psi \circ \varphi$ is also an upward planar drawing of G.

Definition 3.3. We say that a function $\psi : \mathbb{R}^2 \to \mathbb{R}^2$ is *order preserving in* y, if for any two points, (x_1, y_1) and (x_2, y_2) in $\mathbb{R}^2$, $y_1 \geq y_2$ if and only if $\psi_y(x_1, y_1) \geq \psi_y(x_2, y_2)$, where $\psi_y(p)$ denotes the y component of $\psi(p)$.

Lemma 3.3. *If ψ is a transformation that is order preserving in y, then ψ maintains upward planarity.*

Proof. Consider an upward planar drawing φ of a graph G. In order to conclude that ψ maintains upward planarity, we must show that $\psi \circ \varphi$ is an upward planar drawing of G.

First we show that $\psi \circ \varphi$ is a planar drawing. Since φ is a planar drawing, we know that for edges ϵ_1 and ϵ_2 with $\epsilon_1 \neq \epsilon_2$, $\widehat{\varphi(\epsilon_1)} \cap \widehat{\varphi(\epsilon_2)} = \emptyset$. Define $J_1 = \varphi(\epsilon_1)$ and $J_2 = \varphi(\epsilon_2)$. To show that $\psi \circ \varphi$ is planar, we must show that $\widehat{\psi \circ J_1} \cap \widehat{\psi \circ J_2} = \emptyset$. Now, suppose that $\widehat{\psi \circ J_1} \cap \widehat{\psi \circ J_2} \neq \emptyset$. Let a be a point in $\widehat{\psi \circ J_1} \cap \widehat{\psi \circ J_2}$. Then there is a point $b_1 \in \hat{J}_1$ such that $\psi(b_1) = a$, and a point $b_2 \in \hat{J}_2$ such that $\psi(b_2) = a$. By Lemma 3.2, ψ is a one-to-one function, so we must have $b_1 = b_2$. But this gives us a point which is in both J_1 and in J_2, contradicting the planarity of φ. Thus a cannot exist, and $\widehat{\psi \circ J_1} \cap \widehat{\psi \circ J_2} = \emptyset$. In other words, $\psi \circ \varphi$ is a planar drawing.

Now we show that $\psi \circ \varphi$ is upward. Let (v, w) be a (directed) edge, and let $J = \varphi((v, w))$. The curve J points upward if and only if for every $a < b$, $J_y(a) < J_y(b)$. Now consider $\psi \circ J$. Given numbers a and b such that $a < b$, set $a' = J(a)$ and $b' = J(b)$. Since J points upward, we have $a' < b'$. But ψ is order preserving in y, so $\psi \circ J(a) = \psi(a') < \psi(b)' = \psi \circ J(b)$. Since $a < b$ implies that $\psi \circ J(a) < \psi \circ J(b)$, we can conclude that $\psi \circ J = \psi \circ \varphi((v, w))$ also points upwards. $\square$

Some basic transformations are flipping a drawing about the x or y axis, scaling a drawing, and translating a drawing. We can define these here formally. We define the functions from $\mathbb{R}^2$ to $\mathbb{R}^2$ $\mathsf{flip\_x}(x, y) = (x, -y)$ and $\mathsf{flip\_y}(x, y) = (-x, y)$; these functions flip a drawing about the x and y axis respectively. The flip_y function can be used to help us take advantage of left-right symmetry. The flip_x function produces a drawing in which all edges that pointed upward in the original drawing now point downward, and vice versa. This is used in our edge reversal results later in this chapter. The function $\mathsf{scale}_{(\sigma_x, \sigma_y)}(x, y) = (\sigma_x x, \sigma_y y)$, which takes two parameters σ_x and σ_y, scales a drawing horizontally by a factor of σ_x, and vertically by a factor of σ_y. The function $\mathsf{translate}_{(\tau_x, \tau_y)}(x, y) = (x + \tau_x, y + \tau_y)$, which takes two parameters τ_x and τ_y, translates a drawing. The scale and translate functions will be used in Section 5 to shrink parts of a drawing so that they fit within a given region. It is easy to see that all of these functions are continuous one-to-one functions, and hence are transformations. As well, flip_y, scale, and translate are all order preserving in y, and hence maintain upward planarity.

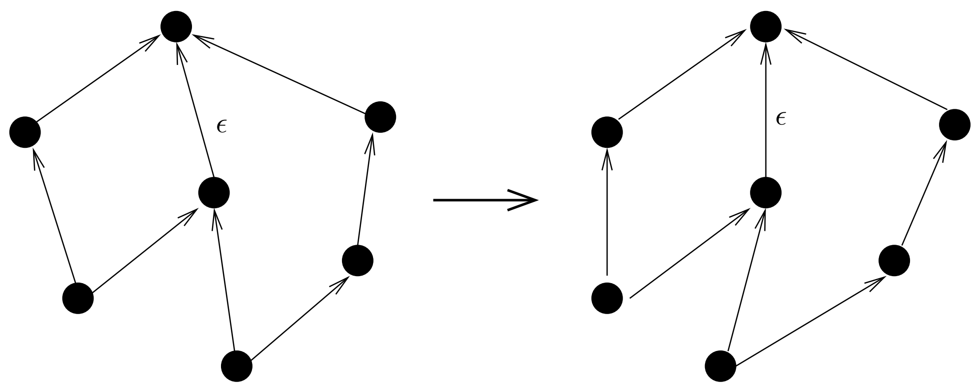


Figure 3.1: An example of skew applied to a drawing so that ϵ is vertical

In Chapter 4, we will want to produce a drawing of an upward planar graph in which a specified edge is vertical. From Theorem 1.3, we know that we can obtain a straight line upward planar drawing of the graph. We can then apply a transformation that produces a drawing in which the given edge is vertical. We now describe this transformation and show how to use it to produce our desired drawing.

We define a function, skew_σ, which takes one parameter σ, as $\text{skew}_\sigma(x, y) = (x+\sigma y, y)$. Figure 3.1 shows an example of skew applied to a drawing. Each component of skew is continuous, therefore the entire function is continuous. It is invertible, since the inverse of skew_σ is $\text{skew}_{-\sigma}$, hence is one-to-one, and is order preserving in y, since the y coordinate does not change. Therefore, by Lemma 3.3, skew_σ maintains upward planarity.

Note also that skew_σ maps straight lines to straight lines. Therefore if we apply skew_σ to a straight line drawing, the resulting drawing will be a straight line drawing.

Lemma 3.4. *The function skew_σ maps straight lines to straight lines.*

Proof. Every straight line can be described by an equation of the form $Ax + By + C = 0$, for some constants A, B, and C. Now, set $(x', y') = \text{skew}_\sigma(x, y) = (x + \sigma y, y)$. Looking at each component separately, we have $x' = x + \sigma y$ and $y' = y$. Using $y' = y$, and by rearranging, we obtain the equation $x = x' - \sigma y'$. Then $Ax + By + C = 0$ becomes $A(x' - \sigma y') + By' + C = 0$. Rearranging, we get $Ax' + (-A\sigma + B)y' + C = 0$, which is the equation of a straight line. $\square$

Remark 3.1. If we have a straight line drawing of a graph G, and we want to ensure that a specified edge ϵ is drawn vertically, we let $\frac{1}{\sigma}$ be the slope of ϵ. In other words, ϵ is represented by a line that can be described by the equation $y = \frac{1}{\sigma}x + b$ for some constant b. Rearranging this equation, we obtain $x = \sigma b + \sigma y$. If we apply $\text{skew}_{-\sigma}$ to this line, we get $(x', y') = \text{skew}_{-\sigma}(x, y) = (x - \sigma y, y) = (\sigma b + \sigma y - \sigma y, y) = (\sigma b, y)$, which defines a vertical line.

In this section, we have defined a transformation that can be applied to graph drawings, and have given conditions under which a function is a transformation. We then

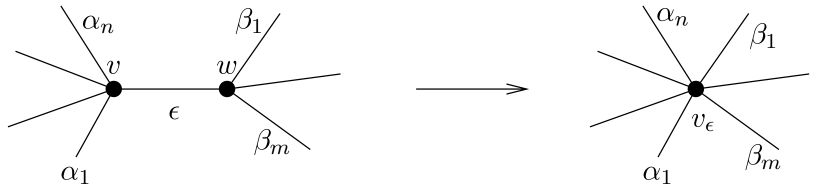


Figure 3.2: Clockwise ordering of edges around v_ϵ

defined a number of transformations that will be used in later sections.

3.2 Edge contraction and planar embeddings

Contracting an edge in a graph is a common operation on graphs. In Chapter 4, we will be considering a graph G with a given embedding, and contracting an edge ϵ in the graph, giving us a new graph G/ϵ. Therefore, we to define an embedding for G/ϵ that is, in some sense, similar to our embedding for G. We define this as follows.

Definition 3.4. Given a planar embedding Γ of a graph G, and the edge $\epsilon = (v, w)$ of G, we create Γ/ϵ, an embedding of G/ϵ. For a vertex u of G, if u is not an endpoint of ϵ, u has the same clockwise order of edges in Γ/ϵ as in Γ. If u is an endpoint of ϵ, u is not in G/ϵ, so we do not need to define the clockwise ordering of its incident edges. For the new vertex v_ϵ, we construct its edge list in Γ/ϵ from the edge lists of v and w in Γ. Let $(\epsilon = \pi_0, \pi_1, \ldots, \pi_n)$ be the clockwise order of the edges around v in Γ, and let $(\epsilon = \rho_0, \rho_1, \ldots, \rho_m)$ be the clockwise order of the edges around w in Γ. In Γ/ϵ, we make $(\pi_1, \ldots, \pi_n, \rho_1, \ldots, \rho_m)$ be the clockwise order of edges around v_ϵ. This process is illustrated in Figure 3.2. If an embedding Γ' can be obtained in this way from Γ, we say that Γ' is *derived from* Γ. By definition, Γ/ϵ is derived from Γ.

Lemma 3.5. *If Γ is a planar embedding of the graph G, and $\epsilon = (v, w)$ is an edge of G, then Γ/ϵ is a planar embedding of G/ϵ.*

Proof. Given a planar embedding Γ of G, let φ be a planar drawing of G corresponding to Γ. As shown independently by Fáry [18], Stein [28], and Wagner [31], every planar graph has a straight line planar drawing. Therefore we can assume, without loss of generality, that φ is a straight line drawing. Now consider $\varphi(\epsilon)$. We wish to find a region around ϵ in which no vertices other than v and w, and no edges other than those incident to v and w, are drawn. For example, we can set δ to be half the value of the minimum of the set of distances from ϵ to each vertex other than v and w, and from ϵ to each edge other than those incident to v or w. The distance between a vertex u and the edge ϵ is defined as the shortest distance between u and any point on ϵ, and the distance between an edge π and ϵ is defined as the shortest distance between a point from π and a point from ϵ. Then our requirement is satisfied by the region defined as the set of points within a distance of δ of ϵ. We call this region $\mathcal{R}$, and the complement of this region $\overline{\mathcal{R}}$.

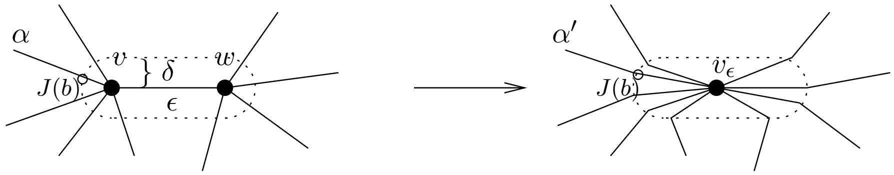


Figure 3.3: Drawing the edges incident to v_ϵ

We now create a drawing φ' of G/ϵ. For all vertices and edges that are in both G and G/ϵ, let φ' be the same as φ, and let $\varphi'(v_\epsilon)$ be the midpoint of $\varphi(\epsilon)$.

Now we must define $\varphi'(\alpha')$ for each edge α' incident to v_ϵ in G/ϵ. Let α be the edge in G corresponding to α', and let $J = \varphi(\alpha)$. Without loss of generality, we can assume that α is incident to v, and that $J(1)$ is the endpoint of J corresponding to v. We split J into two parts: the part that is drawn in $\overline{\mathcal{R}}$ and the part that is drawn in $\mathcal{R}$. Let b be such that $J(b)$ is at a distance of exactly δ away from ϵ. In other words, $J(b)$ is the boundary between these two parts. We then define $J' = \varphi'(\alpha')$ by letting $J'(c) = J(c)$ for all $c \leq b$. Hence the part of the curve that is in $\overline{\mathcal{R}}$ remains the same. For $c > b$, we define $J(c)$ to be the line from $J(b)$ to $\varphi'(v_\epsilon)$. This procedure is illustrated in Figure 3.3.

We can see that this procedure will produce a planar drawing. The portion of the drawing that is more than δ away from ϵ remains the same, so if the original drawing was planar, the new drawing is planar as well. The portion that is less than δ away from ϵ is composed of straight lines that all meet at $\varphi'(v_\epsilon)$, therefore if any two lines intersect at any other point, they must be colinear. But if they are colinear, this implies that they intersect at a point that is δ away from ϵ, which cannot happen if the original drawing was planar.

This procedure also produces a drawing that corresponds to Γ/ϵ. The drawing remains the same around all vertices that are in both G and G/ϵ, so we need only to consider the edges drawn around v_ϵ. The order of the edges drawn around v_ϵ is defined by the order of the edges around the area that is δ away from ϵ. This order is defined by the order of the edges around v and w, which gives us the order specified in Γ/ϵ. Since this drawing is planar and corresponds to Γ/ϵ, we have shown that Γ/ϵ is a planar embedding. $\square$

3.3 Edge reversal

In this section, we discuss the effect of reversing the orientations of edges, and define some notation. First, we define notation to represent reversal of all the edges in a graph and reversal of a single edge. Then we will show that a graph G is upward planar if and only if the graph formed by reversing all the edges in G is upward planar. This result is used in Chapter 4 to take advantage of vertical symmetry.

Definition 3.5. We denote by $\overleftarrow{G}$ the graph G in which the orientation of every edge is reversed. In other words, $V(\overleftarrow{G}) = V(G)$, and $E(\overleftarrow{G}) = \{(w,v)|(v,w) \in E(G)\}$.

Definition 3.6. We denote by $G_{\overleftarrow{\epsilon}}$ the graph G in which the orientation of $\epsilon = (v,w)$ is reversed. In other words, $V(G_{\overleftarrow{\epsilon}}) = V(G)$, and $E(G_{\overleftarrow{\epsilon}}) = E(G) \cup \{(w,v)\} - \{\epsilon\}$

Lemma 3.6. *The graph G is upward planar if and only if $\overleftarrow{G}$ is upward planar.*

Proof. Since $G = \overleftarrow{\overleftarrow{G}}$, it suffices to show that if G is upward planar, so is $\overleftarrow{G}$. Let G be upward planar with upward planar drawing φ. Now, if we reverse the orientation of all the edges in G, all the edges will be pointing downwards in φ. Next, we apply the flip_x transformation on φ, which flips the drawing about the x axis and causes all the edges to now be pointing upwards. Thus we have created an upward planar drawing for $\overleftarrow{G}$, and hence $\overleftarrow{G}$ is upward planar. $\square$

Chapter 4

Edge contraction and upward planarity

In this chapter, we investigate contracting an edge ϵ in an upward planar graph G that has upward planar embedding Γ, and consider the conditions under which the embedding Γ/ϵ of G/ϵ derived from Γ, as defined in Definition 3.4, is still upward planar. This will help us to "join together" two upward planar graphs in the next chapter by drawing each graph separately, drawing an edge between the graphs, and contracting the edge.

