8-2014

Survey of Graph Embeddings into Compact Surfaces

Sophia N. Potoczak

A SURVEY OF GRAPH EMBEDDINGS INTO COMPACT SURFACES

By

Sophia N. Potoczak

Bachelor of Arts, Albion College, 2012

A THESIS

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Arts

(in Mathematics)

The Graduate School

The University of Maine

August 2014

Advisory Committee:

Robert Franzosa, Professor of Mathematics, Advisor

Andrew Knightly, Associate Professor of Mathematics

Benjamin Weiss, Assistant Professor of Mathematics

THESISACCEPTANCE STATEMENT

On behalf of the Graduate Committee for Sophia N. Potoczak, I affirm that this manuscript is the final and accepted thesis. Signatures of all committee members are on file with the Graduate School at the University of Maine, 42 Stodder Hall, Orono, Maine.

Robert Franzosa, Professor of Mathematics (Date)

LIBRARY RIGHTS STATEMENT

In presenting this thesis in partial fulfillment of the requirements for an advanced degree at The University of Maine, I agree that the Library shall make it freely available for inspection. I further agree that permission for "fair use" copying of this thesis for scholarly purposes may be granted by the Librarian. It is understood that any copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Sophia N. Potoczak (Date)

A SURVEY OF GRAPH EMBEDDINGS INTO COMPACT SURFACES

By Sophia N. Potoczak

Thesis Advisor: Dr. Robert Franzosa

An Abstract of the Thesis Presented
in Partial Fulfillment of the Requirements for the
Degree of Master of Arts
(in Mathematics)
August 2014

A prominent question of topological graph theory is "what type of surface can a nonplanar graph be embedded into?" This thesis has two main goals. First to provide a necessary background in topology and graph theory to understand the development of an embedding algorithm. The main purpose is developing and proving a direct constructive embedding algorithm that takes as input the graph with a particular order of edges about each vertex. The embedding algorithm will not only determine which compact surface the graph can be embedded into, but also determines the particular embedding of the graph on the surface. The embedding algorithm is then used to investigate surfaces into which trees and a class of the complete bipartite graphs can be embedded. Further, the embedding algorithm is used to investigate non-surface separating graph embeddings.

DEDICATION

This thesis is dedicated to my mother, Julieann, and my father, Dennis.

ACKNOWLEDGEMENTS

I would like to thank the Mathematics and Statistics Department at the University of Maine for the opportunity to develop as a mathematician and teacher. Most importantly, I would like to thank my advisor, Bob Franzosa, for his academic support and encouragement throughout my time in the master's program. I must thank my committee Andy Knightly and Ben Weiss for their thoughtful comments. Lastly, I offer thanks all of the professors in the department as each individual inspired me to enjoy mathematics from the perspective of many different fields.

Finally, I thank my family and friends for their support through my academic endeavors.

Contents

List of Tables

List of Figures

Chapter 1
INTRODUCTION

The study of topology and graph theory began in the eighteenth century with Leonhard Euler's solution to the famous Konigsberg Bridges Problem. In the city of Konigsberg, Prussia, the Pregel River flowed such that the river separated the city into four different regions. There were seven bridges that crossed the river and connected the regions. Many people in the city of Konigsberg considered the problem of whether one could walk through the city and cross each bridge exactly once. In 1735, Euler proved that it was impossible to do such a walk through the city [1]. The Konigsberg Bridge problem relates to graph theory because the bridges and pathways in the city represent the edges of a graph and the intersections of these pathways represent the vertices of the graph. Today Euler's solution is characterized as a particular type of graph that contains a "walk" that only uses each edge exactly once. Throughout the following centuries, after Euler's solution in 1735, many mathematicians made major advances in the field of topological graph theory.

Beginning in the nineteenth century, mathematicians began using graphs to design and analyze electrical circuits [1]. A circuit can be modeled as a graph by using the components of the circuit as vertices and the wires connecting the components as the edges. In circuit design, it is of concern to have a minimum number of crossings between the wires. One can use the theory developed in topological graph theory to determine if it is possible to place a circuit on a planar circuit board such that the wires do not cross. This question is directly related to a fundamental problem in graph theory. Given a graph, how do we determine if we can draw the graph on the plane without having any edges cross? If we cannot draw a graph on the plane without crossings, then what is the minimal number of crossings we can have in

such a drawing? If one cannot draw a graph on a planar surface without any edges crossing, then what type of surface could one draw the graph such that the edges do not cross?

In the 1940s, while Paul Turán, a Hungarian mathematician, worked at a labor camp loading bricks from a kiln on a rail cart to transport the bricks to a storage area, the inefficiency of the transport method led him to the second question. In particular, there were a number of kilns where the bricks were made and a number storage yards where the bricks were stored. Each kiln was connected by rail to every storage yard and each storage yard was connected to each kiln. Turán and other labor camp workers were responsible for loading bricks onto a cart and pushing the cart along the rails to a designated storage area. However, an issue arose at the crossing of the railroads; frequently the carts would tip over at the crossings and cause loss of time and productivity. Turán thought, to optimize the efficiency of the brick transport method, the number of rail crossings ought to be minimized. Turán quickly realized this was a relatively simple task for the fixed number of kilns and storage areas at the brick factory; but given m kilns and n storage areas the difficulty of the problem increased immensely [13]. Ultimately, Turán posed the still unsolved problem of the minimum number of crossings in what is known as the complete bipartite graph on m and n vertices.

In the following chapters, the last question of the type of surface on which a graph can be drawn will be addressed by developing a constructive algorithm that embeds graphs into compact surfaces. For G a graph, we can choose a particular ordering of edges about each vertex, this is called a rotation system. The rotation system of the graph acts as the input for the embedding algorithm. The algorithm will construct an embedding by first forming the components of the surface such that the boundary of these components is given by edges in the graph. We can then

glue together these components with the edges, as prescribed, to create the surface with a particular drawing of the graph such that the edges do not cross.

In the first chapter, a brief introduction with necessary background and terminology of topology, graphs, and compact surfaces will be presented. Next, the Topological Graph chapter will provide a detailed introduction to graph theory and graph embeddings. The Constructing and Classifying Compact Surfaces chapter continues with the brief introduction to constructing compact surfaces and discusses a proof of the Compact Surface Classification Theorem. In the Embedding Algorithm Chapter, a constructive algorithm is developed that will determine an embedding of a graph into a particular compact surface. Finally, in the Consequences chapter some results regarding embedding of trees and the complete bipartite graphs $K_{3,n}$ are proved using the embedding algorithm. The texts *Graphs on Surfaces* by Mohar and Thomassen [10] and *Topological Graph Theory* by Gross and Tucker [7] provide further reading on some of the materials presented in this thesis.

1.1 Introduction to Topology

A *topological space* X is a set with a collection of open subsets that establishes a notion of proximity on X. The collection of open sets is called a *topology* and satisfies the following properties

(i) $\varnothing$ and X are open sets;

(ii) the intersections of finitely many open sets is an open set;

(iii) the union of any collection of open sets is an open set.

A topological space can have a smaller collection of open sets, called a basis, that determines the topology. A *basis* is any collection of open sets, called *basis elements*, with the property that all open sets in the topology can be expressed as a union of sets in the basis. For instance, the plane, $\mathbb{R}^2$, is a topological space with a basis of

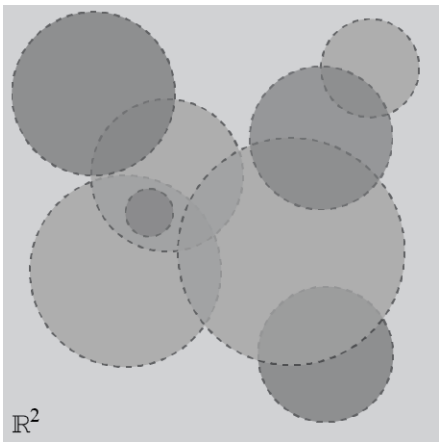

Figure 1.1. The plane $\mathbb{R}^2$ with basis elements that are open balls.

all open balls. In Figure 1.1, a collection of some basis elements are shown.

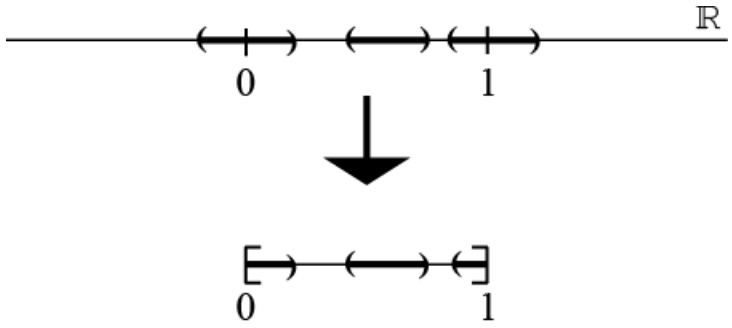

Figure 1.2. An open set in the subspace topology on $A = [0, 1]$ inherited from $\mathbb{R}$.

Given a topological space X, we can create other topological spaces that will inherit certain properties from the underlying topology. In particular, a *subspace* of X is a subset $A \subset X$ with a topology on A. The topology is given by defining open sets of A to be the intersection of A with the open sets of X. For example, consider the real line $\mathbb{R}$ where open sets are defined to be unions of open intervals. Let $A = [0, 1] \subset \mathbb{R}$ be a subspace of the real line. To determine the open sets of A, we must intersect A with sets of the form $I = (a, b)$, where $a < b \in \mathbb{R}$. Sets of the form (c, d) where $0 < c < d < 1$ are open in $A = [0, 1]$ and sets of the form $[0, c)$ and

$(c, 1]$ where $0 < c < 1$ are open in A (see Figure 1.2). All open sets in $[0, 1]$ then are unions of open sets of these three types.

An important subspace of $\mathbb{R}^2$ that will play a role in the following chapters is called the open disk. The open disk, denoted D, is defined to be the points in the plane distance less than one away from the origin (see Figure 1.3).

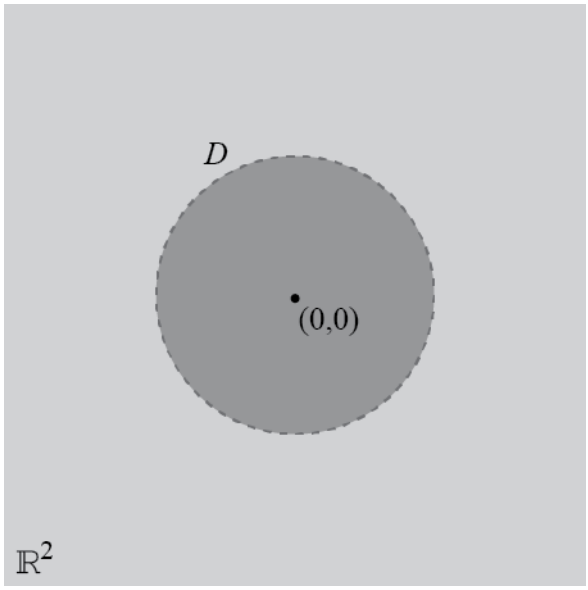

Figure 1.3. The open disk in $\mathbb{R}^3$.

Let X and Y be topological spaces. We can take the product of the two spaces and denote this set $X \times Y$. We call $X \times Y$ the *product space* of X and Y. Because X and Y each have associated topologies, the set $X \times Y$ will inherit a topology called the product topology from this construction. A basis element for $X \times Y$ is of the form $U \times V$ where U is open in X and V is open in Y (see Figure 1.4). The plane can be thought of as a topological space formed under the product topology, $\mathbb{R} \times \mathbb{R}$ where the open sets in $\mathbb{R}$ are unions of open intervals. Thus the basis elements for the product topology on $\mathbb{R} \times \mathbb{R}$ are given by open rectangles formed by taking the product of two open intervals in $\mathbb{R}$.

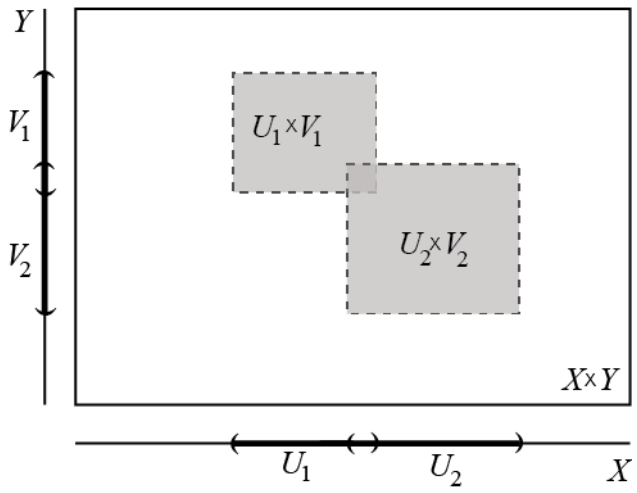

Figure 1.4. Examples of basis elements for the product topology on the set $X \times Y$.

Open sets of topological spaces help define many important topological properties. We define a *closed subset of* X to be a set whose complement in X is open. A topological space X is *compact* if for every open cover of X there exists a finite open subcover. When a topological space is a subset of Euclidean space compactness can be characterized as being closed and bounded. A topological space X is called *connected* if there does not exist a pair of disjoint nonempty open sets whose union is X. There are many ways a topological space can be broken down into its fundamental pieces. One way that we have already seen is a basis set of a topological space. Another way we can look at a topological space is by defining components, where a *component* is a maximal connected subset of the topological space. In topology the study of open sets allows us to formulate alternate definitions of fundamental concepts in analysis. One such example is a continuous function. In particular, let f be a function mapping a topological space X to a topological space Y. Then f is said to be a *continuous function* if the preimage of every open set is open. Informally, a continuous function sends points that are close in one space to points that are close in the other; this allows for studying topological spaces with continuous

functions to understand the structure and properties of the spaces. For instance, continuous functions preserve certain topological properties, like connectedness and compactness. If X is a connected topological space and $f : X \to Y$ is a continuous function, then the image of X under f is a connected subset of Y. For another example, the fundamental equivalence relation in topology, homeomorphism, is defined using continuous functions. In detail, let X and Y be topological spaces. Define a *homeomorphism* h to be a continuous bijection $h : X \to Y$ with continuous inverse $h^{-1} : Y \to X$.

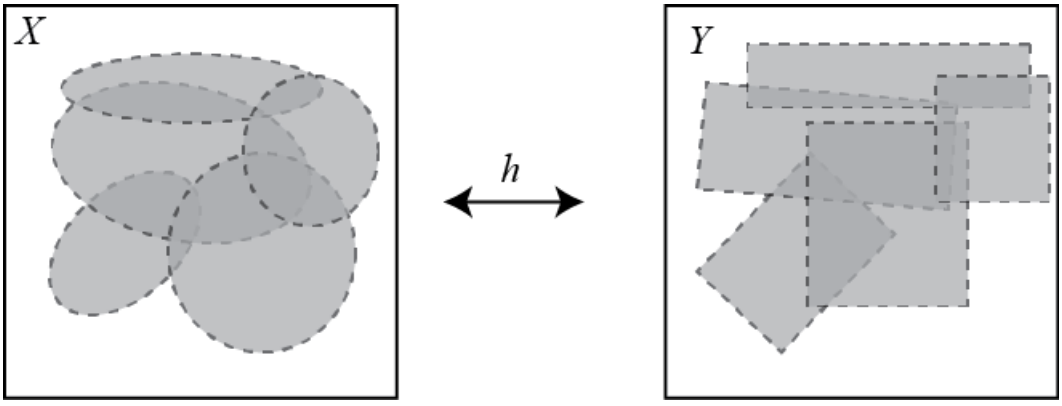

Figure 1.5. Under a homeomorphism open sets of X are mapped to open sets of Y

Two topological spaces are said to be *homeomorphic* or *topologically equivalent* if there exists a homeomorphism between the spaces (Figure 1.5). Continuous functions can also be used to study how one topological space sits within another topological space. This is accomplished via a function called an *embedding*. Such a function $f : X \to Y$ will homeomorphically map X to the subspace $f(X)$ in Y but f is not necessarily onto Y. Continuous functions can also be used to construct a new topological space from some topological space X. To see this, we will first define a partition on the topological space X. Let X' be the set of equivalence classes given by the partition on X, then define a function $q : X \to X'$ such that q identifies collections of points in X to a new point in the constructed topological space X'.

The function q essentially partitions the space X into equivalence classes such that an equivalence class is mapped to a single point in the new space X'. A topology, called the *quotient topology*, is defined on X' such that U is open in X' if and only if $q^{-1}(U)$ is open in X. A topological space constructed using the quotient topology is called a *quotient space*.

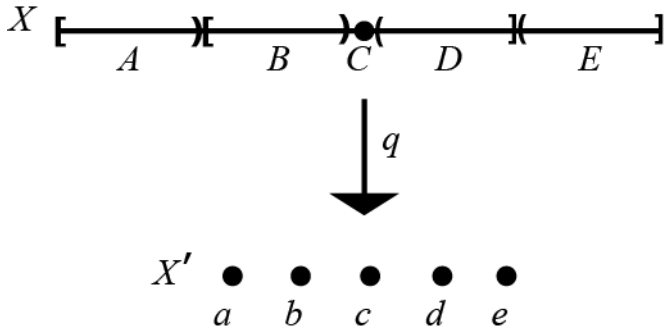

Figure 1.6. Each subset of X is collapsed to a single point in the quotient space X'.

For example, let X be an arbitrary topological space and let A, B, C, D, E be subsets of X as pictured in Figure 1.6. Define a function $q : X \to X'$ such that q maps each subset of X to a single poin;, for instance $q(A) = a$ where $a \in X'$. Therefore the quotient space X' is defined as the image of X under the function q where a set is open in X' if and only if its preimage under q is open in X. Now with a basic understanding of the importance of open sets and continuous functions in topology, we can delve into the study of topological graph theory.

1.2 Graph Theory

In graph theory, a graph G is considered to be a combinatorial object, meaning $G = \{V, E\}$ where V is the set of vertices and E is the set of unordered edges such that E consists of two-element subsets of V. [2]. When the edge set E and vertex set V of a graph are finite, then the graph is considered finite. In the combinatorial

definition of a graph one is not concerned with how to represent these edges and vertices; but more so, that given $i, j \in V$ such that $\{i, j\} \in E$ then the vertices are considered to be connected by an edge. In this definition an edge does not have any associated properties besides connecting vertices. In topology, a topological graph is considered to be a set of vertices with a set of edges; however, the topological graph is considered to be a topological space with specific properties due to its construction. In the following chapters, whenever the term graph is used, one can assume it is referring to the topological graph rather than the combinatorial graph. Additionally, we can assume each graph considered is finite.