Throughout this section, we let G be an upward planar graph with upward planar embedding Γ, and $\epsilon = (s, t)$ will be the edge that we wish to contract.

Note that if s has degree one, then contracting ϵ has the same effect as removing s from the graph. A subgraph of an upward planar graph is also upward planar, so G/ϵ is upward planar. Similarly, if t has degree one, G/ϵ is upward planar. Therefore we can assume from now on that both s and t have degree greater than one.

Since s and t have degree greater than one, we will consider the edge-ordering neighbours of ϵ. Throughout this section, we will use the following labels. Let α and β be the clockwise and counterclockwise neighbours, respectively, of ϵ around t in the embedding Γ, and let γ and δ be the counterclockwise and clockwise neighbours of ϵ around s (Figure 4.1). Let $a \neq t$ be an endpoint of α, $b \neq t$ be an endpoint of β, $c \neq s$ be an endpoint of γ, and $d \neq s$ be an endpoint of δ.

As shorthand in this section, if α or β is oriented towards t, we say that it is *oriented inwards*, and similarly for when γ or δ is oriented towards s. If α or β is oriented away from s, we say that it is *oriented outwards*, and similarly for γ and δ.

Since ϵ is an arbitrary edge, we must consider all possibilities with respect to the orientations of the edges α, β, γ, and δ. First, we will consider the case where α and β are both oriented outwards, and we will use symmetry for the case where γ and δ are both oriented inwards.

In the proof of Lemma 1.1 by Hutton and Lubiw, the embedding in the resulting graph is derived from Γ, so we can strengthen the statement of the lemma.

Theorem 4.1. *If* $\deg^-(t) = 1$*, then* G/ϵ *is upward planar with upward planar embedding* Γ/ϵ.

This handles the case for when α and β are both oriented outwards: if α and β are

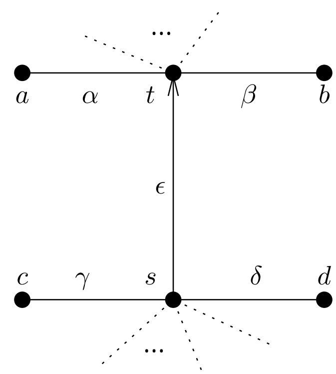


Figure 4.1: The vertices around the edge ϵ

oriented outwards, then by Theorem 1.2, all the edges between α and β in the clockwise ordering around t must be outgoing edges, and hence $\deg^-(t) = 1$. Note that in this case, the orientations of γ and δ do not matter. Now, by symmetry, we also have the following corollary, which handles the case for when γ and δ are both oriented inwards:

Corollary 4.1. *If* $\deg^+(s) = 1$*, then* G/ϵ *is upward planar with upward planar embedding* Γ/ϵ*.*

Proof. From Lemma 3.6, we know that if we reverse the directions of all the edges in G, the resulting graph $H = \overleftarrow{G}$ is also upward planar. By reversing all of the edge directions, all the incoming edges to s become outgoing edges, and vice versa, so we now have $\deg^-(s) = 1$. Therefore, by Theorem 4.1, contracting the edge ϵ results in a graph $K = H/\epsilon$ that is upward planar with planar embedding Γ/ϵ. If we reverse the direction of all the edges again, the resulting graph $\overleftarrow{K}$ is upward planar.

The graph $\overleftarrow{K}$ was obtained by reversing the orientation of the edges in G, contracting ϵ, and reversing the edges in the resulting graph. We now show that contracting ϵ has no effect on the edge directions, and that it does not matter if we contract ϵ before or after we reverse the other edges. Certainly, it will have no effect on edges that are not incident to s or t, thus we need only consider edges incident to s or t. If we have the edge (v, s) for some vertex v, but no edge between v and t, then if we contract ϵ, we get the edge (v, v_ϵ). If we then reverse all the edges, this gives us the edge (v_ϵ, v). If, instead, we reverse all the edges before contracting ϵ, we obtain the same edge (v_ϵ, v).

Now if we have both the edges (v, s) and (v, t), then contracting ϵ gives us the edge (v, v_ϵ), and reversing this gives (v_ϵ, v). However, if we reverse the edges (v, s) and (v, t), this gives us the edges (s, v) and (t, v), and contracting ϵ will again produce the edge (v_ϵ, v). We can handle the case where we have both the edges (s, v) and (t, v) similarly.

Finally, we consider the case where we have the edges (s, v) and (v, t). If we contract ϵ, this gives us the edges (v_ϵ, v) and (v, v_ϵ), and reversing both the edges gives us the same set of edges. If we first reverse (s, v) and (v, t), this gives us the edges (v, s) and

(t, v), and contracting ϵ gives us (v, v_ϵ) and (v_ϵ, v). Hence in all cases, if we contract ϵ before we reverse the other edges of the graph, we obtain the same result as if we reversed the edges of the graph before we contracted ϵ.

Thus it does not matter if we contract ϵ before or after we reverse the other edges, and so we can obtain the same graph by reversing the edges in G, reversing the edges again, and contracting ϵ. Reversing the edges twice yields the original graph, which means that $\overleftarrow{K} = G/\epsilon$, and hence since $\overleftarrow{K}$ is upward planar, so is G/ϵ. □

Next, we look at the case where α and γ are oriented outwards, and β and δ are oriented inwards.

In order to use some of the theorems proved in other papers, we wish to turn G into a single-source graph. To do this, we show that if G is not a single-source graph, we can find a single-source graph H that has G as a subgraph, and that has the same orientations of the edge-ordering neighbours of ϵ. Thus to prove that G/ϵ is upward planar, we can show that H/ϵ is upward planar. Since G/ϵ is a subgraph of H/ϵ, we can then conclude that G/ϵ is upward planar.

As it turns out, we can construct H in such a way that α, β, and δ are still edge-ordering neighbours of ϵ. Since γ is oriented outwards, we must ensure then that the counterclockwise neighbour of ϵ around s in H is oriented outward.

Lemma 4.1. *If α and γ are oriented outwards, and β and δ are oriented inwards, there is a single-source graph H that has G as a subgraph, and the following conditions hold:*

1. *α the first edge incident to t clockwise from ϵ in H,*
2. *β is the first edge incident to t counterclockwise from ϵ in H,*
3. *the first edge incident to s counterclockwise from ϵ is oriented outward, and*
4. *δ is the first edge incident to s clockwise from ϵ.*

Proof. The proof of this theorem is similar to the proof of Lemma 1.5 given by Hutton and Lubiw [22]. The proofs differ in the edges that we add to each source of G, since the conditions required for the graph H are different: the requirement in Lemma 1.5 is that every non-source vertex v in G has the same indegree in G as in H.

The idea behind the proof is that we create a new source, which will be the single source of H, and we "resolve" each source, other than our newly created source, by adding an incoming edge (w, v) to each source v such that v is no longer a source; we must select w appropriately so that our requirements hold. By "resolving" each source other than our newly created source, only the newly created source is a source in H.

Since G is upward planar with embedding Γ, let φ be an upward planar drawing of G that corresponds to the embedding Γ and has width w and height h. Without loss of generality, we can assume that the drawing is centred at $(0, 0)$, and that it is a straight line drawing. Now, we add new vertices p, q, l, and r, which are drawn at the coordinates $(0, -2h)$, $(0, 2h)$, $(-2w, 0)$, and $(2w, 0)$, respectively, and add the edges (lines)

(p, l), (p, r), (r, q), and (l, q). So far, we have only added a diamond around the drawing, so the drawing is still upward planar. The vertex p will become the single source of H.

Now to "resolve" each source v other than p, we must add a new incoming edge (w, v) so that v will no longer be a source; we now show how to select w. We want to choose w so that when we draw the edge (w, v), it will be drawn "between" two outgoing edges of w. This will ensure that if (w, v) is a new edge-ordering neighbour of ϵ, it will be the counterclockwise neighbour of ϵ around s and will be an outgoing edge, which ensures that our required conditions hold. Thus the properties that we want for w are that it shares a common face with v, and that its two incident edges on this face are both outgoing edges. We call a vertex with these properties a *target* vertex.

In the drawing φ, we draw starting from v a vertical ray j downwards parallel to the y axis until it intersects a vertex or an edge. Since v is inside the diamond as specified above, j will eventually intersect either (p, l) or (p, r) if it does not intersect any edge or vertex from G. Let F be the face in which j is drawn. First, we consider the case where j intersects a vertex w_1, and define a process for finding a target vertex w. If w_1 is not a target vertex, we let ϵ_1 be an incoming edge of w_1 on the face F, and w_2 be the tail of ϵ_1. We then check if w_2 is a target vertex, and if not, we let ϵ_2 be an incoming edge of w_2 on the face F, and w_3 be the tail of ϵ_3. We continue this process of following incoming edges until we find a target vertex. The following is a pseudocode description of our process:

> $i \leftarrow 1$
> **while** (w_i is not a target vertex)
> $\epsilon_i \leftarrow$ an incoming edge of w_i on the face F
> $w_{i+1} \leftarrow$ the tail of ϵ_i
> **return** w_i

We know that we will eventually reach a target vertex, otherwise we will eventually arrive back at w_1 and we will have found that F is a directed cycle, contradicting the upward planarity of G. If j instead intersects an edge, we let w_1 be its tail, and we go through the same process as if j intersected w_1.

We can show the resulting graph is upward planar by demonstrating how to draw the edge (w_n, v) such that there are no edge crossings. We do this by drawing (w_n, v) "close to" the path $E = (\epsilon_1, \ldots, \epsilon_{n-1})$. We want to find a number $\zeta > 0$ such that no other edges or vertices from F are drawn within a distance of ζ from the path E. We can find such a ζ by, for example, taking ζ to be half the minimum distance from any edge from E to any vertex or edge on F, other than those that are part of E. We then draw the vertical line segment k_0 from v to the point that is ζ above ϵ_1. Then, for each edge ϵ_i, $i < n - 1$, we draw the line segment k_i parallel to ϵ_i at a distance of ζ away from ϵ_i. (We may have to increase or decrease the length of k_i so that one of its endpoints is the same as an endpoint of k_{i-1}, and the other is the same as an endpoint of k_{i+1}.) Rather than drawing the last segment k_{n-1} parallel to ϵ_{n-1}, we instead draw it as a straight segment from the endpoint of k_{n-2} to w_n. Since each segment k_i, $i > 0$ is drawn within a distance of ζ from E, by definition of ζ, k_i will not cross any edge of G. As well, k_0 is a segment

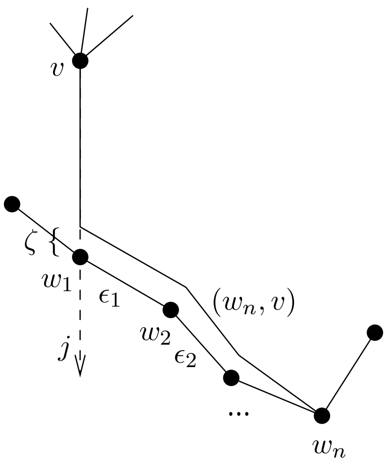


Figure 4.2: "Resolving" the source v

within the ray j and does not include the first point at which j crosses a vertex or an edge, hence k_0 does not cross any edge of G. Therefore we have drawn (w_n, v) such that there are no edge crossings. This process is illustrated in Figure 4.2.

From this procedure, we see that when we add the edge (w_n, v), its edge-ordering neighbours around w_n are both outgoing edges. Neither s nor t is a source, so if we add any edges (w_n, v) incident to s or t, s (or t) must be fulfilling the role of w_n, and hence the edge-ordering neighbours around s (or t) must both be outgoing edges. Of the edges α, β, γ, δ, and ϵ, the only pair of adjacent outgoing edges are γ and ϵ. Thus if we add a new edge that is a neighbour of ϵ, it must be between γ and ϵ in the clockwise ordering around s. This edge will be an outgoing edge; therefore our required conditions hold. $\square$

Theorem 4.2. *If the edges α and γ are oriented outwards, and β and δ are oriented inwards, then G/ϵ is upward planar with upward planar embedding Γ/ϵ.*

Proof. By Lemma 4.1, we can assume without loss of generality that G is a single-source graph: since G is upward planar, we can use Lemma 4.1 to find a single source graph H that has G as a subgraph, and in which the edge-ordering neighbours of ϵ have the same orientations as in G. After contracting the edge ϵ, we obtain the upward planar graph H/ϵ that has G/ϵ as a subgraph. Since H/ϵ is upward planar, so is G/ϵ.

Since we can assume that G is a single-source graph, we can then use Theorem 1.5 to show that G/ϵ is upward planar as follows. Since φ is an upward planar drawing, we know by Theorem 1.5 that for every vertex $v \in V(G)$, v is a sink on the outer face of φ_v, the drawing induced on $P_G(v)$. To show that G/ϵ is upward planar, we must produce a planar drawing φ' corresponding to Γ/ϵ such that for every $v \in V(G/\epsilon)$, v is a sink on the outer face of φ'_v. We will let φ' be the drawing of G/ϵ that we produce in the proof of Lemma 3.5. This drawing may not be upward planar, but by Theorem 1.5, there is an upward planar drawing strongly equivalent to φ' if G/ϵ does not have a directed cycle and every vertex v is a sink on the outer face of φ'_v. We will first show that v is a sink in φ'_v, that G/ϵ does not have a directed cycle, and finally that v is on the outer face of φ'_v.

Since Γ is an upward planar embedding of G, let φ be an upward planar drawing of G that corresponds to Γ. Without loss of generality, we can assume by Theorem 1.3 that the drawing is a straight line drawing. We can also assume that ϵ is drawn vertically: if not, we can apply the skew transformation appropriately, as described in Remark 3.1, to produce a drawing in which ϵ is drawn vertically.

For each vertex v of G, exactly one of the following is true:

1. $v = s$ or $v = t$,
2. v is a successor of t,
3. v is a successor of s and not a successor of t, or
4. v is not a successor of s.

When we contract ϵ, one of four things can happen to $P_{G/\epsilon}(v)$ for a given v in $V(G)$, corresponding to the four situations listed above:

1. v is removed from the graph if v is s or t, and is replaced with v_ϵ. In this case, $v \notin V(G/\epsilon)$, so we do not need to consider v;
2. s and t are removed from $P_{G/\epsilon}(v)$ and replaced with v_ϵ;
3. v_ϵ and the contents of $P_G(t) - \{s,t\}$ are added to $P_{G/\epsilon}(v)$, and s is removed; or
4. $P_{G/\epsilon}(v)$ is the same as $P_G(v)$.

We then need to show that in each case, v is a sink on the outer face of φ'_v. If (4) occurs, v will still be a sink on the outer face of $\varphi'_v = \varphi_v$. If (2) occurs, φ'_v will not have any extra edges, so v will still be a sink, and φ'_v will not have any new faces, so v will still be on the outer face. Note that for the new vertex, v_ϵ, $P_{G/\epsilon}(v_\epsilon)$ is the same as $P_G(t)$, but with s and t removed, and replaced with v_ϵ, so this is handled by case (2). Now all that remains to be shown is that if (3) occurs, v must also be a sink on the outer face of φ'_v.

First, suppose that v is not a sink. Since v was a sink in φ_v but is not in φ'_v, and since the vertices of φ'_v are $P_G(v) \cup P_G(t) \cup \{v_\epsilon\} - \{s,t\}$, it must have a successor from $P_G(t) \cup \{v_\epsilon\} - \{s,t\}$. In other words, there is a vertex $w \in P_G(t) \cup \{v_\epsilon\} - \{s,t\}$, $w \notin P_G(v)$, such that (v,w) is an edge in G. Since w is a predecessor of t and v is a predecessor of w, v is also a predecessor of t.