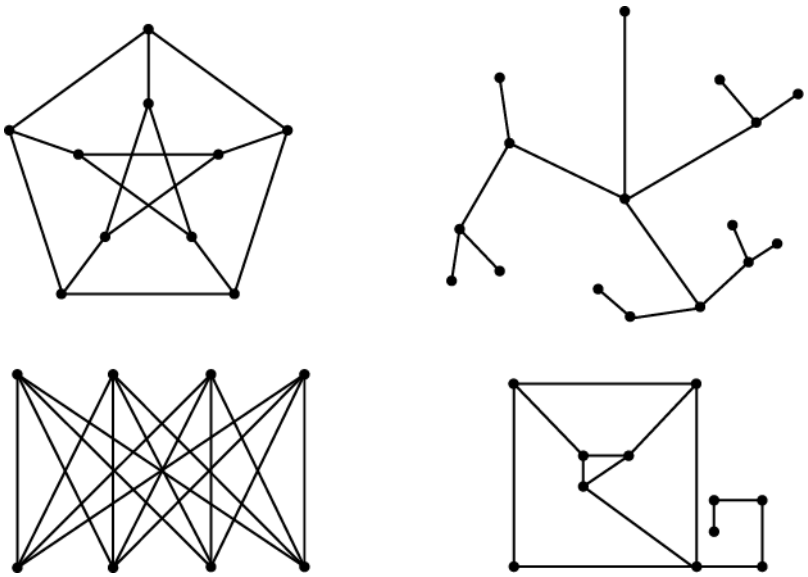

Figure 1.7. Some examples of graphs

A graph can be pictured as a collection of vertices and edges, where the edges are drawn as lines or curves and the vertices are drawn as points. Some examples of graphs can be seen in Figure 1.7. The top left graph in Figure 1.7 is called the Petersen graph which was constructed by Julius Petersen in 1898 and is a useful example of a small graph that provides counterexamples to many problems in graph theory [1]. The top right graph is called a tree; a tree is characterized as a graph

that contains no cycles where a cycle is a path along edges in the graph such that the path begins and ends at the same vertex with no other vertices nor any edges repeated within the path. The bottom left graph is a complete bipartite graph which is described so that each vertex in the top row is connected to every vertex in the bottom row and vice versa. Lastly, the bottom right graph does not belong to any significant family of graphs, but it is simply a collection of vertices and edges.

In topological graph theory, the graph is treated as a topological space which allows for the study of embedding graphs into topological spaces. In particular, topological graph theory is concerned with embedding graphs into compact surfaces. In the Topological Graph chapter, some results regarding graph embeddings are explored. In order to study embeddings into compact surfaces, it is necessary to introduce some preliminary definitions regarding surfaces and their construction.

1.3 Compact Surfaces

A *surface* is a topological space S that satisfies the following three properties:

(i) Every $x \in S$ is contained in an open set that is homeomorphic to the open disk.

(ii) For every $x_1, x_2 \in S$ there exist disjoint open sets u_1 and u_2 such that $x_1 \in u_1$ and $x_2 \in u_2$.

(iii) There is a basis for the topology on S consisting of countably many open sets.

Condition (i) indicates that a surface locally looks like an open disk (see Figure 1.8). Conditions (ii) and (iii) are technical requirements that restrict the class of surfaces to the simplest examples. The plane is an example of a surface.

All finite graphs can be embedded into a closed and bounded surface in Euclidean space, which is equivalent to compactness, thus the focus in this thesis will be

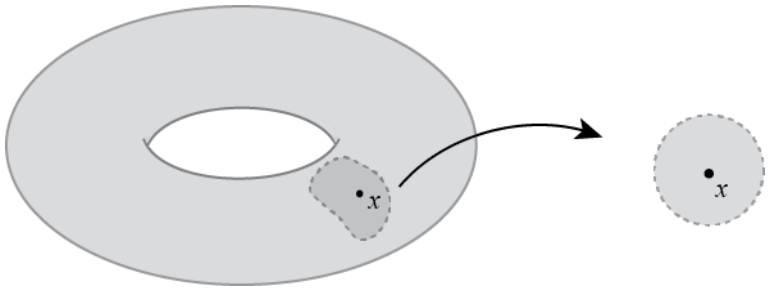

Figure 1.8. The torus is locally homeomorphic to the open disk.

compact surfaces. The examples of surfaces we present in the remainder of this section are all compact, however this is not a comprehensive list of all possible compact surfaces.

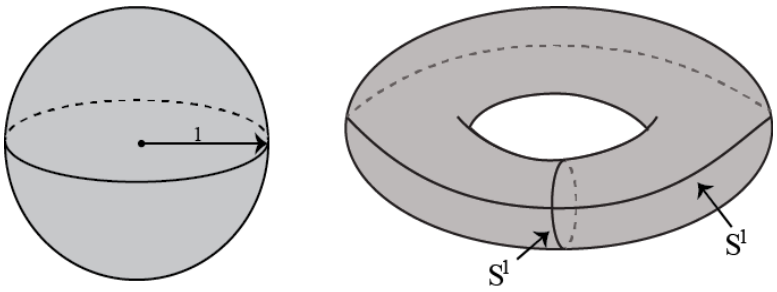

Figure 1.9. The 2-Sphere is on the left and the Torus is on the right.

First we will consider the *2-sphere*, denoted S^2, defined by the set of points distance one away from the origin in $\mathbb{R}^3$, see Figure 1.9. It is a compact surface. Here the sphere is considered to be a subspace of $\mathbb{R}^3$ and is given by

$$S^2 = \{(x_1, x_2, x_3) \in \mathbb{R}^3 | {x_1}^2 + {x_2}^2 + {x_3}^2 = 1\}.$$

Another compact surface is called the *torus*, denoted T, viewed as a product space it is $S^1 \times S^1$, where S^1 is the unit circle. The torus is formed by starting with one circle S^1 and for each point in the first circle there is a corresponding circle, thus forming the torus, as shown in Figure 1.9.

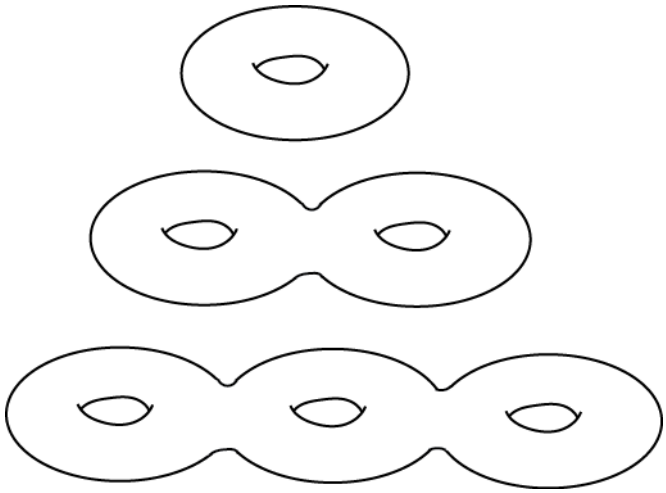

Figure 1.10. A single hole torus, a two hole torus, and a three hole torus.

Compact surfaces can be glued together, using an operator called the connected-sum that acts like addition on surfaces. Informally, we could create a torus with two holes by gluing together two tori (see Figure 1.10). Similarly, we can create an n hole torus by adding together n tori. A surface that is homeomorphic to an n hole torus is said to have *genus* n. The idea of gluing together different surfaces will be made precise in the Compact Surface chapter.

A compact surface can be represented in many different ways. For example, every compact surface can be represented as a quotient space where a partition is defined on a polygon. To see this consider the torus again. Not only can the torus be formed as described above, but it can be formed using a partition on the unit square that will identify the opposite edges of the square together (see Figure 1.11).

Informally, the torus can be formed by gluing opposite sides of the square together. After one of the pairs of sides is glued together the result is a cylinder with two circle boundaries remaining to be glued. Lastly, to glue the circle boundaries together it amounts to essentially pulling one circle around, stretching the cylinder, until the two circle boundaries meet. This gluing can be made precise by defining a

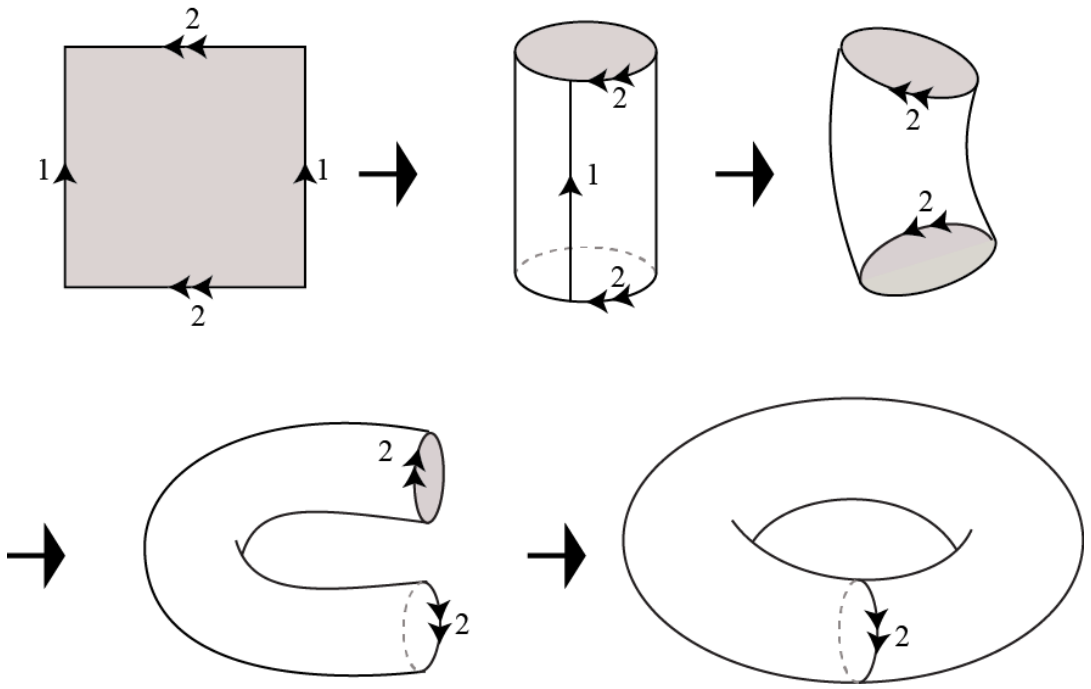

Figure 1.11. The construction of a torus from the unit square.

quotient topology that is induced by a partition on the unit square. To begin, take the product space $[0,1] \times [0,1]$, where $[0,1]$ is the unit interval in $\mathbb{R}$, to form the unit square. Then we define the partition on the unit square as follows:

$A_{x,y} = \{(x,y)\}$ for every x and y such that $x, y \in (0,1)$;

$B_y = \{(0,y),(1,y)\}$ for every y such that $y \in (0,1)$;

$C_x = \{(x,0),(x,1)\}$ for every x such that $x \in (0,1)$;

$D = \{(0,0),(0,1),(1,0),(1,1)\}$.

Then define $q : [0,1] \times [0,1] \to T$. Under the function q the first pair of opposite sides to be glued together is given by the set B_y which identifies the vertical sides of the square to a single edge in the torus. Similarly with the sets C_x and D the function q maps the horizontal edges of the square together and the corners of the square to a single point. The function q acts as the identity map on points in the

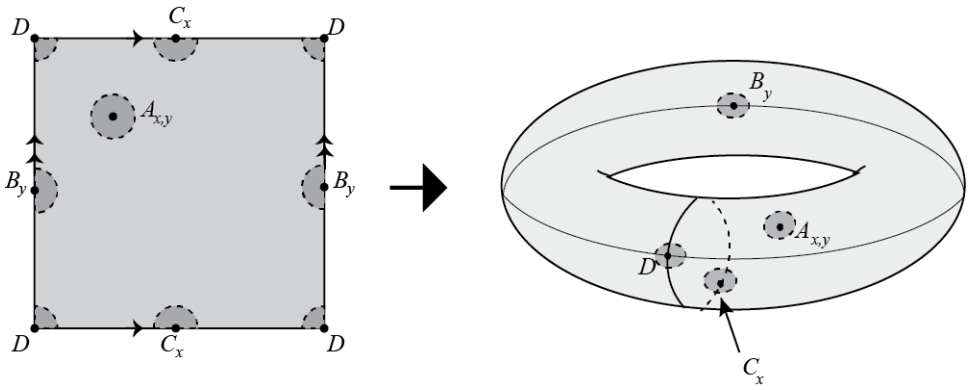

Figure 1.12. A partition on the unit square showing that every point in the torus is contained in an open-disk neighborhood.

set $A_{x,y}$. Figure 1.12 shows how this partition on the unit square results in a torus and how every point in the torus is in an open set homeomorphic to the open disk.

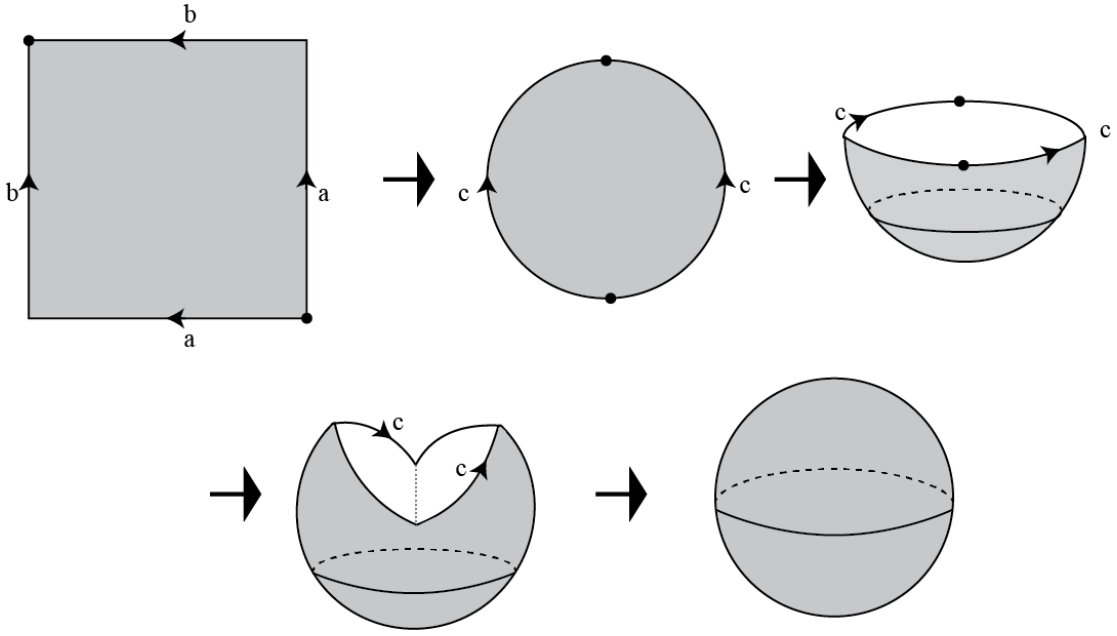

Figure 1.13. The construction of a sphere from the unit square.

Another compact surface that can be constructed in this way is the sphere. To see this, begin with a unit square with the edges labeled as they are in Figure 1.13. Then we will relabel the edges such that a new edge c represents the two edges of edge

a going into edge b. Finally, the c edges must be glued together. To visualize this imagine zipping up the boundary of a bowl along the c edges to create the sphere. As we will see in the Compact Surfaces Chapter, the sphere and torus are not the only compact surfaces that can be constructed by gluing pairs of edges together in polygons. In fact, we will see that every compact surface is homeomorphic to a polygon with pairs of edges identified together. When studying graph embeddings into surfaces it will be convenient to represent the compact surfaces in this way.

Chapter 2
THE TOPOLOGICAL GRAPH

A *topological graph* is constructed as a quotient space with the partition function defined on the sets of edges and vertices. Here we will not explicitly define the partition function; but, instead we will discuss the gluing of points that results from defining such a function. Let E_G denote the set of edges and let V_G denote the set of vertices of some graph G. Let $I = [0, 1] \subset \mathbb{R}$. Suppose i and j are vertices in V_G connected by an edge; denote this edge $e_{i,j}$. To form the edge $e_{i,j}$ in the topological graph G, we will glue the 0 end point of I to vertex i and we will glue the 1 end point of I to vertex j; creating an edge connected to vertices i and j. Each edge of the graph will be formed in this way. The set of edges E_G will describe the gluing that must occur to create a picture of the graph G. Since each edge is a closed and

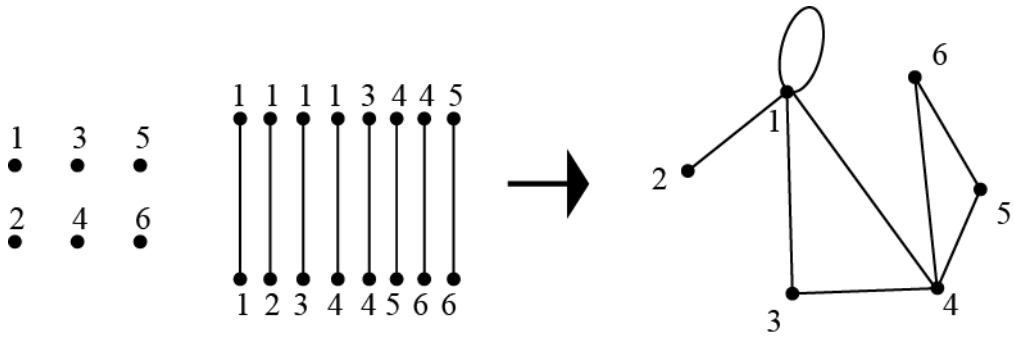

Figure 2.1. A graph is obtained by gluing the end points of the edges to the corresponding vertices

bounded interval with vertices glued to the end points, the specific vertices prescribe how the edges will be glued together creating G. For instance, let $e_{i,j}, e_{j,k} \in E_G$ be edges in the graph G. Since both of the edges $e_{i,j}$ and $e_{j,k}$ are formed in the way described, we know these edges are incident to vertices i and j and vertices j and k, respectively. Notice, these two edges contain a shared vertex, so the two edges are

glued together on the shared vertex j. In other words, two edges of the graph will be glued together only if the edges contain a shared vertex as a subspace. Consider the vertex and edge sets presented in Figure 2.1. Here V_G consists of 6 vertices, each labeled with a number, and E_G has 8 edges. For instance, the first edge shown in Figure 2.1 has vertex 1 at each end. This implies that both ends of the edge are glued to vertex 1 forming a loop. Similarly, the second edge is labeled with vertex 1 and vertex 2, so this edge will also glue to vertex 1 but the other end of the edge will glue to vertex 2. We can continue gluing all of the edges together in this way to create the graph G as shown in Figure 2.1.

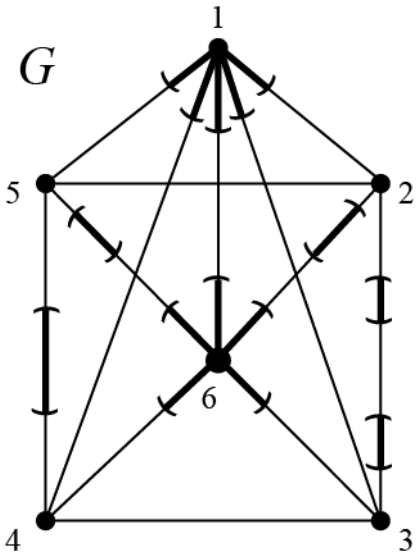

Figure 2.2. Some open sets of the graph G.

Since a graph constructed as described is a topological space, we can define a basis set, $\mathcal{B}$, that determines the open sets of a graph. The basis elements of a graph are open intervals in each edge and open stars about each vertex (see Figure 2.2). The open stars of a graph are considered to be subsets of the graph about each vertex where it contains part of every edge incident to that vertex but no other vertices. In Figure 2.2, the open stars about vertices 1 and 6 are shown and some open intervals

in edges are shown as well. Notice that every graph can be reconstructed by its basis elements; specifically, if one takes the union of all open stars and open intervals in the edges then the whole graph is the result. This implies that every graph is both an open and closed set.

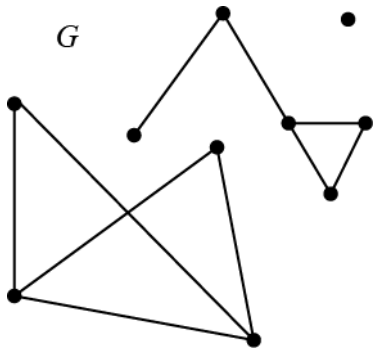

Figure 2.3. The graph shown is an example of disconnected graph with three components

The graph shown in Figure 2.2 is considered to be a connected graph. Connected graphs only have one component, however a graph does not necessarily have to be connected. The graph shown in Figure 2.3 is disconnected and is a union of finitely many components. In general, the graphs considered here will be connected. For further reading on the graph theory background see [2], [7], and [1].

2.1 Introduction and Terminology

In the following section, it is necessary to introduce some basic definitions of a graph in order to study the embedding of graphs into other topological spaces. A graph is considered to be finite if both the vertex set and the edge set are finite sets. It can be assumed that every graph mentioned here is finite. Let G be a graph and let $e_{i,j} \in E_G$ denote an edge connecting the vertices $i, j \in V_G$. The edge $e_{i,j}$ is said to be *incident* to vertices i and j and similarly vertex i and vertex j are said to be incident to edge $e_{i,j}$. In a graph, one can travel from a vertex to another by

traversing a path of edges in a particular order within the graph. In graph theory, such a path is called a walk. More formally, a *walk* in graph G from vertex i to vertex j is given by a sequence of edges, $e_{i,n}e_{n,l}\dots e_{k,m}e_{m,j}$, such that $e_{i,n}$ indicates traveling from vertex i to vertex n along the edge incident to the two vertices. A walk is considered to be closed if it begins and ends on the same vertex. Further, a walk is considered to be directed if each edge in the walk has an associated direction. In particular, in a walk the edges $e_{i,j}$ and $e_{j,i}$ are representing the same edge of the graph but the edges are traversed in opposite directions.

In a graph, if l is an edge that connects a vertex to itself then the edge l is said to be a *loop*. If two edges, e and e^* are incident to the same pair of vertices, then e and e^* are called *parallel edges*. A graph G is said to be *simple* if it contains no parallel edges and no loops. For each vertex $v \in V_G$, the *degree of vertex* v is given by the number of edges incident to the vertex v, counting an edge twice if it is a loop. Finally, two graphs, G and $G^{'}$, are said to be *topologically equivalent* if there exists a homeomorphism, $h : G \to G^{'}$. On the other hand, two graphs are *graph equivalent* if there exists a homeomorphism, $f : G \to G'$ such that f bijectively maps the vertex set of G to the vertex set of G'.

Theorem 2.1.1. *For every graph G there exists a simple graph $G^{'}$ such that G is homeomorphic to $G^{'}$.*

Proof. Let G be a graph and let v_1 and v_2 be vertices that are incident to an edge e. Suppose there is another edge, call it h, that is parallel to the edge e. To construct the simple graph G' so that there are no parallel edges, replace the edge e with two distinct edges e_1 and e_2 and add another vertex v_3, as shown in Figure 2.4.i. The edges are placed in G' so that e_1 is incident to v_1 and v_3 and e_2 is incident to the vertices v_3 and v_2. In the resulting G', the edges e_1 and e_2 with the three vertices will be homeomorphic to the original edge e with the two vertices in the graph G.

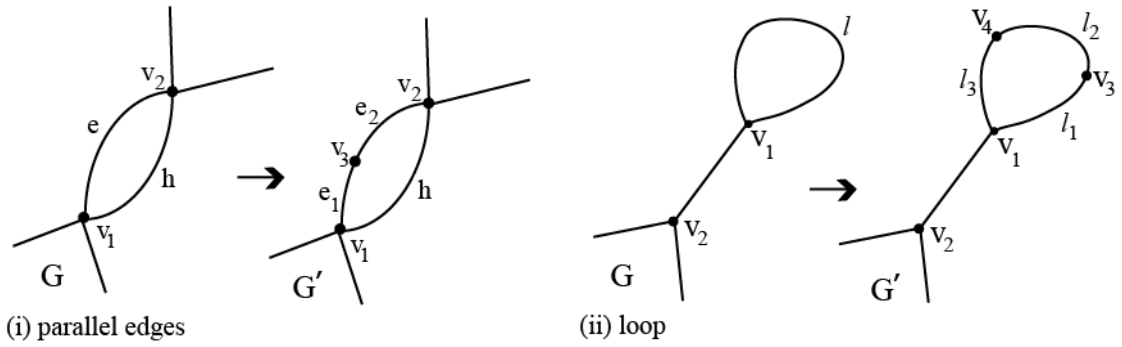

Figure 2.4. The process of turning a graph with parallel edges or loops into a simple graph.

Now suppose the graph G has an edge l joining vertex v_1 to itself, making the edge a loop. To construct the simple graph G' replace l with three distinct edges l_1, l_2, and l_3 and add two distinct vertices v_3 and v_4, as shown in Figure 2.4.ii. The edge l is replaced so that the edge l_1 is incident to vertices v_1 and v_3, the edge l_2 is incident to vertices v_3 and v_4, and the edge l_3 is incident to vertices v_4 and v_1. So in the resulting G' the edges l_1, l_2, and l_3 with associated vertices, taken together, is homeomorphic to the original loop, l, in G. Use this construction of a simple graph G' whenever parallel edges or loops appear in the graph G creating a homeomorphic simple graph. □

As a consequence of the previous theorem, it can be assumed without loss of generality that every graph considered here is a simple graph. In graph theory, any collection of edges and vertices can form a graph; however, throughout the history of graph theory certain families of graphs have been a focus of study. One such family consists of the complete graphs, where a graph G is complete if each pair of distinct vertices is connected by an edge. If G has n vertices then it is said to be the *complete graph on n vertices*, denoted K_n [2]. The graph K_1 consists of the single vertex set, K_2 is given by two vertices connected by a single edge, and K_3 consists

of 3 vertices and 3 edges making a triangle (see Figure 2.5). From this definition, it follows that each vertex in K_n has degree $n-1$.

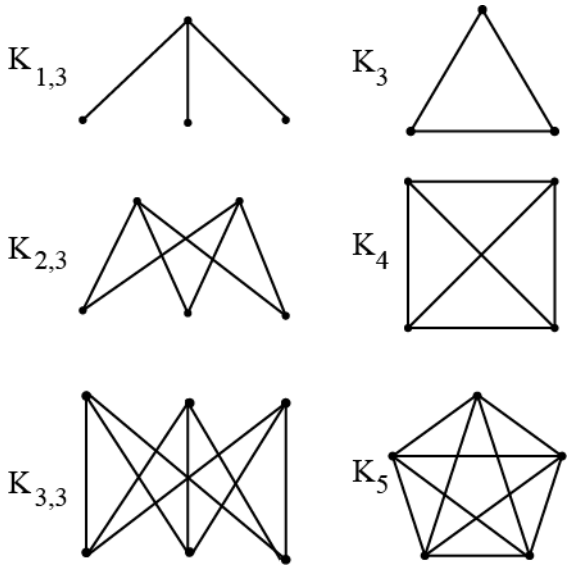

Figure 2.5. (i)The complete graphs K_3, K_4, and K_5 are shown on the right. (ii)The complete bipartite graphs $K_{1,3}$, $K_{2,3}$, and $K_{3,3}$ are shown on the left.

Another family of graphs is the set of *bipartite graphs*. Let G be a bipartite graph. Here vertex set V_G is partitioned into two disjoint subsets, $V_1, V_2 \subset V_G$ such that each edge has one end point contained in V_1 and the other end point contained in V_2 [2]. As a consequence, no two vertices in the same set V_1 or V_2 will be connected by an edge. A more commonly studied family of graphs is the *complete bipartite graphs on m and n vertices*, denoted $K_{m,n}$. For the complete bipartite graphs the vertex set $V_{K_{m,n}}$ consists of $m+n$ vertices partitioned into two subsets V_m consisting of m vertices and V_n consisting of n vertices. As in the previous definition of complete graph, every distinct pair of vertices v and w such that $v \in V_m$ and $w \in V_n$ is connected by an edge and no other edges. From this definition of $K_{m.n}$, each vertex $v \in V_m$ has degree n and each vertex $w \in V_n$ has degree m.

The complete bipartite graph on three vertices $K_{3,3}$ is commonly referred to as the utility graph due to its role in the Water, Gas, Electric Problem first published

in 1917 in *Amusements in Mathematics* authored by Henry Dudeney [3]. In this problem, Dudeney asks if it is possible in a two-dimensional setting to have three houses connected on paper to three utilities (electric, gas, and water) such that the connections between the houses and utilities do not intersect. This brain teaser is impossible and this is a consequence of an important result indicating that the complete bipartite graph $K_{3,3}$ cannot be embedded in the plane.

2.2 Planar Graphs and Some Embedding Results

Let X and Y be topological spaces. Recall, an *embedding* of X in Y is a function $f : X \to Y$ that maps X homeomorphically to the subspace $f(X)$ in Y. Here it is of interest to embed topological graphs into compact surfaces. An embedding of a graph G into a surface S can be considered to be a picture of the graph on the surface such that edges in $f(G)$ only intersect in shared vertices and do not intersect in any other way. Let G be a graph embedded into a compact surface, and define a *face* of the embedded graph to be a component of the complement of the graph in the surface. An embedding of a graph is called a *2-cellular embedding* if each face of the graph is homeomorphic to an open-disk. For instance, an embedding of a graph is not 2-cellular if there exists a face that is not homeomorphic to the open disk. To see this, consider embedding the graph K_3 into the torus. We could embed the graph directly onto the side of the torus and not have the edges of the graph go through the hole. In this case, the embedding will not be 2-cellular because there exists a face of the embedded graph that contains the hole of the torus and hence is not homeomorphic to the open disk. The task of embedding graphs in compact surfaces requires determining the appropriate surface for each picture of a graph, however if one wanted to embed a graph into three-space we can do so without too much difficulty.

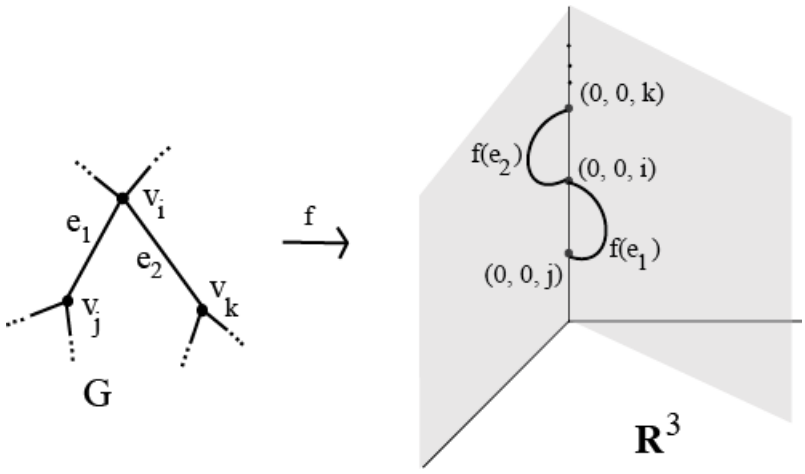

Figure 2.6. An embedding of graph in $\mathbb{R}^3$.

Theorem 2.2.1. *Every graph can be embedded in* $\mathbb{R}^3$.

Proof. Let G be a graph with n vertices $v_1, v_2, \ldots, v_n$ and m edges $e_1, e_2, \ldots, e_m$. We will define an embedding $f : G \to \mathbb{R}^3$. To begin, define f on each of the vertices to be $f(v_i) = (0, 0, i)$ for all $i = 1, 2, \ldots, n$. Thus f maps the set of vertices bijectively to $\{(0, 0, i) | i = 1, \ldots, n\}$ in the z-axis in $\mathbb{R}^3$. Let $P_1, P_2, \ldots, P_m$ be distinct half-planes emanating from the z-axis. If e_i is an edge incident to vertices v_j and v_k, then extend f to the edge e_i such that it maps e_i homeomorphically to the semi-circle in the half-plane P_i adjoining $f(v_j) = (0, 0, j)$ and $f(v_k) = (0, 0, k)$. See Figure 2.6 for an illustration of this embedding. The map f, as defined, is an embedding of the graph G into $\mathbb{R}^3$.

□

Because embeddings are bijective mappings, the previous theorem allows us to embed any graph G into $\mathbb{R}^3$ such that no edges intersect. As has been shown, it is a relatively simple task to construct an embedding for a graph into three-space; however, it is more challenging to determine whether or not a graph can be embedded into the plane. A graph G is considered to be *planar* if G can be embedded into $\mathbb{R}^2$.

Later we will see another equivalent defintion of a planar graph, namely one that can be embedded into the 2-sphere.

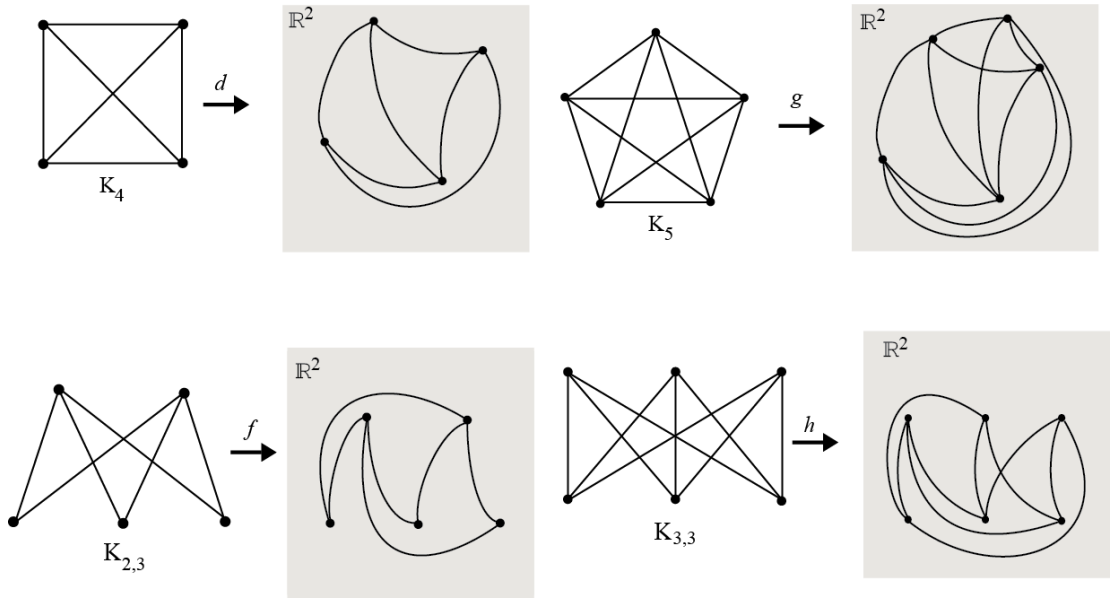

Figure 2.7. The graphs K_4 and $K_{2,3}$ are embeddable in the plane and the graphs K_5 and $K_{3,3}$ are not embeddable in the plane.

Consider the complete graph on n vertices, K_n. If $n \leq 4$ then K_n is planar (see Figure 2.7); however, if $n \geq 5$ then K_n is nonplanar. Also, Figure 2.7 shows how $K_{2,3}$ can be embedded in the plane. However, for the complete bipartite graph on m and n vertices, $K_{m,n}$, if $m, n \geq 3$ then $K_{m,n}$ is nonplanar. See the Figure 2.7 for examples of embeddings of K_4 and $K_{2,3}$. Because our functions $g : K_5 \to \mathbb{R}^2$ and $h : K_{3,3} \to \mathbb{R}^2$ in Figure 2.7 fail to be embeddings, that does not prove no such embeddings exist. That is asserted by the following result first proved by Jazimierz Kuratowski in 1930 [9].