Now we consider where the predecessors of t and successors of s can lie on the plane. We will then use this to show that the edge (v,w) cannot exist by showing that there is no region in which w can be drawn. We define four regions on the plane, illustrated in Figure 4.3. Note that some of these regions overlap.

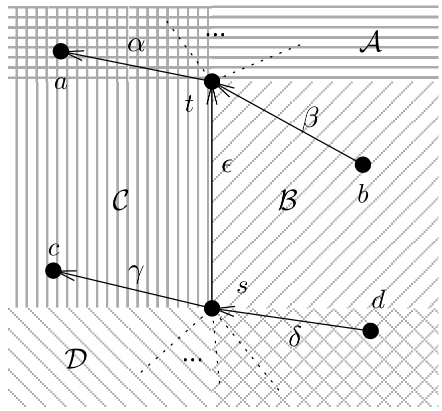


Figure 4.3: Regions in which vertices can lie

1. region $\mathcal{A}$ is the set of points that have larger y coordinate than the vertex t, in other words, the region above t;
2. region $\mathcal{B}$ is the set of points that have larger x coordinate, and smaller y coordinate than the vertex t, in other words, the region below and to the right of t;
3. region $\mathcal{C}$ is the set of points that have smaller x coordinate, and a larger y coordinate than the vertex s, in other words, the region above and to the left of s; and
4. region $\mathcal{D}$ is the set of points that have smaller y coordinate than the vertex s, in other words, the region below s.

We now show that all predecessors of t other than s must be in regions $\mathcal{B}$ or $\mathcal{D}$. Since α is oriented outwards and φ is upward planar, a must be above t, in region $\mathcal{A}$. Since β is oriented inwards, b must be below t, and b must also be to the right of ϵ as a consequence of a being the clockwise neighbour of ϵ around t. Therefore b must be in region $\mathcal{B}$. Similarly, all vertices dominated by t, except s, must be in region $\mathcal{B}$. Since φ is planar, no edges can cross (s, t), and so all predecessors of t, except s, must be to the right of (s, t), or below s, in other words, in regions $\mathcal{B}$ or $\mathcal{D}$.

We can use similar reasoning to show that all successors of s other that t must be in regions $\mathcal{A}$ or $\mathcal{C}$. The vertex d must be in region $\mathcal{D}$. All outgoing edges of s must be drawn above s, and must be to the left of ϵ as a consequence of δ being the clockwise neighbour of ϵ around s. Therefore c must be in the region $\mathcal{C}$, as must all vertices other than s that dominate t. Since no edges can cross (s, t), all successors of s other than t must be in regions $\mathcal{C}$ or $\mathcal{A}$.

Now since v is a predecessor of t, it must be in either $\mathcal{B}$ or $\mathcal{D}$. At the same time, v is a successor of s, and hence must be in either $\mathcal{C}$ or $\mathcal{A}$. However, $\mathcal{B} \cup \mathcal{D}$ does not overlap with $\mathcal{C} \cup \mathcal{A}$. Hence there is no place on the plane in which v can lie. Thus our assumption that there is an edge (v, w) in ϕ'_v is incorrect, so v must be a sink.

Using this same reasoning, we can show that G/ϵ has no directed cycles. We already know that G is acyclic, since it is upward planar. Thus if G/ϵ has a directed cycle, it must have been created by a directed path P from s to t other than the path that consists only of the edge ϵ: after contracting ϵ, the endpoints of this path will be identified, giving us our cycle. If we let $v \notin \{s, t\}$ be a vertex on P, then v is a successor of s and a predecessor of t. However, this means that v must be either in $\mathcal{B}$ or $\mathcal{D}$, and at the same time in either $\mathcal{C}$ or $\mathcal{A}$, which we have seen is impossible. Therefore such a path is not possible, and hence G/ϵ is acyclic.

We will now show that v is not on the outer face of φ'_v. If not, then let F be the outer face of φ'_v. Since v is not on the outer face, F surrounds v. By using the same set of edges as F, and possibly adding ϵ if F includes edges incident to v_ϵ, this corresponds to a cycle Σ in φ that surrounds v. Since all the vertices from φ'_v are from $P_{G/\epsilon}(v) \cup P_{G/\epsilon}(t) \cup \{v_\epsilon\}$, Σ consists only of vertices of $P_G(v) \cup P_G(t)$. We then consider the drawing of Σ in φ, and we will show that Σ cannot exist.

Since Σ surrounds v in the drawing φ, there is at least one edge (v_1, v_2) of Σ that is drawn such that there is a point p on the drawing of (v_1, v_2) that is above and to the left of v. Since the drawing is upward, v_2 must be above v, thus it cannot be a predecessor of v. But $v_2 \in P_G(v) \cup P_G(t)$, so v_2 must be a predecessor of t. Note that v_2 cannot be t: all incoming edges to t must be drawn in region $\mathcal{B}$, whereas v is a successor of s, and so must be in region $\mathcal{A}$ or $\mathcal{C}$. Thus if (v_1, v_2) is in $\mathcal{B}$ and v is in $\mathcal{A}$ or $\mathcal{C}$, no point of the edge (v_1, v_2) can be drawn above and to the left of v. As well, since v must be in the region $\mathcal{A}$ or $\mathcal{C}$, and p is above and to the left of v, p must also be in $\mathcal{A}$ or $\mathcal{C}$. But v_2 is a predecessor of t, and hence must be in $\mathcal{B}$ or $\mathcal{D}$. Hence any curve drawn from p to v_2 must either be downward, or cross ϵ, contradicting the upward planarity of ϕ. Therefore the cycle Σ cannot exist, and hence v must be on the outer face.

Since in all cases we have shown that v is a sink on the outer face of $(\Gamma/\epsilon)_v$, we know from Theorem 1.5 that G/ϵ is upward planar. □

Using left-right symmetry, we have the following corollary:

Corollary 4.2. *If the edges α and γ are oriented inwards and β and δ are oriented outwards, then G/ϵ is upward planar with upward planar embedding Γ/ϵ.*

In the above cases, we showed that contracting ϵ always yields an upward planar graph. We will now show two conditions under which our desired embedding Γ/ϵ is not upward planar.

Theorem 4.3. *If α and δ are oriented inwards and β and γ are oriented outwards, Γ/ϵ is not an upward planar embedding.*

Proof. If we look at the order of edges around v_ϵ in Γ/ϵ, starting with α, and going clockwise, we see that we have α, an incoming edge, followed by some number of edges of unknown orientation, followed by β, an outgoing edge, followed by δ, an incoming edge, followed by some number of edges of unknown orientation, followed by γ, an outgoing edge. Thus the outgoing edges of v_ϵ do not appear consecutively around v_ϵ, and hence by Theorem 1.2 Γ/ϵ is not an upward planar embedding. □

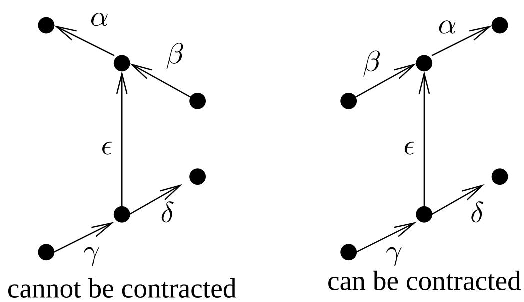


Figure 4.4: Examples of embeddings in which ϵ can or cannot be contracted

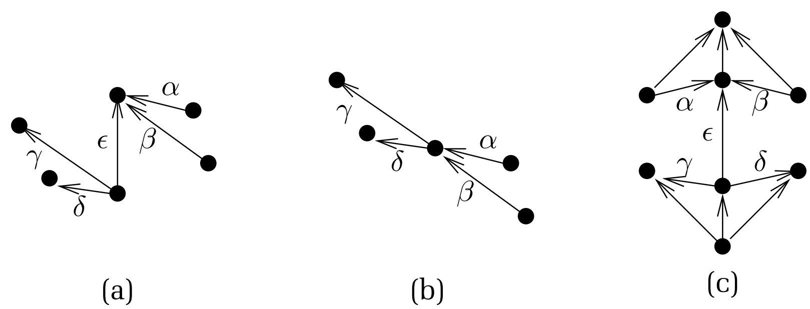


Figure 4.5: α and β are oriented inwards and γ and δ are oriented outwards

And by symmetry, we have the following:

Corollary 4.3. *If α and δ are oriented outwards and β and γ are oriented inwards, Γ/ϵ is not an upward planar embedding.*

Note, however, that in some cases, G/ϵ may be upward planar under a different embedding. For example, consider the graph and the embedding illustrated in Figure 4.4a. In this case, the derived embedding is not an upward planar embedding. However, if we change the embedding to that illustrated in 4.4b, the derived embedding is an upward planar embedding.

In the remaining five cases for the orientations of α, β, γ, and δ (see Table 4.1), Γ/ϵ may or may not be an upward planar embedding. For example, for the graph and embedding in 4.5a, the derived embedding is an upward planar embedding as shown in 4.5b. However, for the graph and embedding in 4.5c, the derived embedding is not an upward planar embedding (although G/ϵ is upward planar with a different embedding) since the derived embedding would violate Theorem 1.2.

However, we can show that if we reverse the direction of the edge ϵ, and the resulting graph $G_{\overleftarrow{\epsilon}}$ is still upward planar with the same embedding, then Γ/ϵ is an upward planar embedding. Two of the remaining cases will be shown as corollaries; the proofs for the remaining three cases are similar, so we only prove the first case, and give the changes needed to prove the last two cases.

Theorem 4.4. *If α, β, and γ are oriented inwards and δ is oriented outwards, G/ϵ is upward planar with embedding Γ/ϵ if and only if $G_{\overleftarrow{\epsilon}}$ is upward planar with embedding Γ.*

Proof. First, we show that if G/ϵ is upward planar with embedding Γ/ϵ, then $G_{\overleftarrow{\epsilon}}$ is upward planar with embedding Γ. Let G/ϵ be upward planar with embedding Γ/ϵ. We can obtain $G_{\overleftarrow{\epsilon}}$ and embedding Γ by splitting the vertex v_ϵ into two vertices, s and t, such that the edges incident to each of s and t are consecutive in the ordering of the edges of v_ϵ in Γ, and by adding the edge (t, s). In their proof of Lemma 1.4, Hutton and Lubiw [22] show that this graph is upward planar given that Γ/ϵ is upward planar.

Now we show that if $G_{\overleftarrow{\epsilon}}$ is upward planar with embedding Γ, G/ϵ is upward planar with embedding Γ/ϵ. Let $G_{\overleftarrow{\epsilon}}$ be upward planar with embedding Γ. Now, if we look at ϵ in $G_{\overleftarrow{\epsilon}}$, we see that we now have the conditions required for applying Corollary 4.1, and hence $G_{\overleftarrow{\epsilon}}/\epsilon = G/\epsilon$ is upward planar with embedding Γ/ϵ. □

Theorem 4.5. *If α is oriented inwards and β, γ, and δ are oriented outwards, G/ϵ is upward planar with embedding Γ/ϵ if and only if $G_{\overleftarrow{\epsilon}}$ is upward planar with embedding Γ.*

Proof. The forward direction is proved as in the proof of Theorem 4.4.

Since γ and δ are oriented outwards, we use Theorem 4.1 rather than Corollary 4.1 for the reverse direction. □

Theorem 4.6. *If α and β are oriented inwards and γ and δ are oriented outwards, G/ϵ is upward planar with embedding Γ/ϵ if and only if $G_{\overleftarrow{\epsilon}}$ is upward planar with embedding Γ.*

Proof. Again, the forward direction is proved as in the proof of Theorem 4.4.

Since α and β are oriented inwards, the reverse direction can be proved as in Theorem 4.4 as well. Alternately, we can also use Theorem 4.1 rather than Corollary 4.1 since γ and δ are oriented outwards. □

Again, we have the following corollaries by symmetry:

Corollary 4.4. *If α, β, and δ are oriented inwards, and γ is oriented outwards, G/ϵ is upward planar with embedding Γ/ϵ if and only if $G_{\overleftarrow{\epsilon}}$ is upward planar with embedding Γ.*

Corollary 4.5. *If α, γ, and δ are oriented outwards, and β is oriented inwards, G/ϵ is upward planar with embedding Γ/ϵ if and only if $G_{\overleftarrow{\epsilon}}$ is upward planar with embedding Γ.*

Using Theorem 1.2, we can obtain a necessary condition for Γ/ϵ to be an upward planar embedding in these last cases.

Corollary 4.6. *If γ and δ are oriented outwards, at least one of α or β are oriented inwards, and Γ/ϵ is an upward planar embedding, then* $\deg^-(s) = 0$.

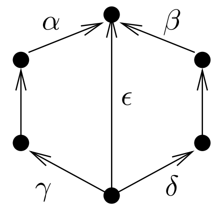
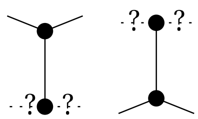
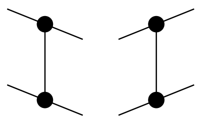
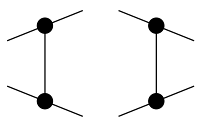
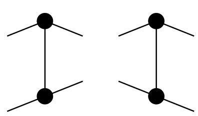
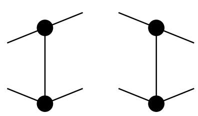
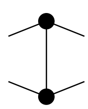


Figure 4.6: G/ϵ is not upward planar

Corollary 4.7. *If α and β are oriented inwards, at least one of γ or δ are oriented outwards, and Γ/ϵ is an upward planar embedding, then $\deg^+(t) = 0$.*

Note, however, that these conditions are not sufficient. Figure 4.6 shows a graph in which $\deg^-(s) = 0$ and $\deg^+(t) = 0$, but Γ/ϵ is not upward planar, since G/ϵ will have a directed cycle.

The results from this chapter are summarized in Table 4.1.

orientation of edges	Is Γ/ϵ upward planar?	reference
	yes	Thm. 4.1/Corol. 4.1
	yes	Thm. 4.2/Corol. 4.2
	no	Thm. 4.3/Corol. 4.3
	$\overleftarrow{\epsilon}$	Thm. 4.4/Corol. 4.4
	$\overleftarrow{\epsilon}$	Thm. 4.5/Corol. 4.5
	$\overleftarrow{\epsilon}$	Thm. 4.6

We represent the orientation of the edges graphically. ·?·· means that the orientation for this edge does not matter. $\overleftarrow{\epsilon}$ means that Γ/ϵ is upward planar if and only if $G_{\overleftarrow{\epsilon}}$ is upward planar with embedding Γ.

Table 4.1: The effect of the orientations of α, β, γ, and δ on contracting ϵ.

Chapter 5

Joining subgraphs

One way to create an upward planar drawing of a graph G is to split G into several smaller subgraphs, create an upward planar drawing of each subgraph, and combine the drawings of the subgraphs to form a drawing of G. In this section, we will be considering how to do this for the case where each pair of subgraphs shares at most one common vertex. We will also only consider the case where we split G into two subgraphs G_1 and G_2; using induction we can handle the case where G is split into more than two subgraphs.

An equivalent way of looking at this procedure is that we are given two upward planar connected digraphs G_1 and G_2 that do not share any common vertices. We then form the graph G by specifying a vertex v_1 of G_1, and a vertex v_2 of G_2, and identifying them. The same graph can also be obtained by adding the edge (v_1, v_2), and contracting (v_1, v_2). Thus to prove that G is upward planar, we will in general show that we can draw G_1 in a face of G_2, or vice versa, add the edge (v_1, v_2), and contract (v_1, v_2). In order to add (v_1, v_2), we will be defining a notion of visibility of vertices "from above" and "from below".