Theorem 2.2.2 (Kuratowski's Theorem)**.** *A graph is planar if and only if it contains no subspace homeomorphic to $K_{3,3}$ and no subspace homeomorphic to K_5.*

The proof will be omitted here but can be found in [9], [12]. This result is a powerful tool to determine whether a graph is planar. However, it is rather difficult

to determine if a graph contains no subspace homeomorphic to K_3 or contains no subspace homeomorphic to $K_{3,3}$. So, the converse of the result is much easier to apply to determine when a graph is nonplanar. In other words, for any graph G, if $K_{3,3}$ or K_5 can be embedded in the graph, then G cannot be embedded in the plane. Recall Dudeney's Water, Gas, and Electric problem. As mentioned previously, it is impossible in two-dimensions to connect each house to each utility without crossing the lines. This is because the water, gas, and electric problem is modeled by the complete bipartite graph $K_{3,3}$. By Kuratowski's Theorem, $K_{3,3}$ is nonplanar and hence Dudeney's brain teaser is impossible to solve.

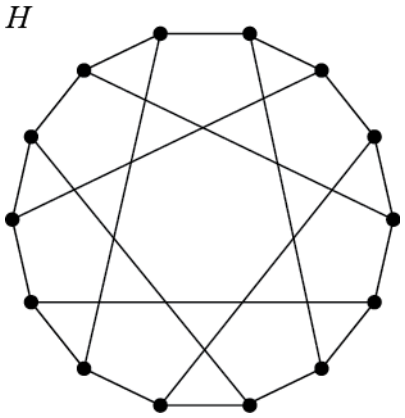

Figure 2.8. The Heawood graph.

Consider the Heawood graph H shown in Figure 2.8.To determine if the Heawood graph is planar or not, we can apply Kuratowski's Theorem. If either K_5 or $K_{3,3}$ can be embedded in the Heawood graph, then H contains one of those graphs as a subspace and would hence be nonplanar. It turns out that the complete bipartite graph $K_{3,3}$ can be embedded into H. Notice, H cannot contain K_5 as a subspace since each vertex in the complete graph K_5 has degree 4, but in the Heawood graph each vertex has degree 3. An embedding of $K_{3,3}$ into the Heawood graph is shown in Figure 2.9, where the embedded $K_{3,3}$ graph is indicated by bolded edges.

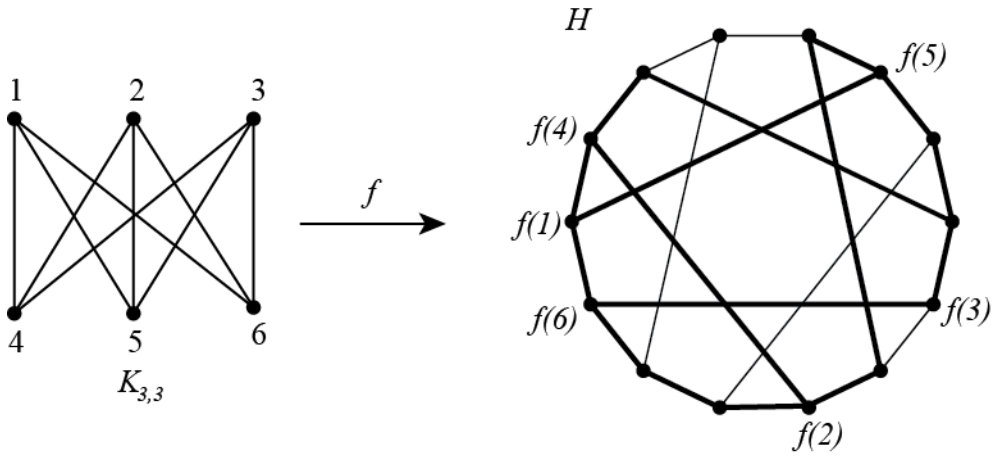

Figure 2.9. An embedding of $K_{3,3}$ into the the Heawood graph is shown, indicating the Heawood graph is nonplanar by Kuratowski's Theorem.

Another important result involving planar graphs is what is known as Euler's formula, and later we will see a generalization of this formula regarding compact surfaces.

Theorem 2.2.3 (Euler's Formula for Planar Graphs)**.** *Let G be a nonempty graph in the plane having C components. If there are V vertices, E edges, and F faces associated with G, then $V - E + F = C + 1$.*

The proof of Euler's Formula for Planar graphs will be omitted here but can be found in [1]. Euler's Formula for planar graphs is not a test for planarity since it requires an embedding of the graph to determine the number of faces. Consider the embedded graph shown in Figure 2.10. The graph G has one component, 12 vertices, and 14 edges. Lastly, we need to determine the number of faces of this embedded graph; to do so, take the complement of the graph in the plane and what remains are four pieces, each labeled with f_i where $i = 1, \ldots, 4$. Now we can apply Euler's Formula for planar graphs to see that $V - E + F = C + 1$ for this planar graph. Every graph considered in the remainder of this paper is connected, so each graph will only have one component. Thus from Euler's Formula for Planar graphs, if a graph is planar then $V - E + F = 2$.

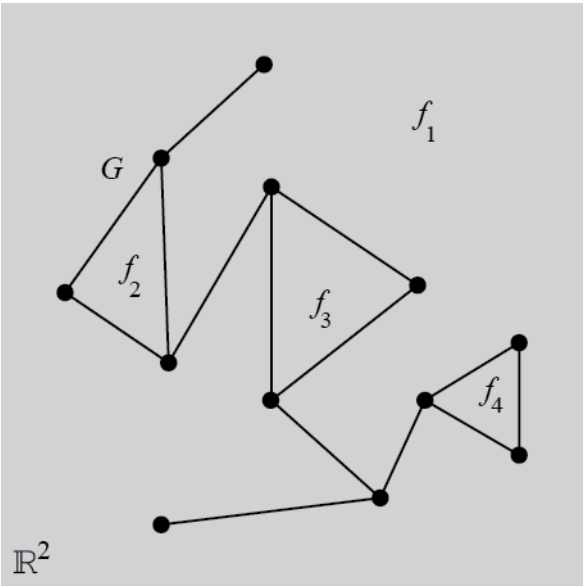

Figure 2.10. A graph G embedded into the plane that satisfies Euler's Formula for planar graphs.

Another useful tool that allows us to determine which graphs are planar and where these nonplanar graphs can be embedded is the crossing number of a graph. First, we must make precise the idea of a drawing of a graph to understand the crossing number. Given a graph G, a *drawing of* G is a continuous function $d : G \to \mathbb{R}^2$. Let d be a drawing of a graph. The drawing, d, is considered to be a *good drawing* if it satisfies the following conditions: no point in the image of d corresponds to more than two points of the graph; there are only finitely many points in the image of d that correspond to two points of the graph, these are called the *crossing points* of the graph; and no crossing point of the drawing corresponds to a vertex of the graph. See Figure 2.11 for some examples of good drawings of graphs.

Theorem 2.2.4. *Every graph G has a good drawing.*

Proof. Let G be a graph. Embed G into $\mathbb{R}^3$ using the embedding f as defined in the proof of Theorem 2.2.1 so that the vertices are mapped to the z-axis and the

edges map to semi-circles in half-planes emanating from the z-axis. To obtain a good drawing of G, compose the embedding of G in $\mathbb{R}^3$ with a projection onto a plane P in $\mathbb{R}^3$. Note that this does not necessarily produce a good drawing of G at this point. However, if the plane P is chosen such that it is parallel to the z-axis and not perpendicular to any of the half-planes containing the images of the edges of G, then the resulting drawing of G will satisfy two of the three necessary requirements for a good drawing. In particular, this drawing of G only has finitely many crossing points and none of these crossing points correspond to vertices of G. However, if any of the points in the resulting drawing of G correspond to more than two points in the graph G, then simply deform the semi-circles in the drawing such that any point in the drawing of G corresponds to no more than two points in the graph G. Thus, this drawing of G satisfies the requirements for a good drawing of a graph. □

Since it is known that all graphs have good drawings, we can now define an important property of a graph known as the crossing number. Let *the crossing number of a graph* G, denoted $\nu(G)$, be the minimum number of crossing points in any good drawing of G. Any good drawing of a graph provides an upper bound for the crossing number of a graph. For example, from Figure 2.11 we can see that the complete graph K_4 has a crossing number of 0, while K_5 has a crossing number at most 1 and K_6 has a crossing number at most 3. Similarly, the complete bipartite graph $K_{1,n}$ has a crossing number of 0 for every n but the graph $K_{3,4}$ has a crossing number of at least 1 and at most 2 by the good drawing shown in Figure 2.11.

The crossing number of a graph is an important property of the graph when studying graph embeddings, as the following theorems illustrate. The proof of the following theorem follows the proof presented in [1].

Theorem 2.2.5. *(i) Crossing number is a topological invariant of graphs; that is, if graphs* G_1 *and* G_2 *are homeomorphic then,* $\nu(G_1) = \nu(G_2)$.

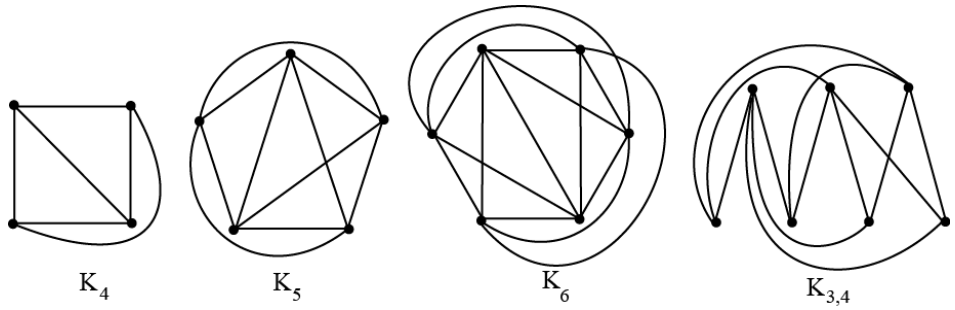

Figure 2.11. Good drawings of the graphs K_4, K_5, K_6, and $K_{3,4}$ showing each graph's crossing number.

(ii) A graph can be embedded in the plane if and only if its crossing number is 0.

Proof. The second part is immediate; if the crossing number of a graph is zero then it is trivial to construct an embedding of the graph in the plane and vice versa. Consider the first part. Let G_1 and G_2 be homeomorphic graphs. Suppose that $d_1 : G_1 \to \mathbb{R}^2$ is a good drawing of G_1. We claim that there exists a homeomorphism $h : G_2 \to G_1$ such that the function d_2, given by $d_2 = d_1 \circ h$, is a good drawing of G_2. Thus, for every good drawing d_1 of G_1 there is a corresponding good drawing d_2 of G_2 such that the images of d_1 and d_2 are identical. It follows that $\nu(G_1) = \nu(G_2)$.

It remains to show the existence of such a homeomorphism $h : G_2 \to G_1$. Given $h^* : G_2 \to G_1$, a homeomorphism, the only way that $d_1 \circ h^*$ could fail to yield a good drawing is if h^* maps one or more degree-2 vertices of G_2 to points of G_1 that correspond to the crossing points of the drawing d_1. If this were to occur, the homeomorphism h^* can be adjusted to a homeomorphism h mapping all degree-2 vertices of G_2 to points of G that do not correspond to crossing points of d_1. □

An immediate result of this theorem is that a graph G is nonplanar if and only if $\nu(G) \geq 1$. The previous theorem allows us to determine if a graph can be embedded in the plane, however it does not allow one to determine where a nonplanar graph could be embedded. The following theorem begins the task of determining what sort of surface into which a nonplanar graph could be embedded.

Theorem 2.2.6. *If a graph G has crossing number n, then G can be embedded in a surface homeomorphic to a torus with n holes.*

Proof. Let G be a graph with an associated good drawing with n crossings. Using an embedding of the plane in the sphere, we can map the good drawing of G from the plane to the sphere. In a neighborhood of each crossing in the resulting image in the sphere, put a tube, with one of the crossing strands running over the tube and the other crossing strand running through the tunnel in the surface that results from the addition of the tube, as shown in Figure 2.12.

Figure 2.12. The addition of a tube to the sphere to eliminate a crossing in the embedding of the graph.

Continue this process until all of the crossings are removed by adding tubes. The resulting embedding of G maps into a sphere with n tubes, a surface homeomorphic to a torus with n holes. □

The complete graph K_5 and the complete bipartite graph $K_{3,3}$ both have a crossing number of 1, and by the previous theorem we know that these graphs are embeddable on a torus (see Figure 2.13). To see how these graphs are embedded on the torus, we will construct the embedding directly for K_5. To do so, we use the quotient space representation of the torus, in other words the unit square $[0,1]\times[0,1]$ with the opposite edges identified together to form the torus. First map the set of 5 vertices to the circle of radius $0 < \varepsilon < 1/2$ centered in the unit square. Map the 5 edges that connect the vertices to their immediate neighbors on the circle in the torus and for the remaining edges we will use the gluing of the opposite sides of the unit square to avoid any crossings in the embedding of K_5 in the torus (see Figure

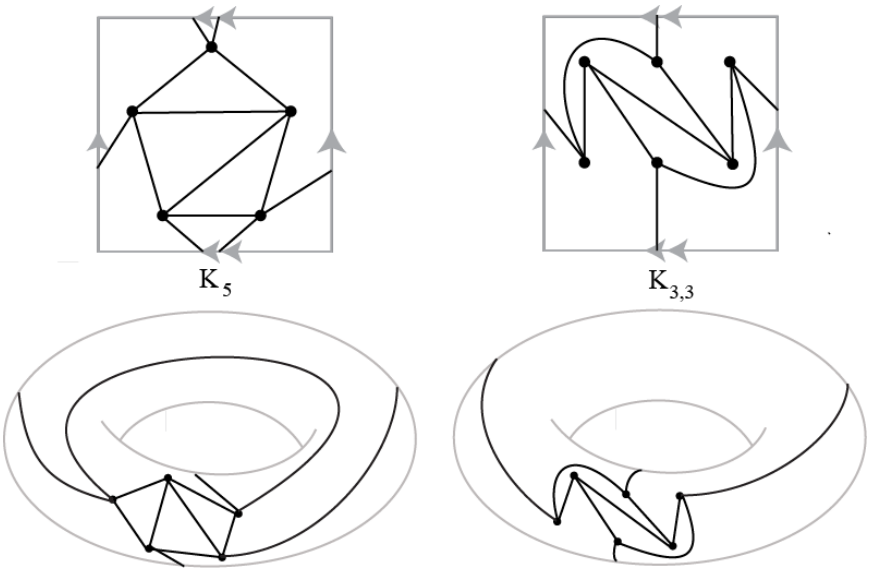

Figure 2.13. An embedding of the complete graph K_5 and the complete bipartite graph $K_{3,3}$ on a torus.

2.13). As shown in Figure 2.13, a similar construction of an embedding can be done for the complete bipartite graph $K_{3,3}$.

However, this theorem does not provide conditions for determining the minimal genus-m surface that a graph with crossing number n can be embedded. To see this, consider the complete bipartite graph $K_{3,4}$. As previously shown the graph $K_{3,4}$ can be drawn with only two crossings (see Figure 2.11). To see that the crossing number of $K_{3,4}$ is indeed two, suppose we could remove one of the two crossings in the good drawing of $K_{3,4}$. However, removing an edge from a crossing would not change the fact that we can embed $K_{3,3}$ into the resulting graph. Since we know $K_{3,3}$ is a subspace of $K_{3,4}$, this implies $\nu(K_{3,4}) > 1$, and as observed previously $\nu(K_{3,4}) \leq 2$. Therefore the crossing number of $K_{3,4}$ is 2. Thus, Theorem 2.2.6 implies that $K_{3,4}$ can be embedded in torus with two holes. But an embedding can be constructed for $K_{3,4}$ such that it embeds in a torus with a single hole (see figure 2.14).

Theorem 2.2.6 guarantees the existence of an embedding into a compact surface for any given good drawing of a graph. In Theorem 2.2.6, its usefulness is limited

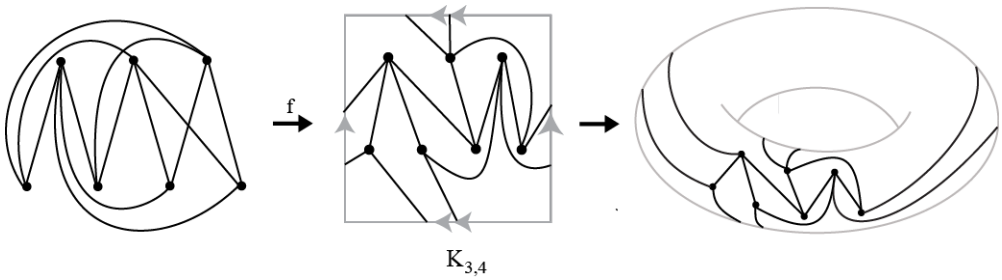

Figure 2.14. An example of the graph complete bipartite graph $K_{3,4}$ embedded into a torus.

since it requires a good drawing of a graph in order to determine the crossing number of the graph. In general, it is a difficult problem to determine the crossing number of an arbitrary graph and in fact it is an incomplete problem for many well-known families of graphs.

2.3 Rotation Systems of Graphs

In this section, we will introduce a combinatorial approach to embedding graphs and this approach will be useful in the development of an embedding algorithm in chapter 4. Let $G = \{V, E\}$ where $V = \{1, 2, 3, \ldots, n\}$ and E is given by $E = \{e_{i,j} : \text{for some } i, j \in V\}$. Recall that an edge labeled $e_{i,j}$ indicates that the edge is incident to the vertices i and j. Any given good drawing of a graph determines a cyclic ordering of edges about each vertex. Define a *rotation system of a graph* $\Pi = \{\pi_i : i \in V\}$ where the *vertex code* π_i is given by a cyclic permutation of edges about vertex i [10].