In this section, we will assume that G_1 and G_2 are connected and are upward planar. When we need to specify embeddings and outer faces for G_1 and G_2, we will use Γ_1 and Γ_2 for the embeddings, respectively, and F_1 and F_2 for the outer faces.

First, we show that v_1 must be on the outer face of G_1 or v_2 must be on the outer face in G_2 in order for G to be upward planar.

Lemma 5.1. *If G_1 does not have an upward planar embedding in which v_1 is on the outer face, and G_2 does not have an upward planar embedding in which v_2 is on the outer face, G is not upward planar.*

Proof. Suppose that G is upward planar. Then consider an upward planar drawing φ of G, and let φ_1 be the drawing induced on G_1 and φ_2 be the drawing induced on G_2. Suppose that v_1 and v_2 are not on the outer face of φ_1 and φ_2. Then the drawings of the outer faces F_1 and F_2 of G_1 and G_2 surround $v_1 = v_2$.

If the drawings of F_1 and F_2 intersect, then either a vertex or edge in F_1 intersects a vertex or edge in F_2, contradicting the planarity of φ, or F_1 and F_2 share a common vertex, contradicting the fact that we constructed G by identifying only one pair of vertices. Therefore F_1 and F_2 do not intersect. Since they both surround $v_1 = v_2$, one of the cycles must be drawn within the other.

Without loss of generality, assume that F_1 is drawn within F_2. Since G_2 is connected, there is a path P from v_2 to F_2. However, this path will cross F_1. As above, P either crosses F_1 at an edge, contradicting the planarity of φ, or P and F_1 share a common vertex, contradicting our method of constructing G.

Therefore, if G is upward planar with drawing φ, then v_1 must be on the outer face of φ_1 or v_2 must be on the outer face of φ_2. □

We will now define a notion of visibility of vertices from different parts of the outer face, namely the area above or below the drawing of G. This, along with the edge contraction results from the previous chapter, will help us to be able to join together the subgraphs G_1 and G_2, using the following procedure: given an upward planar drawing of G_1 and G_2, we draw G_2 in a face of G_1. The face in which we draw G_2 depends on the visibility of v_1 and v_2. We can then draw the edge (v_1, v_2), and contract this edge using our edge contraction results.

Definition 5.1. Given an upward planar graph G with a specified upward planar embedding Γ and outer face F, we say that the vertex v is *visible from above (below)* if there is an upward planar drawing of G corresponding to the specified embedding and outer face such that a monotone curve that does not cross any edges can be drawn from v to a point above (below) the drawing of G.

If a monotone curve J can be drawn from v to a point p above the drawing of G, we can draw a monotone curve from v to any other point q above the drawing of G as follows. If q is above p, then we just need to draw a straight line from p to q. Otherwise, we follow the curve J from v to p, and stop when we reach a point r that has the same y coordinate as the highest point in the drawing of G. From r, we can then draw a straight line to q, as illustrated in Figure 5.1. Since this line is drawn entirely above the drawing of G, it will not intersect any part of the drawing. Therefore our definition of visibility from above is not dependent on the point to which we draw the curve.

If we are given an upward planar drawing and a vertex v that is on the outer face, we may not be able to draw a monotone curve from v to a point that is above the drawing, even though v may be visible from above by using a different drawing that has the same embedding. Figure 5.2 gives an example of this: Figure 5.2a shows a drawing in which we cannot draw a monotone curve from v to p, and Figure 5.2b shows an alternate drawing of the same graph in which we can draw a monotone curve from v to p. Because of this, we want a condition that is equivalent to the condition in the definition of visibility from above, but is less dependent on the specific drawing that is chosen. From this characterization, we will be able to obtain some visibility results based on the planar embedding.

We will show that v is visible from above if we can draw a monotone curve from v to a point p that is above v and in the outer face. This differs from the condition given

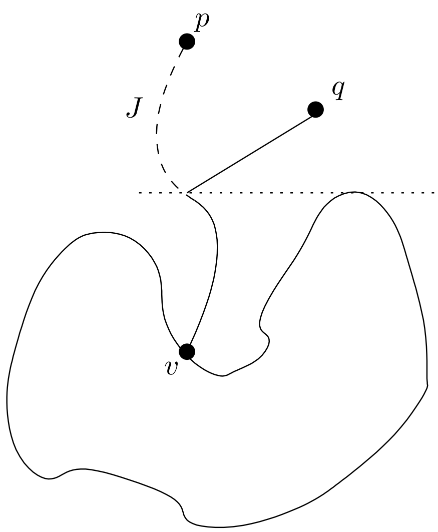


Figure 5.1: Drawing a curve from v to q given a curve from v to p

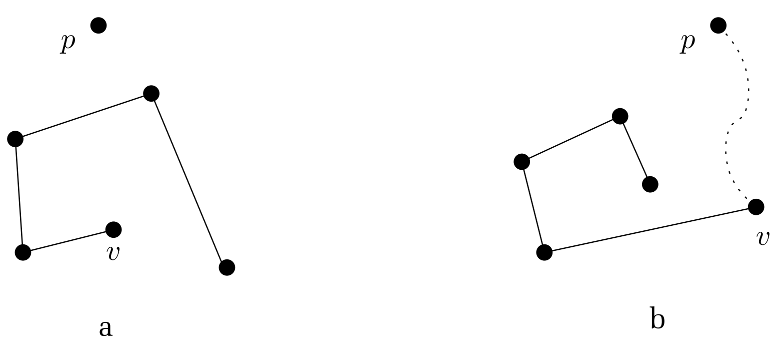


Figure 5.2: v is visible from above, although we cannot draw a monotone curve from v to p in (a)

in the definition since p must be above only v; it does not have to be above the entire drawing of G.

Lemma 5.2. *Given an upward planar graph G and an upward planar drawing φ of G, if a monotone curve that does not cross any edges of G can be drawn from v to a point p that is above (below) v such that p is inside the outer face and not part of the boundary of the outer face, then v is visible from above (below) with respect to the upward planar embedding of φ.*

Proof. To prove this, we will draw a horizontal ray ℓ from p and count how many times ℓ crosses the boundary of the outer face. If ℓ does not cross the boundary of the outer face, we will show how to draw a curve from p to a point above G. Otherwise, we will show, by induction on the number of crossings, that we can redraw G so that ℓ no longer crosses the boundary.

If p is already above the entire drawing of G, then we are done, as we already have the condition in our definition of visibility from above. Otherwise, we draw a horizontal ray ℓ starting at p. If p is to the right of v, we draw ℓ to the right of p, and if p is to the left of v, we draw ℓ to the left. If p has the same x coordinate as v, we can draw ℓ either to left or to the right. Using left-right symmetry, we can assume without loss of generality that ℓ is drawn to the left of p. Note that since p is in the outer face, this ray will cross the boundary of the outer face an even number of times. On the odd-numbered crossings, ℓ moves from the outer face to an interior face; on the even-numbered crossings, ℓ moves from an interior face to the outer face.

We want to ensure that ℓ does not intersect the drawing of a vertex, so we want to show that if ℓ does intersect a vertex, we can move p slightly up or slightly down until ℓ does not intersect any vertices. We can do this since p is inside the outer face but not part of its boundary, and hence there is an open ball around p that is entirely within the outer face. Hence we can move p anywhere within this ball and it will still be inside the outer face, and so this will not affect the result. Since there is only a finite number of vertices, but there are an infinite number of vertical coordinates to which we can move p, we can move p so that ℓ no longer intersects any vertex. Therefore we can assume without loss of generality that ℓ does not intersect any vertices.

We first consider the case where ℓ does not cross the boundary of the outer face. In this case, we have two possibilities. If no part of the graph is drawn above ℓ and to the left of p, then we can draw a ray from p going straight up, until we reach a point above the drawing φ, and we are now done as we have satisfied the condition in our definition of visibility from above. Otherwise, we will construct a curve from p to a point above the drawing φ. To do this, we define the point q as the point of the drawing that is closest to ℓ and that is above ℓ and to the left of p, as illustrated in Figure 5.3. We then let δ be half the distance from q to ℓ, and select a point r that is δ above ℓ and to the left of the leftmost point of the drawing φ. Now we can draw a line segment π from p to r, and a ray ρ from r going straight up. Since no part of φ is drawn within a distance of δ above ℓ, π does not intersect φ, and since ρ is drawn to the left of φ, it will not intersect φ either.

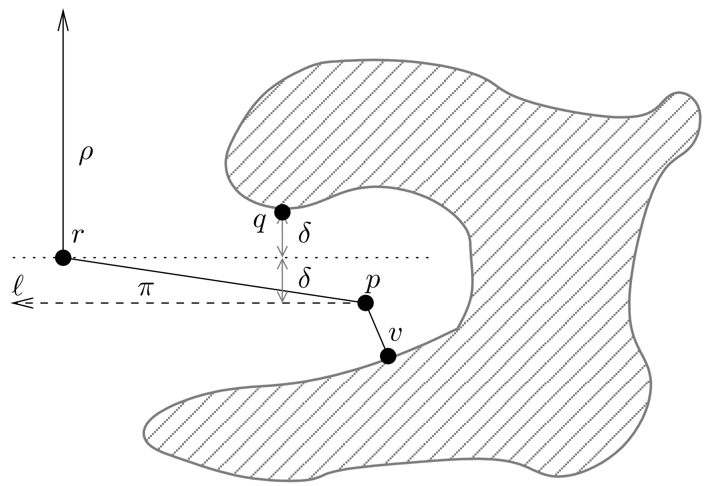


Figure 5.3: The ray ℓ does not cross the boundary of the outer face of G

We will now prove by induction on the number of crossings that we can modify φ so that ℓ does not cross the boundary of the outer face. Our base case is when ℓ crosses the outer face zero times, in which case we are done.

Now, suppose that ℓ crosses boundary of the outer face $k > 0$ times. We label the points at which ℓ crosses the boundary of the outer face $q_1, \ldots, q_k$, numbered from left to right. As noted above, ℓ will cross the outer face an even number of times, so if it crosses the outer face at least once, it crosses the outer face at least twice. As well, at each point q_{2i}, ℓ goes from the outer face to an interior face when going from right to left, and at each point q_{2i+1}, ℓ goes from an interior face to the outer face. Thus the line segment between q_i and q_{i+1} is contained within the outer face if i is even, and is not within the outer face if i is odd.

If we draw a line segment between q_1 and q_2, this separates the drawing φ into two pieces, where two points r_1 and r_2 in φ are part of the same piece if there is a path in φ from r_1 to r_2 that does not cross the line segment (q_1, q_2). Of these two pieces, we label with $\mathcal{A}$ the piece that does not contain v. This is illustrated in Figure 5.4.

The idea for the induction is that we wish to remove the crossing points q_1 and q_2 by "shrinking" $\mathcal{A}$ so that it fits completely within a region above or below ℓ. By "shrinking", we mean that we will apply the scale and translate transformations to $\mathcal{A}$ such that $\mathcal{A}$ is drawn entirely within the given region. To determine into which side of ℓ we will shrink $\mathcal{A}$, we look at the part of $\mathcal{A}$ that is closest to the segment (q_1, q_2). If this part is above ℓ, then shrinking $\mathcal{A}$ into a region below ℓ will remove the crossing points q_1 and q_2, and if this part is below ℓ, then shrinking $\mathcal{A}$ into a region above ℓ will remove the crossing points. Since both cases are symmetrical, we will just consider the case where we shrink $\mathcal{A}$ into a region below ℓ.

We will define a rectangle $\mathcal{R}$ below ℓ; this is the region into which we will shrink $\mathcal{A}$. We label with $\mathcal{B}$ the part of the original drawing that is in $\mathcal{R}$; to ensure that the resulting graph is still upward planar, we will have to shrink $\mathcal{B}$ into a sub-rectangle of

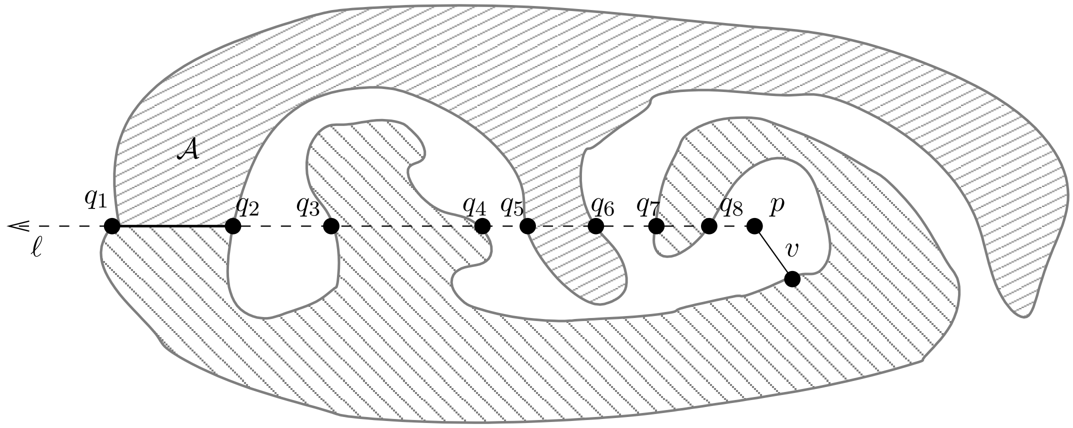


Figure 5.4: Splitting φ into two pieces by drawing the line segment (q_1, q_2).

$\mathcal{R}$. The way in which we shrink $\mathcal{A}$ is illustrated in Figure 5.5. We will now describe our construction in detail.

The conditions that we want for our rectangle $\mathcal{R}$ are as follows.

1. its top edge has the same y coordinate as ℓ;
2. its left edge is to the left of the leftmost point in φ;
3. its right edge is to the right of q_2;
4. all four of its corners are inside the outer face;
5. its left and right edges do not cross the boundary of the outer face of the drawing;
6. the part of the drawing φ that is within $\mathcal{R}$ is a single connected piece; and
7. there are no vertices in $\mathcal{R}$.

We can construct such a rectangle as follows; the process is illustrated in Figure 5.6. To satisfy condition 1, our top edge will be on the ray ℓ. Now consider the third point q_3 from the left at which ℓ crosses the outer face, or the point p if such a point does not exist. We draw a vertical line h_r midway between this point and q_2; our right edge will be on this line, which will satisfy condition 3. We draw a vertical line h_l anywhere to the left of the leftmost point of the drawing of G; our left edge will be on this line, and so condition 2 will be satisfied.

Now in order to satisfy conditions 5 and 6, we must pick a suitable location for the bottom edge. To do this, we define two distances: let d_r be the distance from ℓ to the highest point below ℓ at which h_r crosses the boundary of the outer face, and let d_v be the distance from ℓ to the highest vertex of ϕ below ℓ. If we draw the bottom edge of $\mathcal{R}$ so that it is within d_r of ℓ, then the right edge of $\mathcal{R}$ will not cross the boundary of the outer face, which together with condition 2 will satisfy condition 5. If we draw the

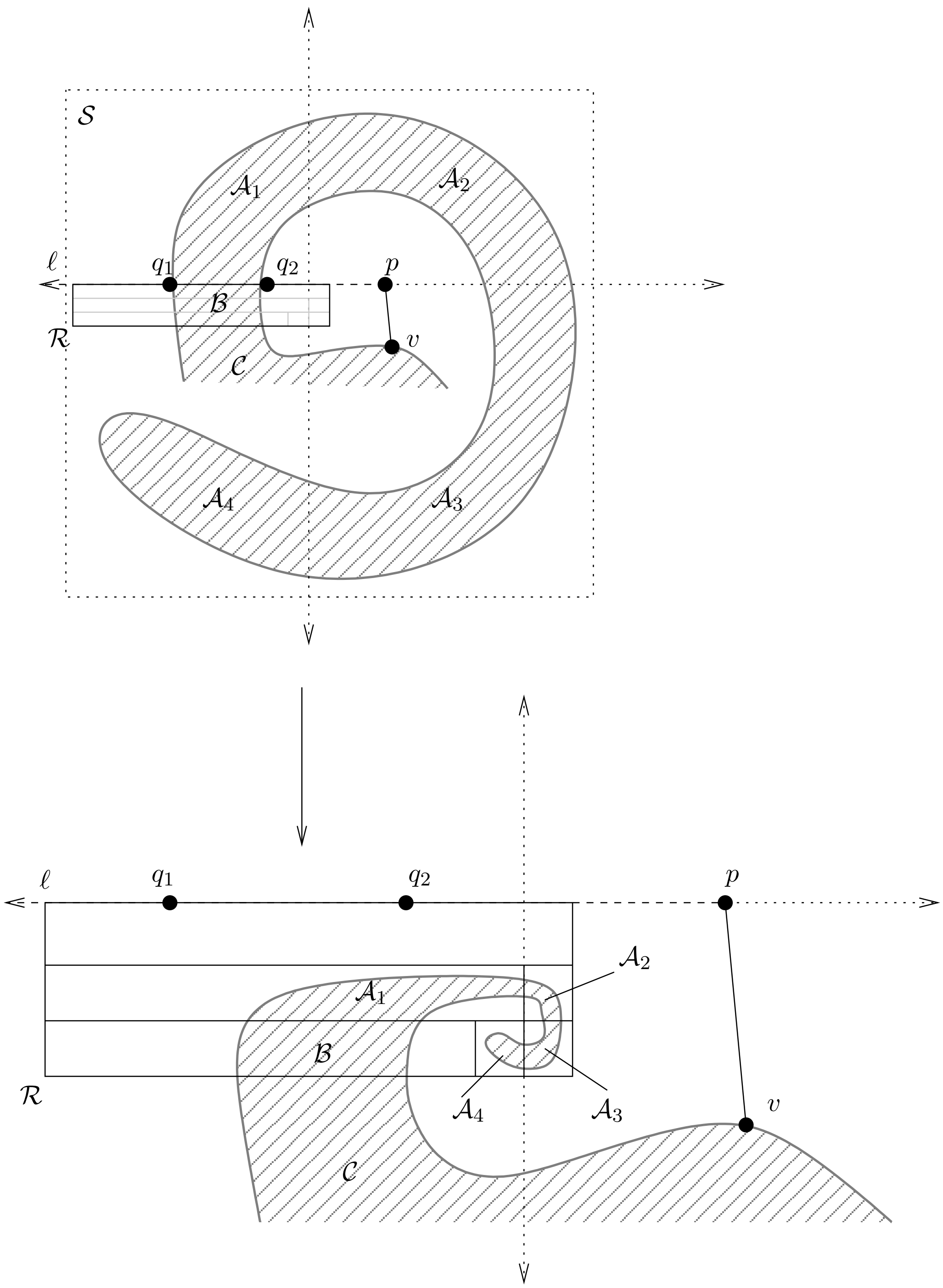


Figure 5.5: Shrinking $\mathcal{A}$ into $\mathcal{R}$. We have magnified the result for clarity. For simplicity, we only show the case where ℓ crosses the boundary of the outer face twice.

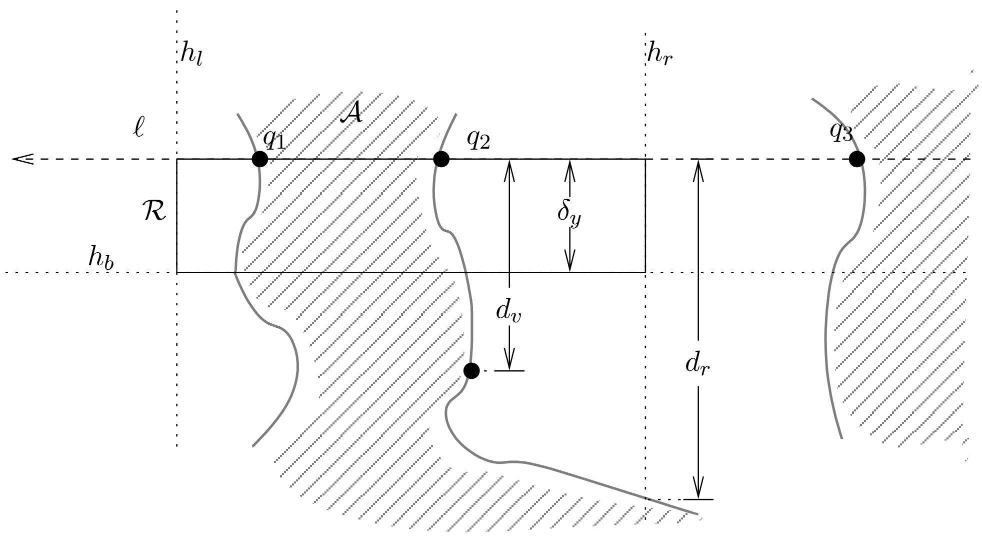


Figure 5.6: Constructing the rectangle $\mathcal{R}$

bottom edge of $\mathcal{R}$ so that it is within d_v of ℓ, then $\mathcal{R}$ will only contain edges, and so conditions 7 will be satisfied. Since the drawing is upward planar, all the edges are monotone, and hence every edge that is drawn in $\mathcal{R}$ must enter $\mathcal{R}$ from the bottom edge of $\mathcal{R}$ and leave from the top edge of $\mathcal{R}$. But by construction, the only place that it can leave $\mathcal{R}$ is between q_1 and q_2, and hence the part of the drawing that is within $\mathcal{R}$ is a single connected piece, satisfying condition 6. Therefore we draw a line h_b, which will define our bottom edge, at a distance of $\delta_y = \frac{1}{2}\min(d_r, d_v)$ below ℓ. Thus the lines h_r, h_l, h_b, and the ray ℓ define the rectangle $\mathcal{R}$. The only remaining condition to be shown is condition 4. The two left corners are within the outer face due to condition 2. The top-right corner of $\mathcal{R}$ is drawn between q_2 and q_3, and hence it is also in the outer face. The right edge of $\mathcal{R}$ does not cross the outer face, so since the top-right corner of $\mathcal{R}$ is inn the outer face, the entire right edge is within the outer face. Thus condition 4 is satisfied.

By construction, no vertices are drawn within $\mathcal{R}$; in particular, v is not within $\mathcal{R}$. As well, no part of $\mathcal{A}$ is in $\mathcal{R}$ since the portion within $\mathcal{R}$ is a single connected piece. Thus we now have partitioned the drawing into three parts: $\mathcal{A}$, as defined earlier; $\mathcal{B}$, the portion of the drawing within $\mathcal{R}$; and $\mathcal{C}$, the remainder of the drawing, which contains v.

We will now split $\mathcal{R}$ into different regions; we will shrink different parts of $\mathcal{A}$ and $\mathcal{B}$ into these regions. Let δ_y be the height of $\mathcal{R}$, as defined previously, and let δ_x be the distance between the right edge of the box and the rightmost part of the drawing within the box. Since the right edge of the box does not cross the outer face, this distance is positive. We divide the box into 6 regions, as illustrated in Figure 5.7. First, we cut the box into thirds horizontally. We call the top third $\mathcal{U}$. The middle third is cut into two parts, cutting at $\frac{\delta_x}{3}$ from the right edge. We call the left part $\mathcal{V}$ and the right part $\mathcal{W}$. Finally, we cut the bottom third into three parts, at $\frac{2\delta_x}{3}$ and $\frac{\delta_x}{3}$ from the right edge.

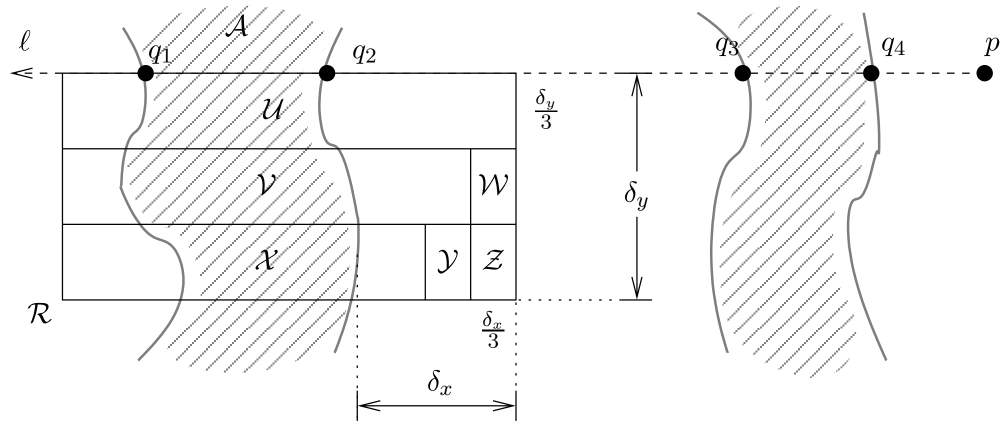


Figure 5.7: Regions of the rectangle $\mathcal{R}$

We call these parts, from left to right, $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$. Note that, by construction and by definition of δ_x, no part of the drawing of G is in $\mathcal{W}$, $\mathcal{Y}$, or $\mathcal{Z}$.

We will now split $\mathcal{A}$ into up to four parts; each of these parts will be shrunk into one of the regions defined above. We draw a vertical line that is $\frac{\delta_x}{3}$ to the left of the box's right edge, and we draw a horizontal line at the same vertical coordinate as ℓ. These two lines split the plane into four quadrants, and thus split $\mathcal{A}$ into up to four parts. We call the part of $\mathcal{A}$ that is in the upper-left quadrant $\mathcal{A}_1$. The part of $\mathcal{A}$ that is in the upper-right quadrant, if such a part exists, we call $\mathcal{A}_2$; the part that is in the lower-right we call $\mathcal{A}_3$, and the part that is in the lower-left we call $\mathcal{A}_4$.

Now we will show how to shrink $\mathcal{A}$ and $\mathcal{B}$. We will first shrink $\mathcal{B}$ into the region $\mathcal{X}$. The part $\mathcal{A}_1$ will be shrunk into $\mathcal{V}$, $\mathcal{A}_2$ will be shrunk into $\mathcal{W}$, $\mathcal{A}_3$ will be shrunk into $\mathcal{Z}$, and $\mathcal{A}_4$ will be shrunk into $\mathcal{Y}$. In our resulting drawing, we will have $\mathcal{U}$ empty, which serves to separate the drawing from ℓ.

We can shrink each part of the drawing with appropriate uses of the scale and translate transformations. Since scale and translate are both transformations that maintain upward planarity, our mapping of each part of the drawing will be upward planar; to ensure that the entire drawing is upward planar, we only need to take care that adjacent parts will "line up". By "lining up", we mean that, for example, if we take a point p that is on the boundary between $\mathcal{A}_1$ and $\mathcal{A}_2$, after shrinking $\mathcal{A}$ into $\mathcal{R}$, p will be mapped to the same point when it is transformed as a part of $\mathcal{A}_1$ as when it is transformed as a part of $\mathcal{A}_2$. To determine the proper scaling factors, we draw a rectangle $\mathcal{S}$ around $\mathcal{A}$; the size of $\mathcal{S}$ does not matter. We split $\mathcal{S}$ into four quadrants using the same lines that we used to split $\mathcal{A}$ into four parts, and we number the quadrants similarly: $\mathcal{S}_1$ is the upper-left quadrant, $\mathcal{S}_2$ is the upper-right quadrant, and so on. Let h_i and w_i be the height and width, respectively, of $\mathcal{S}_i$.

We now specify the scaling factors for $\mathcal{B}$ and each part of $\mathcal{A}$. We must shrink $\mathcal{B}$ into $\mathcal{X}$, which is one third the height of $\mathcal{R}$, therefore the vertical scaling factor for $\mathcal{B}$ will be one third. By definition of δ_x, no point in $\mathcal{B}$ is closer that δ_x from the right edge of $\mathcal{R}$, so we do not have to scale $\mathcal{B}$ horizontally. In order for the bottom edge of $\mathcal{A}_1$ to line up with the top edge of $\mathcal{B}$, we do not want to scale $\mathcal{A}_1$ horizontally. This can be done since the left edge of $\mathcal{V}$ is to the left of any point in the drawing of G, and by construction the right edge of $\mathcal{V}$ has the same x coordinate as the right edge of $\mathcal{A}_1$. Given h_1, the height of $\mathcal{S}_1$ as defined above, the vertical scaling factor for $\mathcal{A}_1$ is $\frac{\delta_y}{3h_1}$. Finally, we set the vertical and horizontal scaling factors for $\mathcal{A}_i$ to be $\frac{\delta_y}{3h_i}$ and $\frac{\delta_x}{3h_i}$, respectively, for $i = 2, 3, 4$.

Note that $h_1 = h_2$, which means that the vertical scaling factor for $\mathcal{A}_1$ and $\mathcal{A}_2$ are the same, and hence the right edge of $\mathcal{A}_1$ will line up with the left edge of $\mathcal{A}_2$ after they are shrunk into their respective boxes. Similarly, the horizontal scaling factors for $\mathcal{A}_2$ and $\mathcal{A}_3$ are the same, and the vertical scaling factors for $\mathcal{A}_3$ and $\mathcal{A}_4$ are the same, and so these parts will line up properly after they are transformed. We do not have to worry about lining up $\mathcal{A}_4$ with $\mathcal{A}_1$, since no part of $\mathcal{A}$ crosses directly from one of these quadrants to the other: the rectangle $\mathcal{R}$ separates these parts in the original drawing.

We have now removed the crossing points q_1 and q_2, and hence have reduced the number of times that ℓ crosses the boundary of the outer face. Thus by induction, we have an upward planar drawing in which ℓ does not cross the outer face. We can then use the construction shown earlier to draw a curve from p to a point above the drawing of G. Therefore v is visible from above, as required. □

Using this characterization, we can show that we can determine if a vertex is visible from above or from below based on its edges, and that every vertex on the outer face is visible from above or from below.

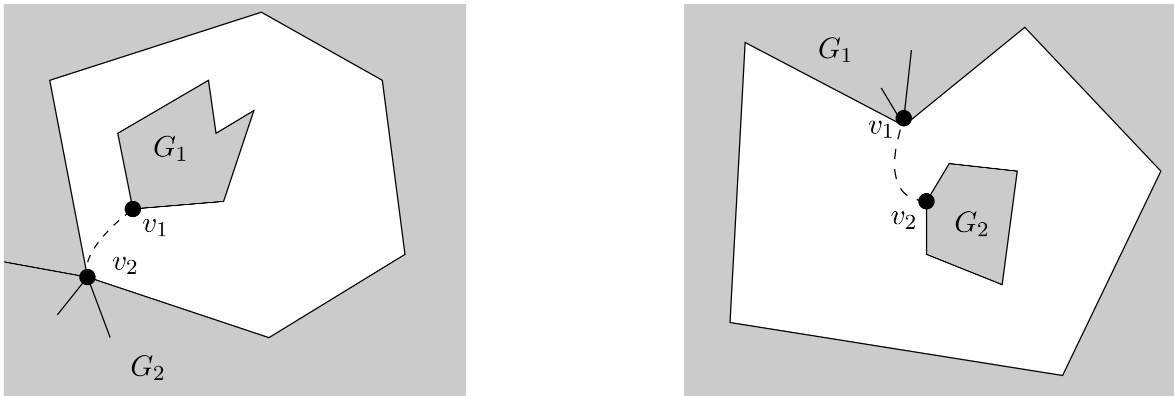
Corollary 5.1. *If the vertex v is on the outer face and has an outgoing (incoming) edge ϵ that is on the outer face, then v is visible from above (below).*

Proof. Since ϵ is on the outer face, there is a region either to the left or to the right of ϵ that is part of the outer face. We can place the point p from Lemma 5.2 in this region, and draw a straight line from p to v. Since ϵ is an outgoing edge, this region will be above v, and hence p will be above v. Thus by Lemma 5.2, v is visible from above. □

Corollary 5.2. *If v is on the outer face, it is visible from above or from below (or both).*

Proof. Given that v is on the outer face, let ϵ be an edge on the outer face incident to v. Either ϵ is an incoming edge or an outgoing edge. In the former case, v is visible from below by Corollary 5.1, and in the latter case, v is visible from above. □

When showing that G is upward planar, we will generally proceed by drawing G_1 and G_2, creating the edge $\{v_1, v_2\}$, and contracting this edge using the results from the previous section. When we draw G_1 and G_2, we will always draw G_1 completely within a face of G_2 or vice versa. We can always draw G_1 within a face F of G_2 as follows: given a drawing of G_1 and a drawing of G_2, pick a region within F and apply the scale transformation to the drawing of G_1 until it fits within this region.



a b

Figure 5.8: Drawing G when v_1 is a source.