A rotation system is a combinatorial object of the graph, only recording information about the order of edges about each vertex. However, a single rotation system of a graph may correspond to many different good drawings of the graph. Alternatively, we could obtain a rotation system from any good drawing of a graph; we may even produce the same rotation system from different drawings of a graph. Most importantly, we will use rotation systems to produce an embedding of a graph

without having knowledge of the associated drawing of the graph. We will show in the Embedding Algorithm chapter how a rotation system can be used to create an embedding of a graph. The different rotation systems of a graph may result in embeddings of the graph into different surfaces. For example, consider the complete

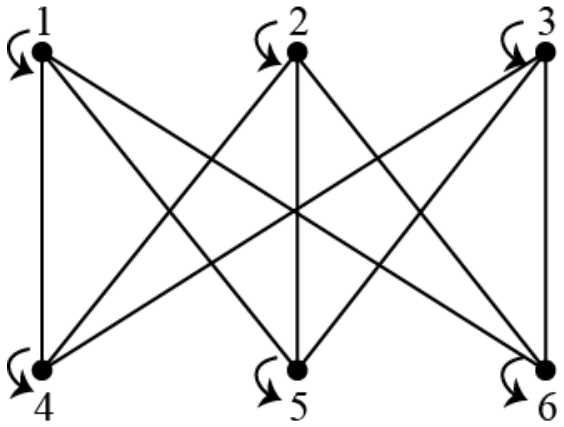

Figure 2.15. A rotation system Π_1 of $K_{3,3}$ that records the particular counterclockwise ordering about each vertex and produces a good drawing of the graph.

bipartite graph $K_{3,3}$. A rotation system $\Pi_1 = \{\pi_i : i = 1, \ldots, 6\}$ of this graph is given by the following:

$$\pi_1 : (e_{1,4}; e_{1,5}; e_{1,6}) \quad \pi_4 : (e_{4,3}; e_{4,2}; e_{4,1})$$
$$\pi_2 : (e_{2,4}; e_{2,5}; e_{2,6}) \quad \pi_5 : (e_{5,3}; e_{5,2}; e_{5,1})$$
$$\pi_3 : (e_{3,4}; e_{3,5}; e_{3,6}) \quad \pi_6 : (e_{6,3}; e_{6,2}; e_{6,1})$$

The rotation system Π_1 for $K_{3,3}$ can be represented as the good drawing of the graph seen in Figure 2.15.

The distinct rotation systems of a graph may induce embeddings of the graph into different surfaces. For instance, we can define another rotation system, $\Pi_2 = \{\pi_i : i = 1, \ldots, 6\}$ for the complete bipartite graph $K_{3,3}$ as follows:

$$\pi_1 : (e_{1,5}; e_{1,6}; e_{1,4}) \quad \pi_4 : (e_{4,1}; e_{4,3}; e_{4,2})$$

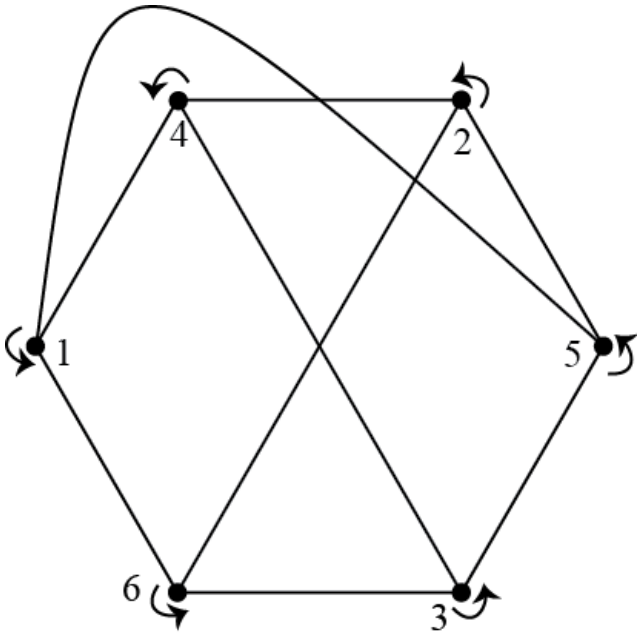

Figure 2.16. Another rotation system Π_2 of $K_{3,3}$ that induces a different good drawing of the graph

.

$$\pi_2 : (e_{2,4}; e_{2,6}; e_{2,5}) \qquad \pi_5 : (e_{5,2}; e_{5,1}; e_{5,3})$$
$$\pi_3 : (e_{3,5}; e_{3,4}; e_{3,6}) \qquad \pi_6 : (e_{6,3}; e_{6,2}; e_{6,1})$$

The vertex codes of this new rotation system of $K_{3,3}$ also record a cyclic counterclockwise ordering of edges about each vertex. The rotation system Π_2 can be visualized via the good drawing of $K_{3,3}$ in Figure 2.16. Even though the drawings of $K_{3,3}$ appear different in Figures 2.15 and 2.16, the graphs are of course topologically equivalent. We will see in the Embedding Algorithm Chapter that the two rotation systems Π_1 and Π_2 will induce embeddings of $K_{3,3}$ into distinct surfaces via the embedding algorithm.

Using the notion of rotation systems of a graph, an embedding algorithm for graphs will be developed in the succeeding chapters. A rotation system of a graph will determine an embedding of a graph in a compact surface. Further, the embedding algorithm will identify the faces of the embedded graph and enable us, by

gluing the faces together, to determine the surface in which the graph is embeddable. However, in order to identify which surface the embedding algorithm constructs for a given graph, it is necessary to understand the classification of compact surfaces theorem which is discussed in the following chapter.

Chapter 3
CONSTRUCTING AND CLASSIFYING COMPACT SURFACES

For further reading on the materials presented regarding compact surfaces and the compact surface classification theorem see [6], [1], and [7].

3.1 Construction of Surfaces

Many familiar and unfamiliar compact surfaces can be constructed using the notion of quotient spaces and gluing edges of polygons together. Recall, the torus can be constructed from the unit square when the opposite edges of the square are identified together. In a similar way, we also saw how the sphere can be constructed from gluing pairs of edges in a square. The torus and sphere are not the only compact surfaces that can be created by gluing different pairs of edges of a square together; in fact, there are two more distinct compact surfaces that can be created from a quotient topology on the unit square. One such compact surface is called the *Klein bottle*, denoted K. To construct the Klein bottle start with the unit square $[0,1]\times[0,1]$ and glue the right and left edges of the square together as in the partition for the torus (see Figure 3.1). Next we glue the top and bottom edges together such that the right side of the top edge glues to the left side of the bottom edge, and the left side of the top edge glues to the right side of the bottom edge.

The quotient space for the Klein bottle first forms a cylinder by identifying the left and right edges of the square together, as shown in Figure 3.1. Then when identifying the top and bottom edges together, one must bend the top edge of the cylinder down and pass it through the side of the surface to connect it to the bottom edge so that the arrows match properly. Note that, the Klein bottle does not actually exist in 3-space since there is no circle of intersection when the top of

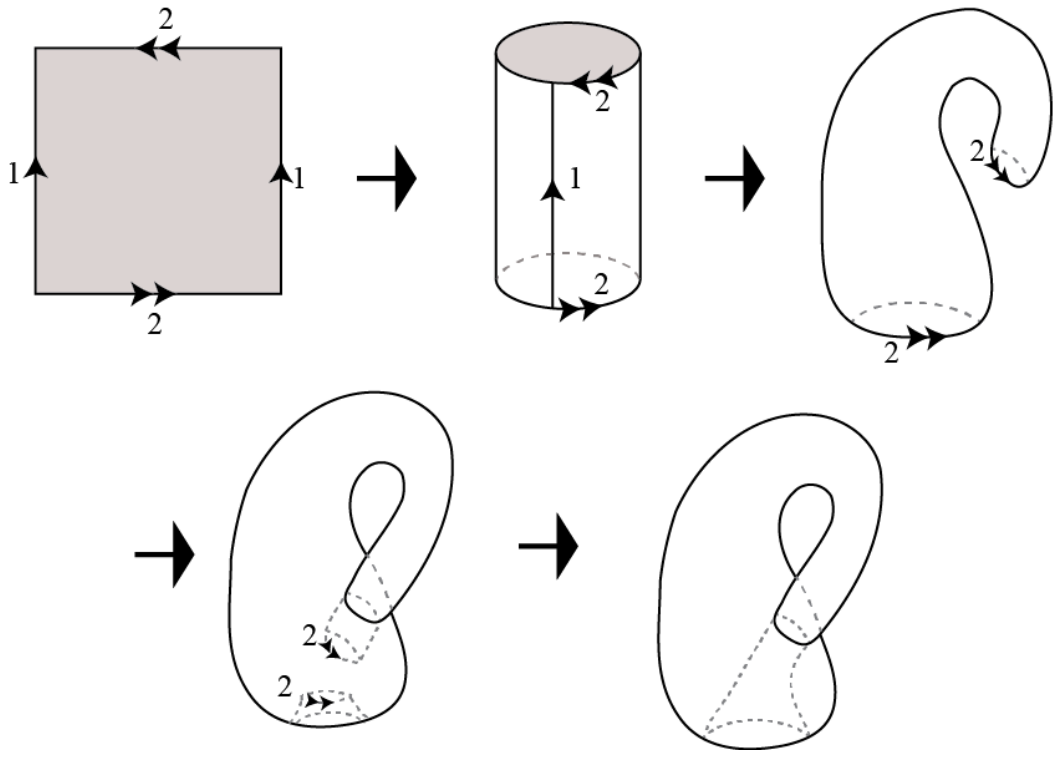

Figure 3.1. The construction of a Klein bottle using a quotient topology on the unit square.

the cylinder passes through, so the Klein bottle can only be constructed in 4-space without self-intersection.

In a similar fashion we can create the last fundamental compact surface, the *projective plane*, denoted P. Once again we start with the unit square with opposite sides identified together (see Figure 3.2). In this case, both pairs of edges a and b are identified together in opposite directions. Notice that the edge a points towards the edge b in both pairs. As shown in Figure 3.2, we can identifiy the two edges a going into b as a single edge, denoted c, resulting in a circle with two edges identified together in opposite directions. In particular, the pair of edges c will be glued together with the top part of the left semicircle gluing to the bottom part of the right semicircle and vice versa. Let p be the points at the north and south poles of the unit disk, also denoting the division between the two semicircles, then as

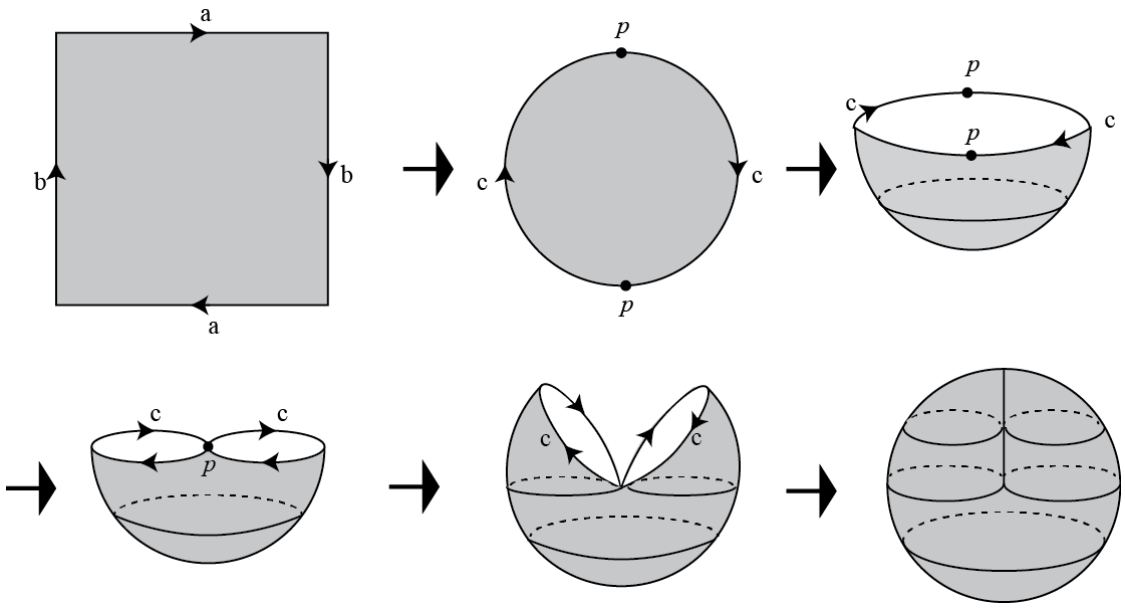

Figure 3.2. The construction of the projective plane using a quotient topology on the unit square.

with the sphere we start with a bowl-like shape. First, we glue together the points p such that the bowl is pinched in the middle along the rim. It remains to glue together the two circles by gluing the back half of the left circle to the front half of the right circle, and gluing the front half of the left circle to the back half of the right circle. However as with the Klein bottle, the gluing of the two circles requires passing part of the surface through itself so that at the seam between the two circles, the surface is intersecting itself. Hence the projective plane only exists in 4-space without self-intersection. When creating compact surfaces with a quotient topology on a polygon allowing for gluing pairs of edges together, we call this a *polygonal representation* of the compact surface.

A diverse collection of compact surfaces can be formed by gluing together other surfaces. This idea is made precise by using the connected-sum operator. Given two surfaces S_1 and S_2, the *connected-sum of* S_1 *and* S_2, denoted $S_1 \# S_2$, is the surface obtained by removing the interior of a disk from each surface and gluing the

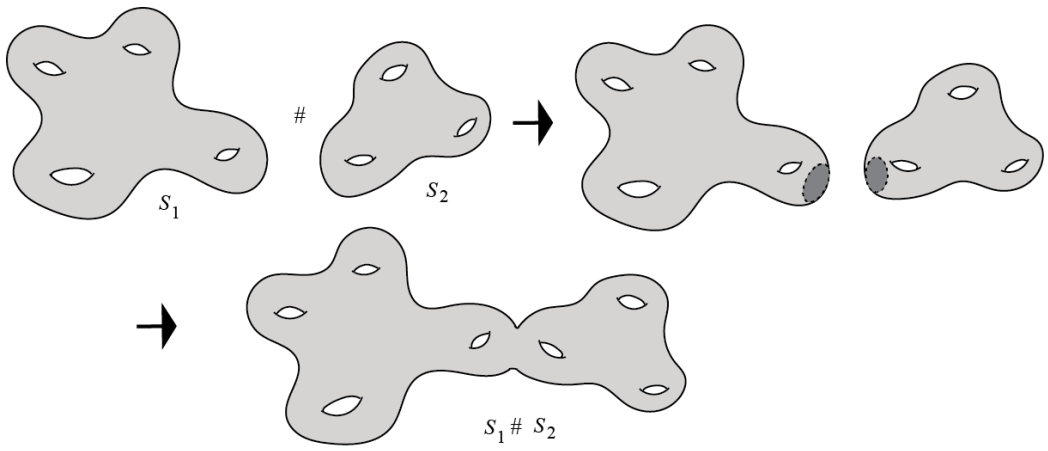

Figure 3.3. The connected-sum of surfaces S_1 and S_2.

two circle boundaries together (see Figure 3.3). The connected-sum of two compact surfaces is a compact surface.

Lemma 3.1.1. *Let S be a compact surface. $S\#S^2 = S$.*

Figure 3.4 illustrates Lemma 3.1.1. Since the connected-sum of a sphere, S^2, with any compact surface, S, is homeomorphic to the original compact surface S, the sphere acts like an identity when thinking of the connected-sum as an additive operator on surfaces. Recall, in Section 1.3 the n-hole torus was introduced, and we were able to glue together n tori to form the n-hole torus by using the connected-sum operator.

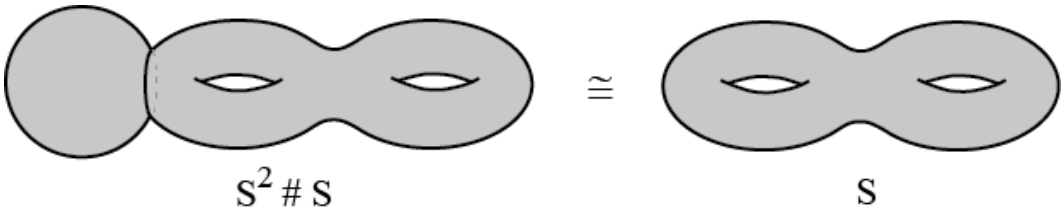

Figure 3.4. The connected-sum of a compact surface with the 2-sphere will be homeomorphic to the original compact surface.

Unfamiliar compact surfaces can be created with the notion of connected-sums, such as the surface $S^2\#P\#T\#T\#K\#P$. However, we will see that these unfamiliar compact surfaces are ultimately homeomorphic to either a sphere, a connected-sum of projective planes, or a connected-sum of tori. Notice that the Klein bottle is not included in the list of possibilities just mentioned. The following Lemma shows that the Klein bottle is the connected-sum of projective planes.

Lemma 3.1.2. *The Klein bottle is homeomorphic to the connected-sum of two projective planes.*

Proof. Begin with two copies of the polygonal representation of the projective plane and we want to perform a connected-sum operation on these two compact surfaces. To do so, we must remove the interior of a disk from each projective plane and we will represent this by the edge 3 shown in Figure 3.5.

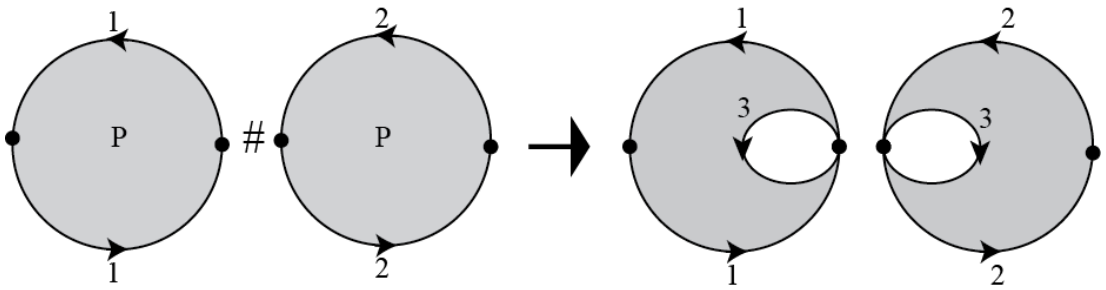

Figure 3.5. The removal of disks from the polygonal representation of projective planes to perform the connected-sum operation.

Once the interior of the disks is removed, we can open up each projective plane on the edge 3 and glue the two projective planes together on edge 3, as shown in Figure 3.6. The result is the polygonal representation of the connected-sum of two projective planes.

To finish, we must cut $P\#P$ on the new edge 4, as shown in Figure 3.7, and then glue the resulting two pieces together on edge 1. After the gluing is completed

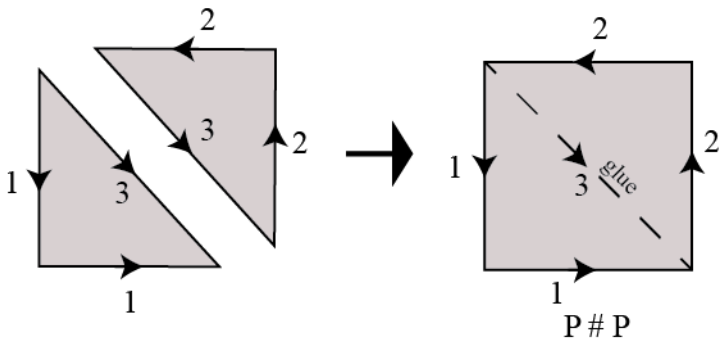

Figure 3.6. A polygonal representation of the connected-sum of two projective planes.