Remark 5.1. In a planar drawing φ of G, G_1 may be drawn entirely within a single face of G_2, or G_1 may be split among several faces of G_2. However, if v_1 is not a cutvertex, G_1 must be drawn entirely within a single face of G_2. Otherwise, we take w_1 and w_2 to be vertices of G_1 that are drawn in different faces of G_2. Since v_1 is not a cutvertex, there is a path in G_1 from w_1 to w_2 that does not contain v_1. However, since this path goes from one face of G_2 to another, it must either cross some edges of G_2, or share common vertices with G_2. In the first case, this contradicts the planarity of φ, and in the latter case, this contradicts the fact that G_1 and G_2 are connected only at a single vertex.

We are now ready to give conditions under which G is upward planar, given that G_1 and G_2 are upward planar.

Lemma 5.3. *If v_1 is a source and is visible from below with embedding Γ_1 and outer face F_1, then G is upward planar.*

Proof. Given an upward planar drawing of G_2, we can draw G_1 in a face that is above v_2, as defined in Remark 2.1. Since v_1 is visible from below, we can draw a monotone curve (edge) from v_2 to v_1. We then have the condition needed to apply Theorem 4.1, so we can contract the edge (v_2, v_1), giving us an upward planar graph, G. This construction is shown in Figure 5.8a. $\square$

And by symmetry, we have the following:

Corollary 5.3. *If v_1 is a sink and is visible from above in Γ_1, then G is upward planar.*

Next, we show that if v_1 is a source but is not visible from below, we can still join obtain an upward planar graph if v_2 is visible from above.

Lemma 5.4. *If v_1 is a source and v_2 is visible from above in Γ_2, then G is upward planar.*

Proof. Since G_1 is upward planar, we have an upward planar drawing of G_1. We can then draw G_2 in the face below v_1, and add the edge (v_2, v_1). Then by Theorem 4.1, we can contract this edge and obtain an upward planar drawing of G. This construction is shown in Figure 5.8a. □

We can show that if both v_1 and v_2 are sources and one of these is on the outer face, then G is upward planar.

Lemma 5.5. *If v_1 and v_2 are sources and v_1 is on the outer face F_1, then G is upward planar.*

Proof. Consider the upward planar embedding Γ_1 of G_1. If v_1 is visible from below, then G is upward planar by Lemma 5.3. Otherwise, by Lemma 5.2, v_1 is visible from above. Then by Lemma 5.4, with the roles of G_1 and G_2 reversed, G is upward planar. □

Using this lemma, combined with Lemma 5.1, we have a necessary and sufficient condition for the upward planarity of G when both v_1 and v_2 are sources. By symmetry, we have the same condition for when v_1 and v_2 are both sinks.

Theorem 5.1. *If v_1 and v_2 are both sources (sinks), then G is upward planar if and only if at least one of G_1 or G_2 has an upward planar drawing in which v_1 or v_2 is on the outer face.*

Now we consider the case where v_1 is a source or sink, but v_2 is not.

Lemma 5.6. *If v_1 is a source, G_1 does not have an upward planar embedding in which v_1 is visible from below, v_2 is not a cutvertex, and G_2 does not have an upward planar embedding in which v_2 is visible from above, then G is not upward planar.*

Proof. We will first show that we can assume without loss of generality that v_1 is not a cutvertex of v_1, which implies, by Remark 5.1, that G_1 must be drawn in one face of G_2. We then consider the faces in which G_1 can be drawn and in each case show that either G is not upward planar, or that v_1 is visible from below.

To show that we can assume that v_1 is not a cutvertex of G_1, we show that if v_1 is a cutvertex, we can find a subgraph G_1' of G_1 that can take the place of G_1 but in which v_1 is not a cutvertex. We can then complete the proof, using G_1' in place of G_1, and if the graph G' that we obtain by joining G_1' and G_2 is not upward planar, then G cannot be upward planar since G' is a subgraph of G. Note that since G_1' is a subgraph of G_1, v_1 is still a source, and G_1' is upward planar.

To find the subgraph G_1', we consider the k components $A_1, \ldots, A_k$ of G_1 that we obtain by removing v_1. Now we create the subgraphs $H_1, \ldots, H_k$ where H_i is the subgraph of G_1 induced on $V(H_i) \cup \{v_1\}$. Intuitively, H_i is the subgraph that we obtain by "adding back" v_1 into A_i. We will show that one of these H_i's can take the place of G_1 in our proof. That is, we will show that our requirement for G_1, that it does not have an upward planar embedding in which v_1 is visible from below, holds for one of the H_i's. If not, each H_i has an embedding in which v_1 is visible from below. Then we can draw each component H_i independently, and create a new vertex v that is drawn below the drawings of each component. Since v_1 is visible from below in each component, we draw an edge from v to the v_1 of each component. We can then contract each of these edges using Theorem 4.1. This produces an upward planar drawing of G_1 in which $v = v_1$ is visible from below, contradicting our assumption that v_1 is not visible from below in G_1. Therefore one of the H_i's must not have an embedding in which v_1 is visible from below, and we take this H_i to be our G_1'. Thus, we can assume without loss of generality that v_1 is not a cutvertex of G_1.

We now show that G is not upward planar. By Remark 5.1, since neither v_1 nor v_2 is a cutvertex, G_1 is drawn entirely within a single face of G_2, and G_2 is drawn entirely within a single face of G_1. We now consider the possible faces of G_2 in which G_1 can be drawn by looking at the directions in which the faces can be, as defined in Remark 2.1. Let P be the face of G_1 in which G_2 is drawn, and let R be the face of G_2 in which G_1 is drawn. Let π_1 and π_2 be the edges on the face P that are incident to v_1, and let ρ_1 and ρ_2 be the edges on the face R incident to v_2. Since v_1 is a source, we know that π_1 and π_2 must both be outgoing edges.

First, we consider the case where R is below $v_2 = v_1$ with ρ_1 and ρ_2 both being incoming edges. But π_1 and π_2 are outgoing edges of v_1, and hence must be drawn above v_1 in an upward planar drawing, and so cannot be drawn in R.

Next, we consider the case where R is to the right or to the left of v_2 with one of ρ_1 and ρ_2 being an incoming edge and the other being an outgoing edge. Without loss of generality, we say that ρ_1 is an incoming edge and that ρ_2 is an outgoing edge. Since v_2 is not visible from above, and ρ_2 is an outgoing edge, Corollary 5.1 implies that R cannot be the outer face of G_2. Now, if we consider the drawing φ_R induced on R, R is a single cycle by definition of a face, and hence every vertex is on the outer face. Therefore, by Lemma 5.1, v_2 is visible from below in φ_R. Since v_2 is visible from below, let J be a monotone curve from v_2 to a point p that is below the drawing φ_R. We then consider the drawing $\varphi_{R \cup G_1}$. Since R is not the outer face of G_2 in φ and G_1 must be drawn inside R, G_1 will not intersect the curve J, and hence v_2 is still visible from below in $\varphi_{R \cup G_1}$ (Figure 5.9). However, $v_1 = v_2$ and hence v_1 is visible from below, contradicting our assumption that v_1 is not visible from below.

Similarly, we can show that if R is above v_2 with ρ_1 and ρ_2 both being outgoing edges, v_1 must be visible from below. Therefore in all cases where G is upward planar, if v_2 is not visible from above, then v_1 must be visible from below. $\square$

Combining the previous lemma with Lemmas 5.3 and 5.4, we have a complete characterization for the upward planarity of G when v is a source in G_1. Again, we can use symmetry to obtain a similar result for when v is a sink in G_1.

Theorem 5.2. *If v_1 is a source (sink) and v_2 is not a cutvertex, then G is upward planar if and only if G_1 has an upward planar embedding in which v_1 is visible from below (above), or G_2 has an upward planar embedding in which v_2 is visible from above (below).*

Finally, we consider the case where neither v_1 nor v_2 is a source nor a sink.

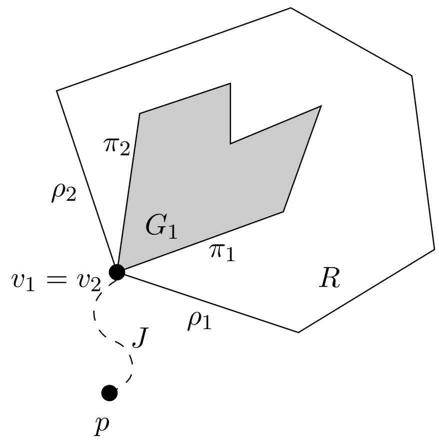


Figure 5.9: v_1 is visible from below in $\psi_{R \cup G_1}$

Lemma 5.7. *If v_1 is on the outer face F_1 and is an endpoint of an incoming edge ϵ_1 and an outgoing edge ϵ_2 such that ϵ_1 and ϵ_2 are edge-ordering neighbours around v_1 and are both on the outer face F_1 of G_1, then G is upward planar.*

Proof. We will first show that if v_2 is a source or sink, then G is upward planar, which will follow by our previous theorem. We will then show, using our edge contraction results, that if v_2 is not a source nor a sink, G is upward planar.

Since v_1 has an incoming edge on the outer face, and an outgoing edge on the outer face, v_1 is visible both from above and from below, by Corollary 5.1. Thus if v_2 is a source or sink in G_2, then by Theorem 5.2, G is upward planar. Therefore we consider the case where v_2 has indegree and outdegree greater than 0 in G_2. Without loss of generality, we can let ϵ_1 be the counterclockwise neighbour of ϵ_2 around v_1.

Now we consider two edges π_1 and π_2 incident to v_2 such that is π_1 is an incoming edge, π_2 is an outgoing edge, and π_1 is the clockwise neighbour of π_2 in the ordering around v_2. These two edges specify a face of G_2 as defined in Remark 2.2. We can then draw G_1 in an region below v_2 in this face. Since ϵ_1 is an outgoing edge from v_1 and is on the outer face of C_1, v_1 is visible from above by Lemma 5.1, and hence we can draw a monotone curve (edge) from v_1 to v_2. We now have the condition needed to apply Corollary 4.2, and hence we can contract the edge (v_1, v_2) and obtain the upward planar graph G. This construction is shown in Figure 5.10

□

Now we show that if the conditions given in the previous lemma are not satisfied, G is not upward planar. However, we must also add the condition that neither v_1 nor v_2 is a cutvertex. Figure 5.11 shows a case where the conditions of Lemma 5.7 are not satisfied, but G is upward planar.

Lemma 5.8. *Suppose that both v_1 and v_2 have indegree and outdegree greater than 0, and that neither v_1 nor v_2 is a cutvertex. If neither G_1 nor G_2 has an upward planar*

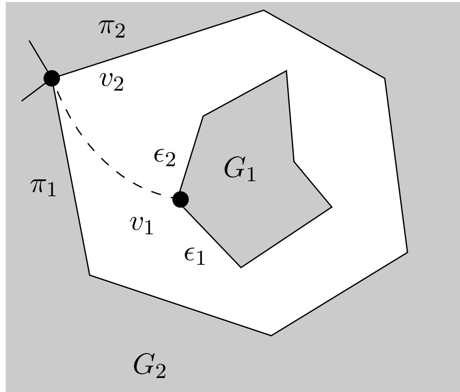


Figure 5.10: Joining G_1 and G_2 when neither v_1 nor v_2 is a source nor a sink

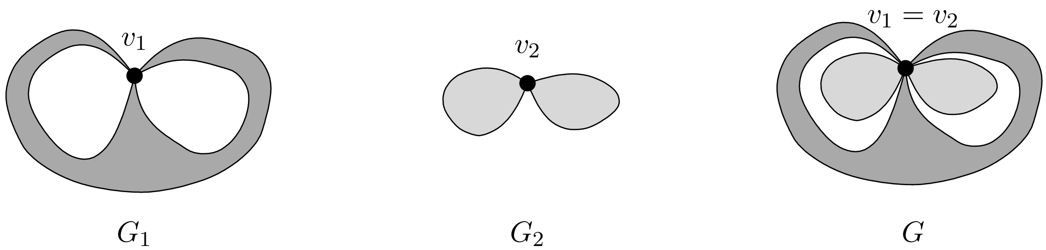


Figure 5.11: G may be upward planar if v_2 is a cutvertex

embedding in which v_1 (v_2) is on the outer face F_1 (F_2) and is an endpoint of an incoming edge ϵ_1 and an outgoing edge ϵ_2 such that ϵ_1 and ϵ_2 are edge-ordering neighbours around v_1 (v_2) and are both on the face F_1 (F_2), then G is not upward planar.

Proof. Suppose that G is upward planar, and let φ be an upward planar drawing of G. Since v_1 and v_2 are not cutvertices, G_1 is drawn entirely within a single face of G_2, and G_2 is drawn entirely within a single face of G_1. We then consider the faces of G_1 in which G_2 is drawn, and the faces of G_2 in which G_1 is drawn. Let P be the face of G_1 in which G_2 is drawn, and let R be the face of G_2 in which G_1 is drawn. Let π_1 and π_2 be the edges on the face P that are incident to v_1, and let ρ_1 and ρ_2 be the edges on the face R incident to v_2. Then we consider the orientations of π_1, π_2, ρ_1 and ρ_2.

First, we consider the case where P is a face above $v_2 = v_1$ with π_1 and π_2 both being outgoing edges of v_1. In this case, P is above $v_1 = v_2$. However, v_2 has incoming edges, which must be drawn below v_2 in an upward planar drawing. Therefore this case is not possible. By symmetry P cannot be a face below v_2 with π_1 and π_2 both being incoming edges. Therefore one of π_1 or π_2 must be an incoming edge, and the other must be an outgoing edge. Using the same reasoning, one of ρ_1 or ρ_2 must be an incoming edge, and the other must be an outgoing edge.

By Lemma 5.1, one of P or R, say P, must be the outer face of ϕ. Since P is the outer face of G, it must also be the outer face of G_1. Therefore G_1 has an upward planar embedding in which v_1 is on the outer face $P = F_1$ and is an endpoint of an incoming edge π_1 and an outgoing edge π_2 such that π_1 and π_2 are edge-ordering neighbours around v_1 and are both on the face F_1. This contradicts the assumption that F_1 does not have such an upward planar embedding, and hence G cannot be upward planar. $\square$

We can now combine the above two lemmas to obtain a necessary and sufficient condition for the upward planarity of G when neither v_1 nor v_2 are cutvertices.