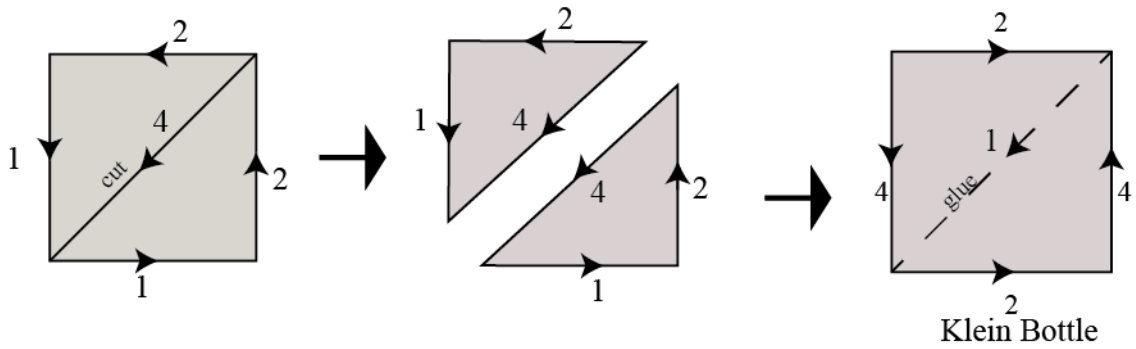

Figure 3.7. Cutting $P\#P$ along the introduced edge 4 and gluing the resulting pieces together on edge 1.

on edge 1, we see that the result is a polygonal representation of the Klein bottle as in Figure 3.1. □

The following lemma allows us to represent a surface that contains a combination of a connected-sum of tori and projective planes as a connected-sum of projective planes.

Lemma 3.1.3. *The connected-sum of a torus with a projective plane is homeomorphic to the connected-sum of three projective planes.*

Proof. Begin with the polygonal representation the torus, T, and the projective plane, P. To construct $T\#P$ we must remove the interior of a disk from both T and

P and then glue the disk boundaries together. In Figure 3.8, the disk boundaries are represented by edge 4.

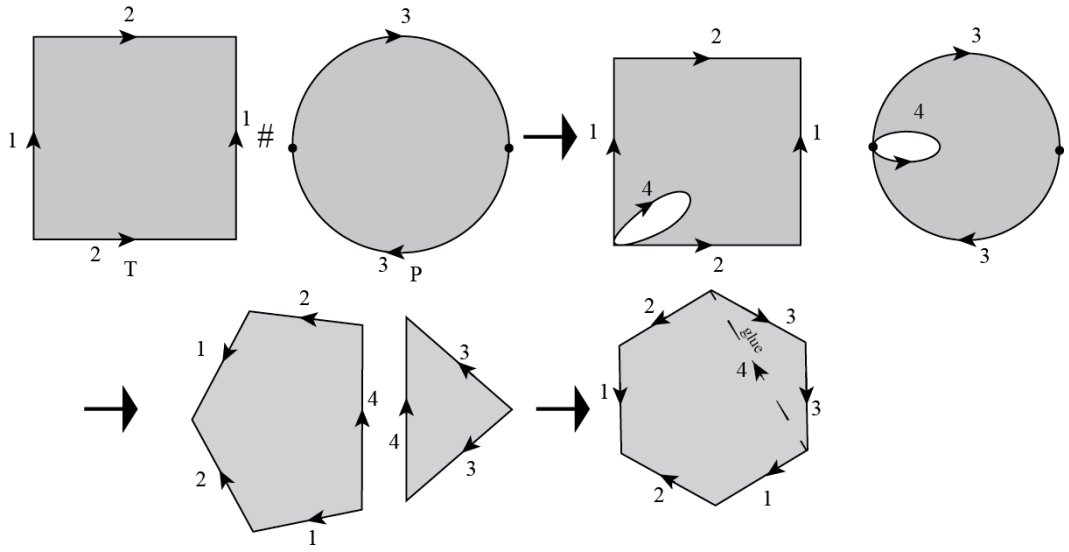

Figure 3.8. Applying the connected-sum operator to the polygonal representations of a torus and projective plane, resulting in the polygonal representation of $T\#P$.

Next, it is necessary to open up the torus and projective plane along the edge 4 and then glue the two pieces together along that edge, as shown in Figure 3.8. The result of this gluing is the polygonal representation of the connected-sum of these two compact surfaces, $T\#P$.

Now, it is necessary to manipulate the polygonal representation of $T\#P$ through

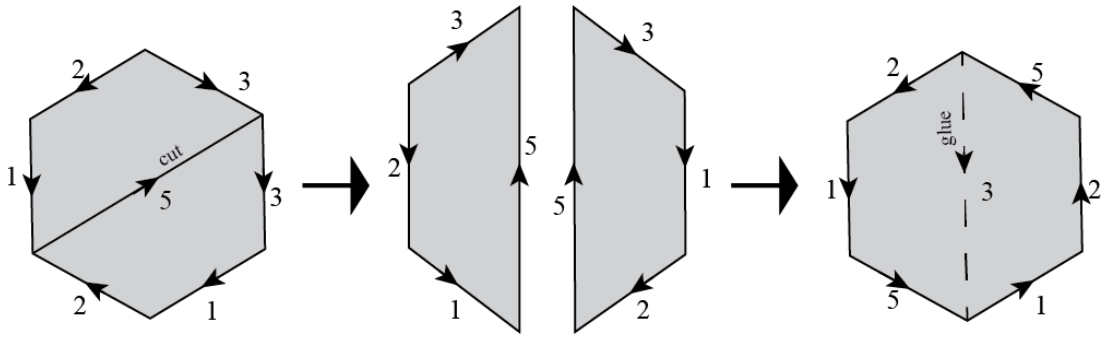

Figure 3.9. Cutting $T\#P$ along a new edge 5 and gluing the resulting pieces together on edge 3.

a series of cutting and gluing operations. First, we will cut along a new edge 5 and this will turn $T\#P$ into two polygons that can be glued together on edge 3.

We will introduce another new edge, labeled 6, cutting apart the polygon into two pieces. The resulting two pieces can be glued together along the edge labeled 1, as shown in Figure 3.10. The polygon that is formed by this gluing represents the connected-sum of a Klein bottle with a projective plane. To see this, the twist pair of edges, edges 6, will form the projective plane. The alternating pairs of edges, 2 and 5, form the Klein bottle because the pair of edges 2 are glued with a twist but the pair of edges 5 are glued straight across, as we had with the construction of the Klein bottle from a square.

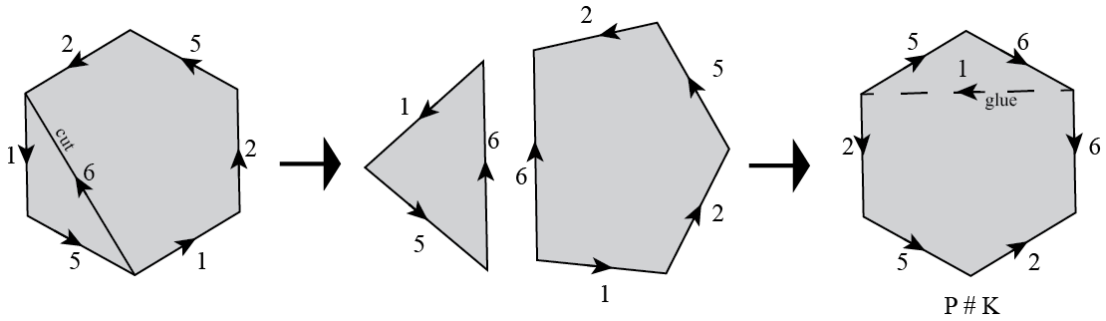

Figure 3.10. Transforming $T\#P$ into $P\#K$ by cutting along a new edge 6 and gluing the pieces together on edge 1.

So, $T\#P = P\#K$. As we saw in Lemma 3.1.2, the Klein bottle is homeomorphic to the connected-sum of two projective planes, and hence $T\#P = P\#K = P\#P\#P$.

□

Lemmas 3.1.2 and 3.1.3, allow us to represent compact surfaces that involve combinations of Klein bottles, tori, projective planes, and spheres as simply the connected-sum of a collection of projective planes as long as there is at least one projective plane or one Klein bottle involved.

Corollary 3.1.4. *Let $m, n, p, r \in \mathbb{Z}^+$. If $m \neq 0$ or $r \neq 0$, then the compact surface $nT\#mP\#pS^2\#rK$ is homeomorphic to $(2n + m + 2r)P$.*

Proof. For any $p \in \mathbb{Z}^+$, the surface $nT\#mP\#pS^2\#rK$ is homeomorphic to $nT\#mP\#rK$ by Lemma 3.1.1. By Lemma 3.1.2, rK is homeomorphic to $(2r)P$. Thus, it suffices only to consider the surface $nT\#(m + 2r)P$. The result will follow by induction on n. Let $n = 1$, we will be considering the surface $T\#(m + 2r)P$. Using Lemma 3.1.3, we see the result holds in the case $n = 1$ as follows:

$$\begin{aligned} T\#(m+2r)P &= (T\#P)\#(m+2r-1)P \\ &= (P\#P\#P)\#(m+2r-1)P \\ &= (2+m+2r)P. \end{aligned}$$

Assume the result holds for $n = k$, so $kT\#mP\#rK\#pS^2 = (2k + m + 2r)P$. Now let $n = k + 1$. The result follows from the induction hypothesis and Lemma 3.1.3.

$$\begin{aligned} (k+1)T\#(m+2r)P &= kT\#T\#P\#(m+2r-1)P \\ &= kT\#3P\#(m+2r-1)P \\ &= kT\#mP\#2P \\ &= (2k+2r+m)P\#2P \\ &= (2(k+1)+m+2r)P \end{aligned}$$

□

The ultimate goal is to create a complete classification of all the possible compact surfaces and this is realized with the following theorem. In the Embedding Algorithm chapter, the Compact Surface Classification Theorem will be important because it allows us to easily identify and distinguish the differnt resulting embedding surface for a graph.

Theorem 3.1.5 (Compact Surface Classification Theorem)**.** *Every compact surface is homeomorphic to exactly one of the following:*

(i) S^2

(ii) $P\#P\#\cdots\#P$ *denoted* nP *for some integer* n*.*

(iii) $T\#T\#\cdots\#T$ *denoted* nT *for some integer* n*.*

The proof of the compact surface classification theorem is completed in two major parts by first identifying all the possible types of compact surfaces and then distinguishing between the possibilities. The distinguishing portion of the proof will be omitted since it is the ideas behind identifying the surfaces that are important to us when we are investigating graph embeddings in the Embedding Algorithm Chapter. The distinguishing part is done using tools of algebraic topology that enable us to conclude, for example, that $4T$ is not homeomorphic to $9T$, that $4T$ is not homeomorphic to $4P$, and that $4P$ is not homeomorphic to $9P$. A proof of the distinguishing part can be found in [6] or [10].

3.2 Polygonal Representations of Compact Surfaces

First it is necessary to understand the polygonal representations of these compact surfaces. To do so, we must introduce triangulations of compact surfaces. Let τ be a triangular region in the plane, and let S be a compact surface. If $f : \tau \to S$ is an embedding, then the image of τ under f is a *triangle in the surface* S (see Figure 3.11). Also, the images of the edges and vertices of τ under f are the edges and vertices, respectively, of the triangle in S. This leads to the definition of a *triangulation* $\mathcal{T}$ *of a compact surface* S, which is a collection of finitely many triangles that cover S such that any two triangles in $\mathcal{T}$ either do not intersect, intersect in a shared vertex, or intersect in a shared edge. A surface with a trianglulation is called a *triangulated surface.* It is desirable to treat every compact surface as a polygon with

pairs of edges glued together in order to identify all the different compact surface possibilities. We discuss below how this can be done via a triangulation.

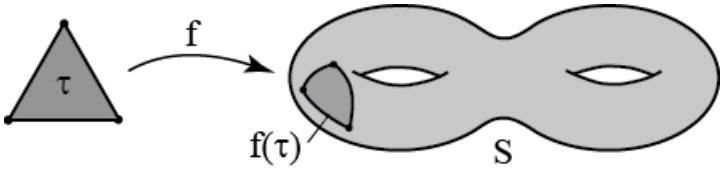

Figure 3.11. The embedding of the triangle τ into the surface.

Theorem 3.2.1 (Radó's Theorem)**.** *Every compact surface is homeomorphic to a triangulated compact surface.*

This result was first proved by Tibor Radó in 1925; the details of the proof are omitted here, but can be found in Radó's original paper [11]. Another proof of this result can be found in [10]. The proof of Radó's Theorem uses techniques from geometric topology but the general ideas behind the proof are discussed as follows. Because a compact surface is locally homeomorphic to $\mathbb{R}^2$ we can cover it with triangles. Since it is compact, we can cover it with finitely many triangles. The result is not necessarily a triangulation because some triangles may not intersect properly. But it is possible to rearrange and subdivide the triangles so that the result is a triangulation of the compact surface.

It follows from Radó's Theorem that every compact surface can be realized as a polygon with pairs of edges identified together via quotient mappings. To see how a triangulated surface leads to the polygonal representation, consider the following argument. Let S be a compact surface with a given triangulation. Begin with one triangle and identify a single edge of that triangle with the one of the edges it shares with another triangle. Glue these triangles together along the edge; the result is a four-sided polygon that is topologically equivalent to a disk. Now, among the remaining triangles in the surface, at least one will glue to a side of the four-sided

polygon's boundary. Take this triangle and glue it to the appropriate side. Again, the result is topologically equivalent to a disk. We continue this process until all

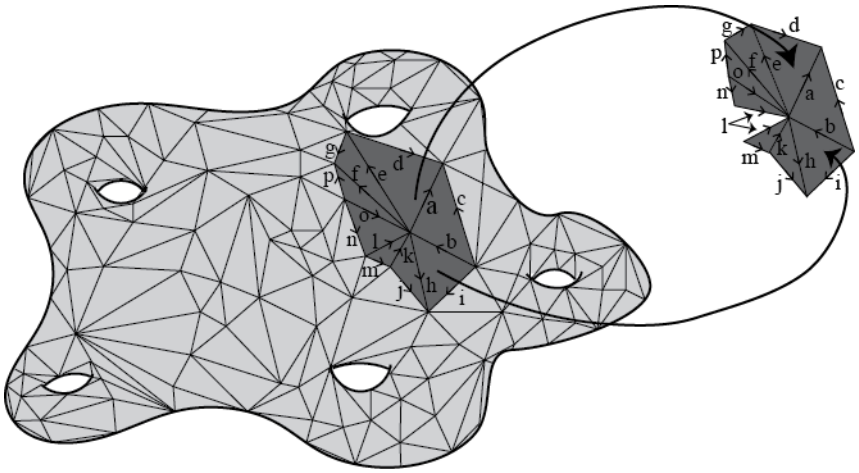

Figure 3.12. The process of constructing a polygonal representation of a surface from a triangulation.

the triangles have been identified together into a polygon homeomorphic to a disk with pairs of edges in the boundary still to be identified together. The result is called the *polygonal representation of the compact surface* S. Figure 3.12 illustrates the process of forming a polygonal representation of a triangulated compact surface, where the collection of dark gray triangles are identified together on shared edges and the result is topologically equivalent to a disk. As shown previously, the polygonal representation of a torus is obtained from a square with opposite edges identified together.

This leads into the definition of the Euler characteristic of a compact surface. Let T be a triangulation of a compact surface S. The Euler characteristic of T is given by $\chi(T) = V - E + F$ where V is the number of vertices, E is the number of edges, and F is the number of triangles in the triangulation. It can be shown that the Euler characteristic of a triangulated surface only depends on the surface,

rather than the chosen triangulation. In particular, the following theorem proves this claim.

Theorem 3.2.2. *Let T_1 and T_2 be triangulations of a compact surface S. Then $\chi(T_1) = \chi(T_2)$.*

A detailed proof of this result can be found in [1]. The idea behind the proof is as follows. First, given a triangulation of a compact surface, a homeomorphic triangulation will have the same Euler Characteristic. Also, given a triangulation, subdividing the triangles does not change the Euler Characteristic. Furthermore, given two triangulations of a compact surface, there exists subdivisions of each triangulation that are homeomorphic. Hence, the Euler Characteristic of a surface is independent of the choice of triangulation.

Thus, we can define the Euler characteristic of a surface S to be $\chi(S) = V - E + F$ for any triangulation of S. We will also adjust the definition of F to the number of faces where a triangulated surface can be thought of as a collection of triangular faces. Examples of triangulations of the sphere, torus, Klein bottle, and projective plane are shown in Figure 3.13. We can use these triangulations to calculate the Euler characteristic for each of these surfaces. We'll go through the details of calculating the Euler characteristic for the sphere; on the triangulation shown in Figure 3.13 there are 18 triangles, 13 vertices, and 29 edges. Now we will apply the formula, $\chi(S^2) = V - E + F = 13 - 29 + 18 = 2$. Similarly, from the other triangulations, we see that $\chi(T) = 0$, $\chi(K) = 0$, and $\chi(P) = 1$.

The connected-sum operator allows us to construct interesting compact surfaces, and the following theorem indicates how we can calculate the Euler characteristic of surfaces constructed in this way.

Theorem 3.2.3. *If S_1 and S_2 are compact surfaces, then $\chi(S_1 \# S_2) = \chi(S_1) + \chi(S_2) - 2$.*

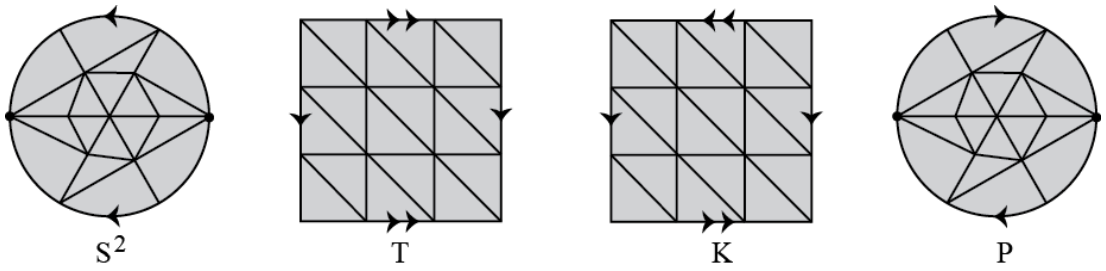

Figure 3.13. Triangulations of the sphere, torus, Klein bottle, and projective plane.

Proof. Since the Euler characteristic is independent of the triangulation, we can choose triangulations of S_1 and S_2 such that the disks whose interiors will be removed under the connected-sum operation correspond to triangles in the triangulations. The connected-sum operation corresponds to removing the interior of a triangle from each surface and identifying the boundaries of the triangles together, gluing vertices to vertices and edges to edges. Notice, each triangle has three edges, three vertices, and one face. When we glue the two triangles together, three pairs of vertices and three pairs of edges will be identified, and two faces will be removed. When $\chi(S_1\#S_2)$ is compared to $\chi(S_1) + \chi(S_2)$, we see that the resulting surface, $S_1\#S_2$, will have three fewer edges, three fewer vertices, and two fewer faces than the original two surfaces. Therefore, the resulting Euler characteristic of $S_1\#S_2$ is given by the following:

$$\begin{aligned}\chi(S_1\#S_2) &= \chi(S_1) + \chi(S_2) - 3 + 3 - 2 \\ &= \chi(S_1) + \chi(S_2) - 2.\end{aligned}$$

□

The following result is a direct consequence of Theorem 3.2.3.

Corollary 3.2.4. *Let $n \in \mathbb{Z}$. For the compact surfaces nT and nP, the Euler characteristic is given by $\chi(nT) = 2 - 2n$ and $\chi(nP) = 2 - n$.*

Notice, $V - E + F$ appears in both the Euler characteristic formula for compact surfaces and Euler's formula for planar graphs. In the following chapter we will see how these two formulas are useful for determining the surface into which a graph can be embedded. As seen in the proofs of Lemmas 3.1.1-3.1.3, it is advantageous to manipulate these compact surfaces as polygons with pairs of edges glued together in order to identify all the different possible compact surfaces. In the next chapter when embeddings of graphs onto compact surfaces are discussed, we will use the polygonal representations of the compact surfaces considered.

3.3 Identification and Classification of Compact Surfaces

In this section we will complete the first part of the proof of the Compact Surface Classification Theorem (Theorem 3.1.5) by identifying all of the different compact surfaces that can be created when gluing pairs of edges in a polygon. We have discussed previously how every compact surface can be represented as a n-gon with pairs of edges glued together. Because we are gluing pairs of edges in the n-gon, it can be assumed that n is even. In those representations there are two different types of pairs of edges being glued. The first type is a *straight pair of edges* (see Figure

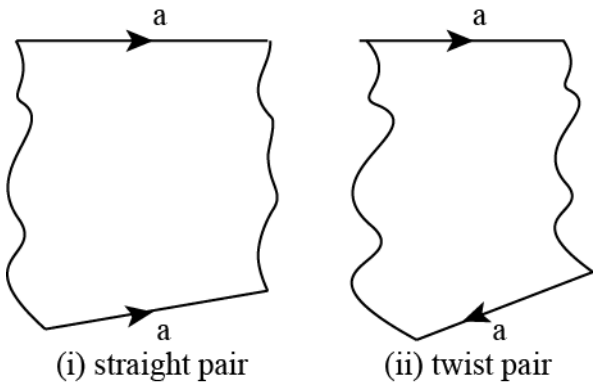

Figure 3.14. The two types of pairs of edges being glued within a $2n$-gon.

3.14.i) where the edges are glued straight across and the second type is a *twist pair*

of edges (see Figure 3.14.ii) where the gluing creates a twist in the surface. Two pairs of edges a and b are said to be *alternating pairs of edges* if they are arranged as shown in Figure 3.15. The alternating pairs can be of both twist pairs, both straight pairs, or a straight and a twist pair of edges. We will see below that an alternating pair of straight-edge pairs will contribute a torus in the connected-sum of the resulting surface while a single twist pair of edges will contribute a projective plane.

Let the distance between edges a, denoted D_a, be the minimum number of edges between the pair of a edges.

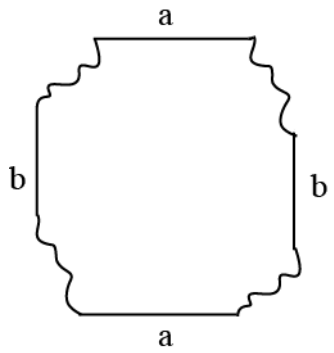

Figure 3.15. Alternating pairs of straight edges.

The following theorem identifies the different possible compact surfaces that can be constructed by gluing pairs of edges together of a $2n$-gon. However, it is important to note that this theorem does not distinguish between these compact surfaces.

Theorem 3.3.1 (The Identification of Compact Surfaces Theorem)**.** *If S is a $2n$-gon with edges glued together in pairs, then S represents either a sphere, a connected-sum of projective planes, or a connected-sum of tori.*

Before beginning the proof of the identification of compact surfaces, we need to introduce a technical lemma.

Lemma 3.3.2. *Let S be a $2n$-gon with pairs of edges glued, and assume that S contains no alternating pairs of edges. Then in S there is a pair of edges, i, for which $D_i = 0$.*

Proof. The proof will follow by induction on n. Let $n = 1$. Then S is a 2-gon with the edges, i, glued as a pair. Clearly, as shown in Figure 3.16, the distance between

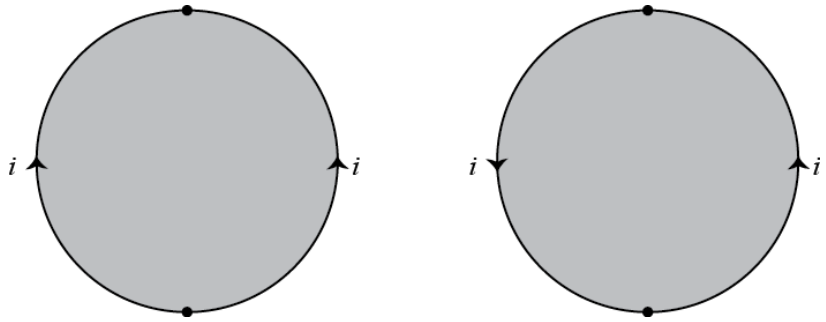

Figure 3.16. The two 2-gons with a pair of edges such that $D_i = 0$.

edges i is zero. Now assume the result holds for $n = k$. Let S^* be a $2(k+1)$-gon containing no alternating pairs. Remove a pair of edges j to obtain a $2k$-gon S. The surface S has no alternating pairs, so there exists a pair of edges i such that $D_i = 0$.

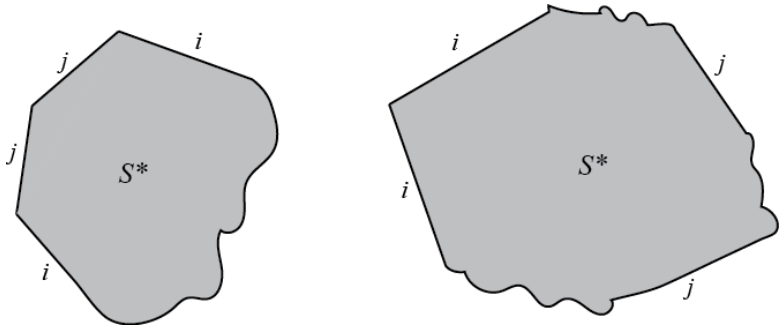

Figure 3.17. The two cases of S^* when obtained by adding the j pair of edges back into S.

Replace the removed edges, j, to obtain S^*. Because there are no alternating pairs of edges in S^*, there are two possible locations for the pair of edges j. As

shown in Figure 3.17, either both j edges are inserted directly between the i edges resulting in $D_i = 2$ and $D_j = 0$ or the pair j are inserted such that $D_i = 0$. □

Now we have the necessary tools to proceed with the proof of the Identification of Compact Surfaces Theorem. The proof will follow by induction on n, where the compact surface is represented as a 2n-gon with pairs of edges glued.

Proof. To begin, we will consider the base cases of $n = 1$ and $n = 2$.For the case where $n = 1$ there are two different possibilities for the resulting compact surface, a 2-sphere or a projective plane (see Figure 3.18).

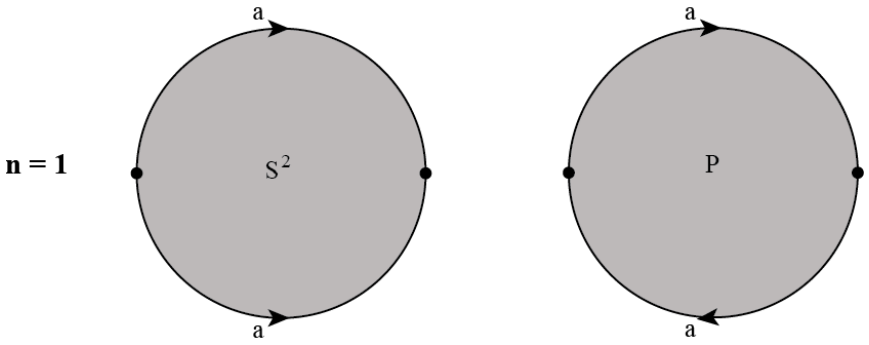

Figure 3.18. The two compact surfaces that result from a 2-gon with the pair of edges identified together.

Consider the case where $n = 2$. In Figure 3.19, we demonstrate that the result holds. In other words, a 4-gon with pairs of edges glued constructs either a torus, a connected-sum of projective planes, or a 2-sphere.

Assume the result holds for all $2n$-gons with edges glued in pairs where $n \in \{1, 2, \ldots, k\}$. Let S^* be a $2(k+1)$-gon with edges glued in pairs.

First consider the case where there is a twist pair of edges, a. If $D_a = 0$ in S^* then the pair of a edges identifies to a projective plane. To see this, recall the construction of the projective plane from the square. In the second step, after the two pairs of edges are identified to be a single pair of edges, the result was a circle

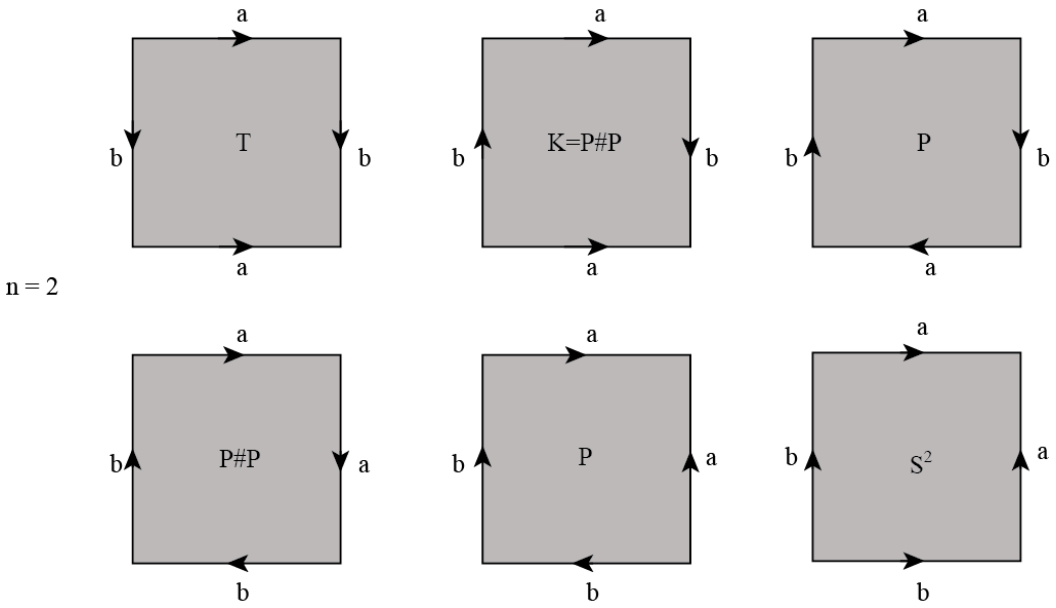

Figure 3.19. The six cases that arise from a 4-gon with pairs of edges glued together.

formed by two twist pair of edges glued together. Therefore S^* is $M\#P$, where M is a $2k$-gon with pairs of edges glued and P is a projective plane (see Figure 3.20).

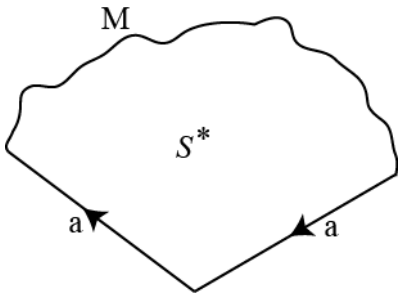

Figure 3.20. The case where S^* contains a twist pair of edges, a, such that $D_a = 0$.

By the induction hypothesis and Corollary 3.1.4, S^* is homeomorphic to one of the following:

(i) $S^2\#P = P$,

(ii) $rP\#P = (r+1)P$ for some integer r, or

(iii) $rT\#P = (2r+1)P$ for some integer r.

Now assume $D_a \neq 0$ in S^* (see Figure 3.21). Let b be an edge on the interior of S^* going from the left side of the edge a in the top half of S^* to the right side of the second edge a in the bottom half of S^*, as shown in Figure 3.21. If we cut along the edge b and glue along the edge a, we see that $D_b = 0$. As in the previous

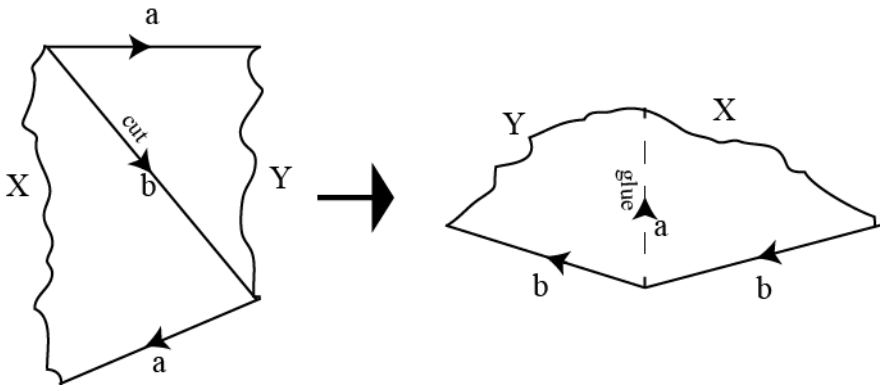

Figure 3.21. The case where S^* contains a twist pair of edges, a, such that $D_a \neq 0$.

case, $S^* = M\#P$, where M is a $2n$-gon with pairs of edges glued. As argued above, $S^* = mP$ for some integer m. Thus assuming the result holds for $2n$-gons with $n = 1, \ldots, k$, it holds in the case where S^* is a $2(k+1)$-gon containing a twist pair of edges.

Next consider the case where there are no twist pairs of edges in S^*. By Lemma 3.3.2, we know that S^* has either an alternating pair of edges or a pair of edges with distance zero. Assume there is a pair of edges, i, in S^* such that $D_i = 0$. These edges must be a straight pair since we are assuming there is no twist pair (see Figure 3.22). Thus $S^* = M\#S^2$, where M is a $2k$-gon. By Lemma 3.1.1 $S^* = M$ and by the inductive hypothesis it follows that S^* is homeomorphic to either S^2, rT, or rP for some integer r.

Now assume there is an alternating pair of edges in S^*. Call the edges in the alternating pairs a and b (see Figure 3.23). To begin, we will make a cut along an edge c that runs from the bottom of edge b across S^* to the bottom of the other

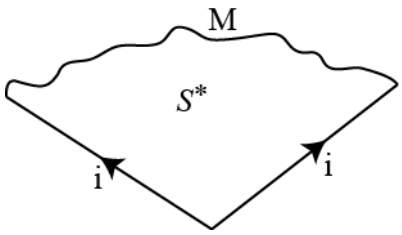

Figure 3.22. The case where compact surface S^* contains no alternating pairs and no twist pairs.

b edge, as shown in Figure 3.23. Once we cut along edge c, we will glue the two pieces along edge a. One more cutting and pasting is necessary, along a new edge d which runs from the top left side of edge c and once again down across S^* to the right bottom edge of c, and we will glue the resulting two pieces together along edge b. Here we see that after the cutting and pasting, S^* becomes $M\#T$ where the

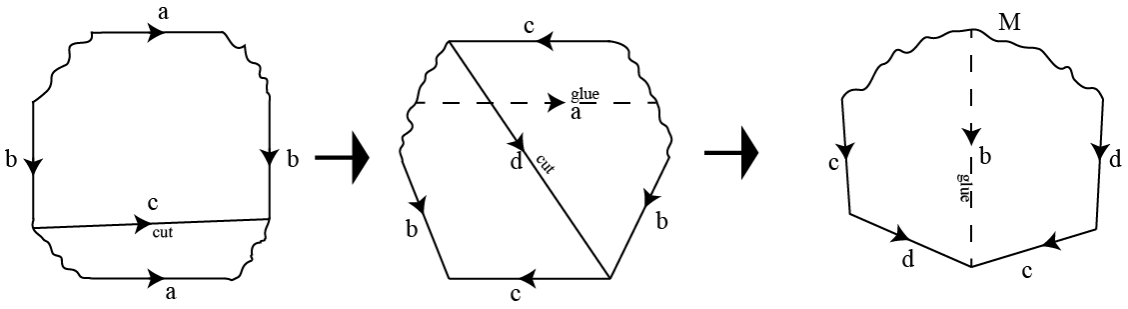

Figure 3.23. The case where S^* contains alternating pairs of straight edges and no twist pairs.

torus is formed by the introduced alternating pair of edges c and d. We know that the pairs of edges c and d form a torus since they are straight alternating pairs of edges. By the inductive hypothesis, Lemma 3.1.1, and Lemma 3.1.4 the surface S^* is homeomorphic to one of the following:

(i) $S^2\#T = T$,

(ii) $rT\#T = (r+1)T$ for some integer r, or

(iii) $rP\#T = (r+2)P$.