Theorem 5.3. *If both v_1 and v_2 have indegree and outdegree greater than 0 and neither v_1 nor v_2 is a cutvertex, then G is upward planar if and only if G_1 or G_2 has an upward planar embedding in which v_1 (v_2) is on the outer face F_1 (F_2) and is an endpoint of an incoming edge ϵ_1 and an outgoing edge ϵ_2 such that ϵ_1 and ϵ_2 are edge-ordering neighbours around v_1 (v_2) and are both on the face F_1 (F_1), then G is not upward planar.*

The three theorems in this section completely characterize when a graph G is upward planar, given that G_1 and G_2 are both upward planar. Theorem 5.1 handles the case in which both v_1 and v_2 are sources, Theorem 5.2 handles the case in which v_1 is a source or sink but v_2 is not, and Theorem 5.3 handles the case in which neither v_1 nor v_2 is a source nor a sink. In order to determine whether or not G is upward planar, then, it suffices to test if G_1 and G_2 have embeddings that satisfy the conditions given in the theorems.

Unfortunately, Theorem 5.2 requires that v_2 is not a cutvertex, and Theorem 5.3 requires that neither v_1 nor v_2 is a cutvertex. We were unable to find a condition that would completely characterize the upward planarity of G and would allow v_1 or v_2 to be cutvertices.

Chapter 6

Biconnected graphs

In this chapter, we investigate the number of possible embeddings of a biconnected graph G. Our goal is to bound the number of possible embeddings by a function $f(k)$, where k is the number of triconnected components in the graph. By Theorem 1.1, we know that a triconnected graph has a single planar embedding; it thus seems reasonable to expect that one may be able to bound the number of possible embeddings of a biconnected graph by a function of the number of triconnected components that make up the graph.

Once we have such a bound, this will allow us to obtain a parameterized algorithm for testing the upward planarity of G. For each of these embeddings, and for each of the at most n possible outer faces, we can use the quadratic time algorithm given by Bertolazzi et al. [4] to test whether the embedding is upward planar. Thus by running a quadratic time algorithm $f(k) \cdot n$ times, we obtain a parameterized algorithm that runs in $O(f(k)n^3)$ time.

To obtain our bound on the number of possible embeddings, we will first show that there are at most eight possibilities for the embedding induced on two triconnected components that share a common vertex. We then obtain a bound on the number of possibilities for the embedding induced on ℓ triconnected components that share a common vertex. Finally, from this we can obtain a bound on the number of possible embeddings for G.

If we are given a planar embedding of a biconnected graph G, along with two triconnected components C_1 and C_2 of G that share a common vertex v, we consider the embedding of $C_1 \cup C_2$. In particular, we consider the edges incident to v. As we will show below, in Lemma 6.1, the edges of C_1 must be consecutive in the ordering around v, as must the edges of C_2. Let π_1 and π_n be the first and last edges of C_1, respectively, in the ordering around v, and let ρ_1 and ρ_m be the first and last edges of C_2, respectively, around v. As shown in Remark 2.2, the edges π_1 and π_n define a unique face P of C_1 and the edges ρ_1 and ρ_m define a unique face R of C_2. We say that C_2 is *drawn in* face P of C_1, and that C_1 is drawn in face R of C_2.

Lemma 6.1. *Given a planar embedding Γ of a graph G, and two triconnected components*

C_1 and C_2 of G that share a common vertex v, the edges of C_1 (C_2) must be consecutive around v.

Proof. Since C_1 and C_2 are triconnected components, they share at most two common vertices; if they share three or more common vertices, we now have at least three paths from any vertex in C_1 to any vertex in C_2, one for each of the common vertices, which would mean that $C_1 \cup C_2$ is triconnected.

Suppose that the edges of C_1 are not consecutive around v. Then consider two edges $\alpha = (v, a)$ and $\beta = (v, b)$ of C_1, such that there is an edge from C_2 between α and β in the clockwise ordering around v, and there is an edge from C_2 between β and α. We then consider the faces of C_2 in which α and β are drawn. Let π_1 be the first edge from C_2 after α in the clockwise ordering of edges around v, and let π_2 be the first edge from C_2 before α. Similarly, we let ρ_1 be the first edge from C_2 after β in the ordering around v, and ρ_2 be the first edge from C_2 before β. The pairs (π_1, π_2) and (ρ_1, ρ_2) each define a face of C_2 as shown in Remark 2.2. We call these faces P and R respectively.

Since C_1 is triconnected, there are at least three vertex-disjoint paths in C_1 from a to b. Since a and b are in different faces, each of these paths must cross the boundary of P, which is a face of C_2, contradicting the fact that C_1 and C_2 share at most two common vertices. $\square$

We will show that for all possible embeddings of G there are at most two possible faces of C_1 in which C_2 can be drawn, and from this we will be able to show that there are only a limited number of possible embeddings for $C_1 \cup C_2$, but we must first prove a lemma that shows that there are at most two faces that contain a given pair of vertices. These two faces will be the two faces of C_1 in which C_2 can be drawn, or the two faces of C_2 in which C_1 can be drawn.

Lemma 6.2. *Given a triconnected planar graph G and two vertices v and w, there is at most one face that contains both v and w if there is no edge (v, w) or (w, v), and at most two faces if such an edge exists.*

Proof. First, we show that if G does not have the edge (v, w) or (w, v), there is at most one face that contains both v and w. The fact that there are at most two faces if the edge (v, w) or (w, v) exists will then follow from this.

Consider a triconnected planar graph G that does not have the edge (v, w) or (w, v), and suppose that there are two or more faces that contain both v and w. Let $P = (\pi_1, \pi_2, \ldots, \pi_n)$ and $R = (\rho_1, \rho_2, \ldots, \rho_m)$ be two such faces, where the π_i's and ρ_j's are the edges that make up the faces, in clockwise order. Without loss of generality, since we are using a circular ordering, we can let v be endpoints of π_1, π_n, ρ_1, and ρ_m. Since w appears on both cycles, let w be the endpoints for π_k, π_{k+1}, ρ_ℓ, and $\rho_{\ell+1}$. (See Figure 6.1.) We will show that $\{v, w\}$ is a cutset by finding vertices a and b such that every path from a to b contains v or w.

Since we do not have the edge (v, w) or (w, v), each of the paths $(\pi_1, \ldots, \pi_k)$, $(\pi_{k+1}, \ldots, \pi_n)$, $(\rho_1, \ldots, \rho_\ell)$, and $(\rho_{\ell+1}, \ldots, \rho_m)$ has length at least two, and hence contains a vertex

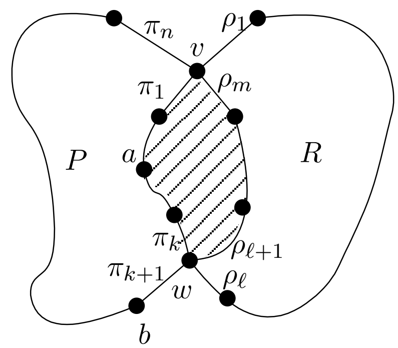


Figure 6.1: Two cycles that contain both v and w

other than v or w. Let a and b be such vertices from $(\pi_1, \ldots, \pi_k)$ and $(\pi_{k+1}, \ldots, \pi_n)$ respectively. In particular, we can let b be an endpoint of π_{k+1}.

Now, consider the cycle $\Sigma = (\pi_1, \ldots, \pi_k, \rho_{\ell+1}, \ldots, \rho_m)$. We will say that a vertex u is "on the same side" of Σ as b if there is a path from u to b that does not contain any of the vertices in Σ. This separates the vertices of G into two groups: those that are not part of the cycle and are on the same side as b, which we will call "outside" and those that are part of the cycle or are not on the same side as b, which we will call "inside". We will say that an edge is "inside" if both its endpoints are inside, and it is "outside" otherwise. Note that no outside vertex c can be drawn inside the cycle Σ. Otherwise any path P from c to b would have to cross Σ, and hence we either have an edge from P crossing an edge from Σ, contradicting the planarity of our drawing, or else P and Σ share a common vertex, contradicting the fact that c is an outside vertex.

For any two consecutive edges ϵ_1 and ϵ_2 on Σ with common vertex d, all edges between ϵ_1 and ϵ_2 in the clockwise ordering around d must be drawn inside Σ, and hence must be inside edges. By our construction of Σ, if d is not equal to v or w, then ϵ_1 and ϵ_2 are consecutive edges on the face P or R, and hence ϵ_2 is the clockwise neighbour of ϵ_1 around d. Therefore there are no edges between ϵ_2 and ϵ_1 in the clockwise ordering around d, which means that for every vertex d in the cycle Σ, d not equal to v or w, all edges incident to d are inside edges.

Note that π_{k+1} (an outside edge, since one of its endpoints is b) the counterclockwise neighbour of π_k around w. As well, all edges (w, c) between π_{k+1} and $\rho_{\ell+1}$ in the clockwise ordering around w must be inside edges: the cycle Σ divides the plane into two regions, one containing b and the other containing c, and so any path from b to c must cross Σ. Similarly, given the edges π_i and π_{i+1} for all i (similarly for ρ_i and ρ_{i+1}) in the cycle Σ, with common vertex d, all edges between π_i and π_{i+1} in the clockwise ordering around d are inside edges. Since π_i and π_{i+1} are consecutive edges in the face P, π_{i+1} is the counterclockwise neighbour of π_i around d, and hence there are no edges between π_{i+1} and π_i in the ordering on d. Therefore all edges around d are inside edges. Thus, for every vertex d in the cycle Σ, d not equal to v or w, all edges incident to it are inside edges.

Since the only vertices in the cycle Σ that can have outside edges are v or w, every path from the inside to the outside must contain either v or w. There is at least one vertex, a, on the inside, and one vertex, b, on the outside. Thus $\{v, w\}$ is a cutset of size two, contradicting the assertion that G is triconnected. Thus if G does not have the edge (v, w) or (w, v), there is at most one face that contains both v and w.

Now, suppose that G has the edge (v, w), and that there are three faces that contain both v and w. Each of these faces is a cycle in the underlying undirected graph that contains v and w, and so we can find at least one path from v to w that does not contain (v, w). The edge (v, w) can be part of at most two faces, and hence one of these faces does not contain (v, w); this face gives us two paths from v to w. Therefore we have at least four vertex-disjoint paths from v to w, and so if we remove the edge (v, w), G is still triconnected. Removal of this edge will, at worst, merge two of the faces: the third face cannot also contain (v, w) and hence will be unchanged. Thus this gives us two faces that contain v and w in a triconnected planar graph that does not contain the edge (v, w), which we know, from the first part of the proof, is not possible. □

Now we can show that there are at most two faces of C_1 in which C_2 can be drawn. By doing this, we will be able to show that we have a limited number of embeddings for $C_1 \cup C_2$.

Lemma 6.3. *Given a biconnected planar graph G and two triconnected components of G, C_1 and C_2, that share a common vertex v, there are at most two faces of C_1 in which C_2 can be drawn, and vice versa.*

Proof. In order to show that there is at most two faces of C_1 in which C_2 can be drawn, we wish to find a vertex w that must be on the same face as v in any drawing of G. Since G is biconnected, v is not a cutvertex. Thus there is a path from C_1 to C_2 that does not contain v. Let w be the last vertex on this path that is in C_1.

Since v is common to C_1 and C_2, and the path from w to C_2 does not contain any vertices from C_1, C_2 must be drawn in a face that contains both v and w. By Lemma 6.2, there are at most two faces of C_1 that contain both v and w. Thus C_2 must be drawn in one of these two faces. □

Using this fact, we show that there are at most eight possible embeddings of $C_1 \cup C_2$. These eight possibilities come from our two choices for the face of C_1 in which C_2 is drawn, our two choices for the face of C_2 in which C_1 is drawn, and our choice for the embedding of C_2 in relation to C_1.

Lemma 6.4. *Given a biconnected planar graph G, and two triconnected components C_1 and C_2 of G that share a common vertex v, there are at most eight possibilities for the embedding of $C_1 \cup C_2$ in any planar drawing of G, up to reversal of all the edge orderings.*

Proof. By Theorem 1.1, since each of C_1 and C_2 is triconnected, they each have a single planar embedding, up to reversal of all the edge orderings. Thus the only area that can give us a choice in the embedding is in how we combine the embeddings for C_1 and C_2. Let Γ_1 be the embedding of C_1, and Γ_2 be the embedding of C_2.

By Lemma 6.3, there are at most two possible faces of C_1 in which C_2 can be drawn, and vice versa, so this gives us four possibilities.

We now have the possibility of reversing the edge orderings in the embedding Γ_1, in the embedding Γ_2, both, or neither. However, note that if we reverse the edge orderings in Γ_1, we obtain the same embedding as if we reverse the edge ordering in the embedding of Γ_2, up to reversal of the edge ordering in the combined embedding: given a vertex w in C_1 with clockwise ordering $(\delta_1, \ldots, \delta_k)$, if we reverse the edge ordering in Γ_2, this does not affect the ordering around w. If we reverse the edge ordering in Γ_1, the resulting ordering around w is $(\delta_k, \ldots, \delta_1)$. If w is in C_2, then reversing the ordering in Γ_2 yields the clockwise ordering $(\delta_k, \ldots, \delta_1)$ while reversing the ordering in Γ_1 does not change the ordering. If w is shared by C_1 and C_2, then let the first ℓ edges $(\delta_1, \ldots, \delta_\ell)$ be edges from C_1, and the rest be edges from C_2. Then if we reverse the ordering in Γ_2, we get the ordering $(\delta_1, \ldots, \delta_\ell, \delta_k, \ldots, \delta_{\ell+1})$, and if we reverse the edge ordering in Γ_1, we get the ordering $(\delta_\ell, \ldots, \delta_1, \delta_{\ell+1}, \ldots, \delta_k) = (\delta_{\ell+1}, \ldots, \delta_k, \delta_\ell, \ldots, \delta_1)$. Thus in all cases, the edge ordering that we obtain by reversing Γ_2 is the reverse of the edge ordering that we obtain by reversing Γ_1.

Similarly, if we reverse the edge orderings in both Γ_1 and Γ_2, we obtain the same embedding as if we do not reverse the edge ordering, up to reversal of the edge orderings in the combined embedding. This means that we only need to consider two choices: not reversing either of the embeddings, or just reversing Γ_2. For simplicity, we will assume without loss of generality that in our drawing, we do not reverse either of the embeddings. We do not lose generality since the embedding for C_2 that we start off with could have been the reversal of Γ_2.

For any vertex w that is shared between C_1 and C_2, we can show that the above choices determine a unique ordering of the edges around w in $\Gamma_{C_1 \cup C_2}$ By Lemma 6.1, since C_1 and C_2 are triconnected, the edges of C_1 must be consecutive in the ordering around w in $\Gamma_{C_1 \cup C_2}$ and the edges of C_2 must be consecutive. Let $P = (\pi_1, \ldots, \pi_n)$ be the face of C_1 in which we draw C_2, and $R = (\rho_1, \ldots, \rho_m)$ be the face of C_2 in which we draw C_1. Without loss of generality, we can let w be an endpoint of π_1, π_n, ρ_1, and ρ_m. Let $D = (\delta_1, \ldots, \delta_p)$ and $E = (\epsilon_1, \ldots, \epsilon_q)$ be the edges around w in clockwise order in C_1 and C_2 respectively. Since these are in circular order, we can let δ_1 and δ_p be edges on the face P, and ϵ_1 and ϵ_q be edges on the face R. Since the edges of C_1 must be consecutive, and we are already given the clockwise ordering of the edges of C_1 and the edges of C_2, the ordering of the edges around w must be $(\pi_1, \ldots, \pi_p, \rho_1, \ldots, \rho_q)$.

Therefore we have at most eight possibilities for $\Gamma_{C_1 \cup C_2}$: two possibilities for the face of C_1 in which C_2 is drawn, two possibilities for the face of C_2 in which C_1 is drawn, and the possibility reversing or not reversing the edge orderings of C_2. $\square$

Using the above lemma, we can see that if a biconnected graph G has k triconnected components, and any vertex is shared by at most two components, G has at most 8^{k-1} possible embeddings, since each component adds eight times more possible embeddings. We now look at the case where a vertex may be shared by more than two components.

Lemma 6.5. *Given a planar biconnected graph G and k triconnected components, $C_1, \ldots,$ C_k, that share a vertex v, $C_1 \cup \cdots \cup C_k$ has at most $(k-1)!8^{k-1}$ possible embeddings.*

Proof. If $k = 1$, we have at most one possible embedding. Now, suppose that we have k triconnected components, and we are given the embedding $\Gamma_{C_1 \cup \cdots \cup C_k}$. In this embedding, we will denote by $F_{i,j}$ the face of C_i in which C_j is drawn.

If we want to add a new triconnected component C_{k+1} to v, then we must determine the position of its edges in the ordering around v. By Lemma 6.4, we add eight embeddings by choosing a face of C_1 in which C_{k+1} is drawn, choosing a face of C_{k+1} in which C_1 is drawn, and choosing an embedding for C_{k+1}. Now we consider the components that are drawn in the same face of C_1 as C_{k+1}. Without loss of generality, we can assume that these are the components C_2 up to C_ℓ. Thus we need to determine the ordering of the components $C_2, \ldots, C_\ell, C_{k+1}$ around v. Without loss of generality, we can assume that the edges of C_2 up to C_ℓ are ordered such that the edges of C_2 appear first, followed by the edges of C_3, and so on until we reach the edges of C_ℓ. Then we can add the edges of C_{k+1} between any pair of components, or before C_2 or after C_ℓ. This gives us $\ell \leq k$ times more possibilities.

Thus when we add component C_{k+1}, we add at most $8k$ times more possibilities, and so by induction, if we have k components, there are at most $(k-1)!8^{k-1}$ possible embeddings. □

Using this lemma, we can obtain a bound on the number of possible embeddings for a biconnected graph that has k triconnected components.

Theorem 6.1. *Given a planar biconnected graph G that has k triconnected components, G has at most $k!8^{k-1}$ possible planar embeddings, up to reversal of all the edge orderings.*

Proof. For each vertex v_i that is shared between two or more triconnected components, the k_i triconnected components that contain v_i together contribute at most $(k_i-1)!8^{k_i-1}$ embeddings, thus the maximum number of possible embeddings is $\sum_i (k_i-1)!8^{k_i-1}$. Since $k_i \leq k$ for all i, we have $\sum_i (k_i-1)!8^{k_i-1} \leq \sum_i (k-1)!8^{k-1}$. There are at most k such vertices v_i, and hence there are at most k terms in the summation: $\sum_i (k-1)!8^{k-1} \leq k(k-1)!8^{k-1} = k!8^{k-1}$. Therefore the maximum number of possible embeddings is $k!8^{k-1}$. □

Using this bound, we can then obtain a parameterized algorithm that tests whether G is upward planar.

Theorem 6.2. *There is an $O(k!8^k n^3)$-time algorithm to test whether a biconnected graph is upward planar, where n is the number of vertices, and k is the number of triconnected components.*

Proof. Our algorithm works as follows: first it divides the input graph G into triconnected components. Then, for each possible embedding of G, we test whether or not it is upward planar.

To divide the graph into triconnected components, we can use the quadratic time algorithm given by Hopcroft and Tarjan [20].

Now we test each possible embedding to see if it is upward planar. By Theorem 6.1, we have $k!8^{k-1}$ embeddings. Euler's formula relates the number of vertices, edges, and faces in a planar graph by the equation $n - m + f = 2$, where n is the number of vertices, m is the number of edges, and f is the number of faces. Therefore, for each embedding, there are $O(n)$ possibilities for the outer face. From Bertolazzi et al. [4], we have an algorithm that takes $O(n^2)$ time to determine whether a given embedding and outer face correspond to an upward planar drawing. Thus we can run the algorithm for each possible embedding and outer face, giving a time complexity of $O(k!8^k n^3)$. $\square$

We now have a fixed-parameter algorithm for determining whether a biconnected graph is upward planar, with the parameter being the number of triconnected components. Note, however, that our bound on the number of possible embeddings is, in many cases, much larger than the actual number of possible embeddings, as can be seen in the proof of Theorem 6.1. For example, if every vertex is common to at most two triconnected components, we have only 8^{k-1} possible embeddings, rather than $k!8^{k-1}$.

Notice that the proof of Theorem 6.1 does not depend on upward planarity. Therefore it could be applied to other graph drawing problems in which we have an algorithm that solves the problem given a specific embedding.

Chapter 7

Conclusions and future work

In this thesis, we first investigated contracting an edge in an upward planar graph. Using these results, we then showed conditions under which joining two upward planar graphs at a single vertex yields a new graph that is upward planar. And finally, we showed that for a biconnected graph, there is a parameterized algorithm for testing its upward planarity where the parameter is the number of triconnected components. Thus we were able to show a successful application of parameterized complexity technique to graph drawing. In more detail:

In Chapter 4, we considered the effect of contracting an edge in an upward planar graph, and completely characterized the conditions under which contracting an edge results in an embedding that is still upward planar. Given an upward planar embedding Γ of a graph G and an edge ϵ of G that we wish to contract, we showed that we can determine whether or not Γ/ϵ is an upward planar embedding based on the orientations of the clockwise and counterclockwise neighbours of ϵ. We gave conditions under which Γ/ϵ is always an upward planar embedding, under which Γ/ϵ is never an upward planar embedding, and under which Γ/ϵ is an upward planar embedding if and only if the graph produced by reversing the orientation of ϵ is upward planar. These conditions are summarized in Table 4.1.

In Chapter 5, we then investigated the conditions under which joining two upward planar graphs G_1 and G_2 produces an upward planar graph G. The way in which we joined G_1 and G_2 was by taking a vertex v_1 from G_1 and a vertex v_2 from G_2 and identifying them. To prove the conditions under which G was upward planar, we showed that we could draw each graph separately, draw an edge from v_1 to v_2, and contract the edge. Using our edge contraction results, we were able to then conclude that the resulting graph was upward planar. The results from Chapters 4 and 5 are extensions of results given by Hutton and Lubiw [22].

Finally, in Chapter 6, we showed that a biconnected graph has at most $k!8^{k-1}$ possible embeddings, where k is the number of triconnected components. From this result, we then obtained a parameterized algorithm, where the parameter is the number of triconnected components, by testing all possible combinations of embeddings Γ and outer faces F to see if there is an upward planar drawing whose embedding is Γ and whose outer face is F.

Our parameterized algorithm only works on biconnected graphs; an obvious extension of our work is to find a parameterized algorithm that can handle general graphs. It may be possible to combine the graph joining results with our biconnected components algorithm to achieve this: our graph joining results give conditions under which two upward planar graphs with given embeddings can be joined together to obtain a new upward planar graph. Since our biconnected components algorithm runs through all possible embeddings for a biconnected graph, we can test whether there is an embedding that, in addition to being upward planar, satisfies the conditions required for joining it to another biconnected component. However, we have two difficulties that we will have to overcome. The first is that the condition given in Theorem 5.2 depends on determining if certain vertices are visible from above or below; we do not yet have an algorithm that can determine whether a given embedding has a drawing in which a given vertex is visible from above or below. The second difficulty is that Theorems 5.2 and 5.3 require that the vertex at which we joint the two subgraphs is not a cutvertex in either subgraph; if we have a vertex that is common to three different biconnected components, we cannot apply these theorems.

Since many interesting graphs are not upward planar, it would be natural to try to determine how close one can get to obtaining an upward planar drawing, and turn upward planarity testing into a maximization problem. Since upward planarity testing is NP-complete, it is unlikely that there is an efficient solution to the maximization problem, but this introduces more possible research directions. One possibility is in approximation algorithms: is it possible to obtain a drawing in which the number of upward edges is at most a fixed number away from the optimal, or at least a fixed percentage of the optimal?

Another possibility is obtaining a parameterized algorithm that determines whether or not a graph has a drawing in which at most $\frac{1}{k}$ of the edges point downward. For $k = \infty$, this is upward planarity testing, which is NP-complete. For $k = 2$, this is trivial: take any drawing of the graph in which no edge is drawn horizontally. Either this drawing or $\mathsf{flip\_x}$ of the drawing has at least half the edges pointing upward. Thus it is possible that between these two extremes, we may be able to obtain a parameterized result.

We can also consider layered upward planar drawings. As discussed in Section 1.1.1, Jünger et al. [23] show that testing whether a graph has a proper layered upward planar drawing can be done in linear time. It may be possible to obtain a parameterized result for testing whether a graph has an layered upward planar drawing in which each edge can be draw between two vertices that are at most k layers apart. This may be a desirable criterion for drawings, since we often want to avoid drawings that have very long edges; restricting the number of layers between the endpoints of the edges restricts the lengths of the edges.

Parameterized complexity is a fairly new area and seeks to find efficient solutions to hard problems. Many problems in graph drawing have been shown to be NP-complete, and so parameterized complexity may be able to offer solutions to many of these problems. Some possible parameters that we can look at are the height, width, or area of the drawing, the maximum indegree or outdegree in a graph, the number of faces in a planar graph, or the number of sources or sinks.

Bibliography

[1] R. Balasubramanian, Michael R. Fellows, and Venkatesh Raman. An improved fixed parameter algorithm for vertex cover. *Information Processing Letters*, 65(3):163–168, 1998.

[2] Paola Bertolazzi and Giuseppe Di Battista. On upward drawing testing of triconnected digraphs (extended abstract). In *Proceedings of 7th ACM Symposium on Computational Geometry*, pages 272–280. ACM Press, 1991.

[3] Paola Bertolazzi, Giuseppe Di Battista, and Walter Didimo. Quasi-upward planarity. *Algorithmica*, 32:474–506, 2002.

[4] Paola Bertolazzi, Giuseppe Di Battista, Giuseppe Liotta, and Carlo Mannino. Upward drawings of triconnected digraphs. *Algorithmica*, 12:476–497, 1994.

[5] Paola Bertolazzi, Giuseppe Di Battista, Carlo Mannino, and Roberto Tamassia. Optimal upward planarity testing of single-source digraphs. *SIAM Journal on Computing*, 27(1):132–196, 1998.

[6] John Adrian Bondy and U. S. R. Murty. *Graph Theory with Applications*. North Holland, New York, 1976.

[7] Gilles Brassard and Paul Bratley. *Fundamentals of Algorithmics*. Prentice-Hall, New Jersey, 1996.

[8] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. McGraw-Hill Book Company, second edition, 2001.

[9] Giuseppe Di Battista, Peter Eades, Roberto Tamassia, and Ioannis G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice-Hall, New Jersey, 1999.

[10] Giuseppe Di Battista and Giuseppe Liotta. Upward planarity checking: "Faces are more than polygons" (extended abstract). In Sue H. Whitesides, editor, *Proceedings of Graph Drawing '98*, volume 1547 of *Lecture Notes in Computer Science*, pages 72–86. Springer-Verlag, 1998.

[11] Giuseppe Di Battista, Wei-Ping Liu, and Ivan Rival. Bipartite graphs, upward drawings, and planarity. *Information Processing Letters*, 36:317–322, 1990.

[12] Giuseppe Di Battista and Roberto Tamassia. Algorithms for plane representations of acyclic digraphs. *Theoretical Computer Science*, 61:175–198, 1988.

[13] Giuseppe Di Battista, Roberto Tamassia, and Ioannis G. Tollis. Area requirement and symmetry display of planar upward drawings. *Discrete and Computational Geometry*, 7:381–401, 1992.

[14] Reinhard Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer, New York, 2nd edition, 2000.

[15] Rod G. Downey and Michael R. Fellows. *Parameterized Complexity*. Monographs in Computer Science. Springer, New York, 1997.

[16] Vida Dujmović, Michael R. Fellows, Michael T. Hallett, Matthew Kitching, Giuseppe Liotta, Catherine McCartin, Naomi Nishimura, Prabhakar Ragde, Frances Rosamond, Matthew Suderman, Sue Whitesides, and David R. Wood. A fixed-parameter approach to two-layer planarization. In *Proceedings of the 9th International Symposium on Graph Drawing*, volume 2265 of *Lecture Notes in Computer Science*, pages 1–15, 2001.

[17] Vida Dujmović, Michael R. Fellows, Michael T. Hallett, Matthew Kitching, Giuseppe Liotta, Catherine McCartin, Naomi Nishimura, Prabhakar Ragde, Frances Rosamond, Matthew Suderman, Sue Whitesides, and David R. Wood. On the parameterized complexity of layered graph drawing. In *Proceedings, 9th Annual European Symposium on Algorithms*, pages 488–499, 2001.

[18] I. Fáry. On straight lines representation of planar graphs. *Acta scientiarum mathematicarum*, 11:229–233, 1948.

[19] Ashim Garg and Roberto Tamassia. On the computational complexity of upward and rectilinear planarity testing. *SIAM Journal on Computing*, 31(2):601–625, 2001.

[20] John Hopcroft and Robert E. Tarjan. Dividing a graph into triconnected compononts. *SIAM Journal on Computing*, 2:136–158, 1972.

[21] John Hopcroft and Robert E. Tarjan. Efficient planarity testing. *Journal of the ACM*, 21(4):549–568, 1974.

[22] Michael D. Hutton and Anna Lubiw. Upward planar drawing of single source acyclic digraphs. *SIAM Journal on Computing*, 25(2):291–311, 1996.

[23] Michael Jünger, Sebastian Leipert, and Petra Mutzel. Level planarity testing in linear time. In S. Whitesides, editor, *Proc. Graph Drawing: 6th International Symposium (GD'98)*, volume 1547 of *Lecture Notes in Computer Science*, pages 224–237. Springer, 1998.

[24] Michael Kaufmann and Dorothea Wagner, editors. *Drawing Graphs: Methods and Models*, volume 2025 of *Lecture Notes in Computer Science Tutorial*. Springer, 2001.

[25] Achilleas Papakostas. Upward planarity testing of outerplanar dags (extended abstract). In Roberto Tamassia and Ioannis G. Tollis, editors, *Proceedings of Graph Drawing '94*, volume 894 of *Lecture Notes in Computer Science*, pages 298–306. Springer, 1995.

[26] C. R. Platt. Planar lattices and planar graphs. *Journal of Combinatorial Theory (B)*, 21:30–39, 1976.

[27] H. L. Royden. *Real Analysis*. Prentice-Hall, New Jersey, third edition, 1988.

[28] S. K. Stein. Convex maps. *Proceedings of the American Mathematical Society*, 1951:464–466, 1951.

[29] Roberto Tamassia and Ioannis G. Tollis. A unified approach to visibility representations of planar graphs. *Discrete and Computational Geometry*, 1(4):321–341, 1986.

[30] Carsten Thomassen. Planar acyclic oriented graphs. *Order*, 5:349–361, 1989.

[31] K. Wagner. Bemerkungen zum vierfarbenproblem. *Jahresbericht der Deutschen Mathematiker-Vereinigung*, 46:26–32, 1936.

[32] Hassler Whitney. A set of topological invariants for graphs. *American Journal of Mathematics*, 55:231–235, 1933.

[33] Peng Zhou. Drawing graphs of bounded treewidth/pathwidth. Master's thesis, University of Auckland, 2001.