Assuming that the result holds for $2n$-gons with $n = 1, \ldots k$, it holds in the case where S^* is a $2(k+1)$-gon containing no twist pairs of edges. Now by induction it follows that the result holds for all $2n$-gons, $n \in \mathbb{Z}_+$. □

This proof completes the identification of all the different compact surfaces that can be constructed by gluing pairs of edges in a $2n$-gon. As mentioned previously to finish the classification of compact surfaces we need to distinguish between these compact surfaces. It can be shown using some techniques from algebraic topology that $nT \neq mT$ whenever $n \neq m$, $nP \neq mP$ whenever $n \neq m$, $nT \neq mP$, $nT \neq S^2$, and $nP \neq S^2$ for integers m and n. A proof of the remaining portion of the Compact Surface Classification Theorem can be found in [6], [10], or [1].

3.3.1 Compact Surface Identification Codes

Given a compact surface represented as a $2n$-gon with edges glued in pairs, we can use the method of cutting and gluing pairs of edges to identify the surface. However, we can define compact surface identification codes that represent the cutting and gluing used in the previous section. These compact surface identification codes can be applied to the collection of $2n$-edges. The collection of $2n$-edges records the clockwise ordering of edges about the $2n$-gon. If an edge is oriented in the counter-clockwise direction it will be denoted with an inverse. Consider a surface M with a straight pair of edges a with $D_a = 0$. In the collection of $2n$-edges, this pair of edges will appear as aa^{-1} and will identify to a sphere in a connected-sum representing the surface.

For instance, consider a surface M that contains a twist pair of edges as shown in Figure 3.24. The twist pair of edges, a, will identify to a projective plane when the distance between the two edges is zero. In Figure 3.24, we will represent the left picture of M as a collection of edges recorded in the clockwise direction, $aXaY$,

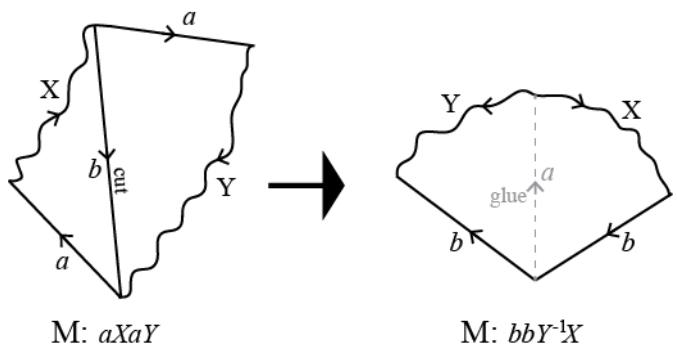

Figure 3.24. The cutting and gluing of surface M that results in the projective plane identification code.

where X and Y are collections of arbitrary edges glued in pairs. We introduce the edge b and cut along it, splitting M into two pieces. We will glue these two pieces together on edges a and the result is a surface homeomorphic to M with a twist pair of edges b such that $D_b = 0$. Now $M = bbY^{-1}X$ where the pair of edges b identifies as a projective plane and Y^{-1} is given by the edges in Y in the opposite order. In detail, if $Y = a_1 a_2 \cdots a_n$, then $Y^{-1} = a_n^{-1} a_{n-1}^{-1} \cdots a_1^{-1}$. Thus, the projective plane identification code is given by

$$M = aXaY \xrightarrow{P \quad I.D.} bbY^{-1}X.$$

In other words, given a compact surface that contains a twist pair of edges a with $D_a \neq 0$, we can apply the projective plane identification code to pull off the projective plane produced by the twist pair of edges.

Now, consider a surface M with alternating pairs of straight edges labeled a and b in Figure 3.25. To derive the torus identification code, let $M = aWbXa^{-1}Yb^{-1}Z$. Then we will introduce an edge c and cut along this edge, as shown in Figure 3.25. The two pieces that result from cutting on edge c will be glued along edges a. At this step, M is homeomorphic to $c^{-1}XWbcb^{-1}ZY$. Next another edge will be introduced and cut along, call this edge d. Once we cut along edge d we will glue the

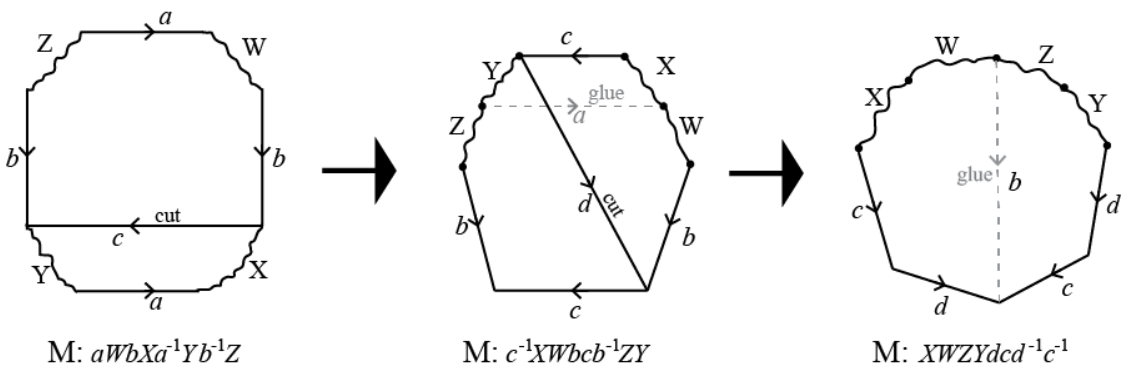

Figure 3.25. The cutting and gluing of surface M that results in the torus identification code.

resulting two pieces together along edge b. The result is a surface homeomorphic to M given by $XWZYdcd^{-1}c^{-1}$ where $dcd^{-1}c^{-1}$ identifies to a tours. Therefore, the torus identification code is given by

$$M = aWbXa^{-1}Yb^{-1}Z \xrightarrow{T \quad I.D.} XWZYdcd^{-1}c^{-1}.$$

Applying the torus identification code to a collection of $2n$-edges containing alternating pairs of straight edges will pull out a single torus from the $2n$-edges.

For example, consider the surface given by $M = abcda^{-1}bc^{-1}defghef^{-1}gh^{-1}$. We will use the compact surface identification codes to identify the surface M as either a connected-sum of tori, a connected-sum of projective planes, or a sphere. First we will apply the torus identification code. In the example below, we will indicate with brackets the edges to which the compact surface identification code is being applied. In particular, the code being applied to the bracketed edges is determined by the label on the arrow the line below the bracketed edges.

$$M = abcda^{-1}bc^{-1}de\underbrace{f}\,g\underbrace{h}\,e\underbrace{f^{-1}}\,g\underbrace{h^{-1}}$$

Here, $W = g$, $X = e$, $Y = g$, and $Z = abcda^{-1}bc^{-1}de$. Now, we can apply the torus identification code to M.

$$\xrightarrow{1T \quad I.D.} egabcda^{-1}bc^{-1}degjij^{-1}i^{-1}$$

The new edges $jij^{-1}i^{-1}$ created by applying the torus identification code form a torus. We can pull these edges identified as a torus to the end of the collection of edges and rewrite M as follows:

$$M = egabcda^{-1}bc^{-1}deg\#T.$$

Now, we will apply another torus identification code to the edges labeled with brackets in the representation of M below.

$$M = eg\underbrace{a}b\underbrace{c}d\underbrace{a^{-1}}b\underbrace{c^{-1}}deg\#T.$$

Next, we will apply the projective plane identification code to edges d, where $X = b$ and $Y = egegb$.

$$\xrightarrow{1T \quad I.D.} \underbrace{d}b\underbrace{d}egegb\#T\#T$$

Notice, in the collection of edges yet to be identified, the edges $b^{-1}b$ will identify to be a sphere.

$$\xrightarrow{1P \quad I.D.} \underbrace{b^{-1}}g^{-1}e^{-1}g^{-1}e^{-1}\underbrace{b}\#T\#T\#P$$

Finally, we will apply the projective plane identification code and then the sphere identification code.

$$\xrightarrow{1S^2 \quad I.D.} \underbrace{g^{-1}}e^{-1}\underbrace{g^{-1}}e^{-1}\#T\#T\#P\#S^2$$

$$\xrightarrow{1P \quad I.D.} \underbrace{ee^{-1}}\#T\#T\#P\#P \xrightarrow{1S^2 \quad I.D.} \#T\#T\#P\#P = 2T\#2P$$

Therefore, by Lemma 3.1.4 the compact surface $M = 2T\#2P = 6P$. In the following chapters, when identifying compact surfaces using the method of cutting and gluing, we will apply the compact surface identification codes as described above.

Now that the proof of the Compact Surface Classification Theorem has been discussed, we can begin constructing an embedding algorithm using the rotation systems of a graph to determine surfaces into which the graph can be embedded.

Chapter 4
EMBEDDING ALGORITHM

In this chapter an embedding algorithm will be developed that uses a rotation system of a graph as input and will determine the surface into which the graph and corresponding rotation system can be embedded. The algorithm discussed here is analogous to the *Heffter-Edmonds Algorithm* and has its origins in the 1890's with Heffter's work [8]. In 1960 Jack Edmonds, unaware of Heffter's work, invented a parallel algorithm using a vertex form of a graph for producing an embedding [5]. Today the algorithm is attributed to both Heffter and Edmonds. The algorithm discussed below takes a direct approach to the derivation by using the compact surface identification codes and constructing the actual embedding of the graph, rather than simply guaranteeing existence.

4.1 2-Cellular Embedding Algorithm

Let $G = \{V, E\}$ be a connected simple graph with vertex set $V = \{i : i = 1, 2, 3, \ldots, n\}$ and edge set $E = \{e_{i,j} : \text{for some } i, j \in V\}$ containing M edges. The edge $e_{i,j}$ indicates a directed edge that is incident to vertices i and j and when the edge $e_{i,j}$ is traversed in a walk one starts at vertex i and travels to vertex j. The edge denoted $e_{j,i}$ will represent the same edge as $e_{i,j}$ only the direction is reversed. Suppose G has an associated rotation system $\Pi = \{\pi_i : i \in V\}$ such that each π_i records a particular counter-clockwise cyclic permutation of the edges incident to vertex i. We call π_i the vertex code for vertex i. Denote the graph G with rotation system Π as Π_G. Each vertex code will be of the form:

$$\pi_i : (e_{i,m_1}; e_{i,m_2}; e_{i,m_3}; e_{i,m_4} \ldots e_{i,m_j})$$

for each $i = 1, 2, 3, \ldots, n$ (see Figure 4.1). The order in which the edges appear in the vertex code, π_i, gives an ordering of edges about vertex i in G that we will realize in an embedding of G in a compact surface. Each edge will appear exactly twice, once in each direction, in a rotation system described in this way.

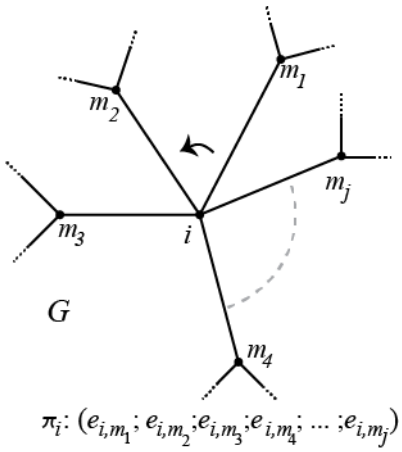

Figure 4.1. The vertex code, π_i, of a particular rotation system of a graph G that records a counterclockwise cyclic ordering about the vertex i

Recall, given a graph embedded into a surface S, if one takes the complement of the embedded graph in S then the result is a collection of components called faces. Further, the boundary of each face is a collection of edges of the embedded graph.

Given a rotation system, in order to determine the embedding of the graph associated to the rotation system we construct what are known as boundary walks. Each boundary walk is a cycle of directed edges in the graph and corresponds to a clockwise walk around the boundary of a face of the embedding. We show below how, by identifying the faces and how to glue them together, the boundary walks determine a polygonal representation of the surface into which the graph is embedded.

Each boundary walk will be constructed as follows:

Let $\mathcal{W}$ denote a closed walk in the graph G starting from vertex $w(1)$ on the first edge appearing in $\pi_{w(1)}$, call it $e_{w(1),w(2)}$. To determine the next edge in the walk, locate the edge $e_{w(2),w(1)}$ in the vertex code $\pi_{w(2)}$. Because we are walking clockwise around

a face of the embedding, the next edge in the walk will be the edge immediately following $e_{w(2),w(1)}$ in the vertex code $\pi_{w(2)}$. We denote that edge by $e_{w(2),w(3)}$, as shown in Figure 4.2. Continue constructing the boundary walk $\mathcal{W}$ in this way until a directed edge repeats. Note, this must occur at some point since there are finitely many edges.

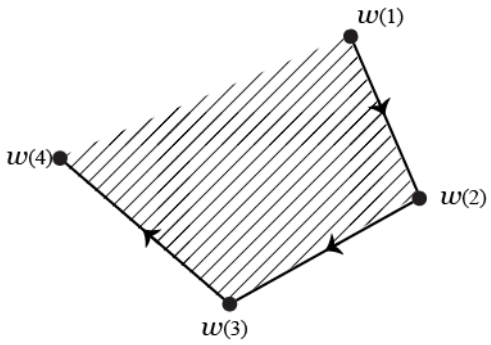

Figure 4.2. The construction of a boundary walk, clockwise around a face of the embedded graph.

We claim that the first directed edge to repeat is the first edge in the boundary walk. Notice that each directed edge occurs exactly once among all the vertex codes in a rotation system. Suppose $\mathcal{W}^*$ is a boundary walk with a repeated directed edge. However, in this boundary walk $\mathcal{W}^*$, the edge preceding the repeated edge must be the edge preceding it in the vertex code. Therefore, if the repeated edge in $\mathcal{W}^*$ is an edge other than the first one, we know the edge preceding it must also repeat. Hence, the first edge must be the first edge that repeats in a boundary walk. Thus the boundary walk $\mathcal{W}$ will be of the form

$$\mathcal{W} = (e_{w(1),w(2)}e_{w(2),w(3)}\dots e_{w(p-1),w(p)}e_{w(p),w(1)}).$$

Continue constructing boundary walks from the rotation system until each directed edge appears in one. The boundary walks are determined by the graph and rotation system and provide us a way to construct a surface and embedding of the

graph in the surface. Let $\mathscr{W}_\Pi$ denote the collection of boundary walks formed from rotation system Π_G. The number of edges in the collection of boundary walks will be twice the number of edges in the graph. The directed edges in the boundary walks bound the faces of the graph embedded in the surface. In order to determine the surface into which the graph G with rotation system Π_G can be embedded, it is necessary to glue together the collection of boundary walks. In other words, because each face is a polygon with edges determined by the graph and rotation system, if one identifies all of the edges together, you will effectively construct the surface with the graph embedding.

For example, let $\mathcal{W}_1$, $\mathcal{W}_2 \in \mathscr{W}_\Pi$ such that

$$\mathcal{W}_1 = (e_{w_1(1),w_1(2)}e_{w_1(2),w_1(3)} \cdots e_{w_1(p-1),w_1(p)}e_{w_1(p),w_1(1)}) \text{ and}$$
$$\mathcal{W}_2 = (e_{w_2(1),w_2(2)}e_{w_2(2),w_2(3)} \cdots e_{w_2(m-1),w_2(m)}e_{w_2(m),w_2(1)}).$$

An illustration of the boundary walks can be seen in Figure 4.3.

Because we would like to glue these two boundary walks together, we will assume

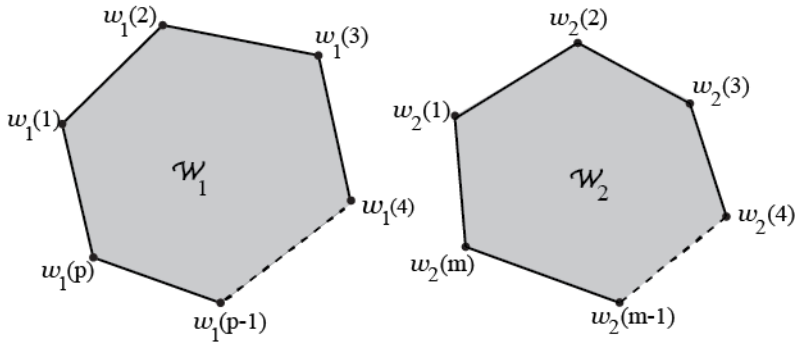

Figure 4.3. Boundary walks, $\mathcal{W}_1$ and $\mathcal{W}_2$, of a graph G with rotation system Π.

the edges $e_{w_2(1),w_2(2)}$ and $e_{w_1(p),w_1(1)}$ represent the same edge, in opposite directions. We will relabel these edges so that $w(1) := w_2(1) = w_1(1)$ and $w(p) := w_2(2) = w_1(p)$. The boundary walks $\mathcal{W}_1$ and $\mathcal{W}_2$ with the relabeled edges are shown as follows

$$\mathcal{W}_1 = (e_{w(1),w_1(2)}e_{w_1(2),w_1(3)}\cdots e_{w_1(p-1),w(p)}e_{w(p),w(1)})$$
$$\mathcal{W}_2 = (e_{w(1),w(p)}e_{w(p),w_2(3)}\cdots e_{w_2(m-1),w_2(m)}e_{w_2(m),w(1)}).$$

Below we discuss how we can glue generally boundary walks to obtain a single polygon with edges glued in pairs.

A boundary walk can start at any of the edges within it. For instance, suppose we are given two boundary walks $\mathcal{W}$ and $\mathcal{W}'$, formed from the same rotation system, that contain the same collection of directed edges but start on different directed edges. The two boundary walks, $\mathcal{W}$ and $\mathcal{W}'$, are cyclic permutations of each other. However, given a rotation system that produces more than one boundary walk, we need to glue together the boundary walks to determine the surface. Let $\mathcal{W}_1$ and $\mathcal{W}_2$ be boundary walks of G_Π such that $\mathcal{W}_1$ contains an edge $e_{i,j}$ and $\mathcal{W}_2$ contains the edge $e_{j,i}$. We can glue together the two boundary walks on this edge, and the gluing results in a single polygon obtained by gluing the two polygons along the corresponding edges. If the shared edges are not at the beginning or end of $\mathcal{W}_1$ or $\mathcal{W}_2$, then we can take cyclic permutations of these boundary walks so that the edge $e_{i,j}$ is at the end of $\mathcal{W}_1$ and the edge $e_{j,i}$ is at the beginning of $\mathcal{W}_2$. After this is accomplished, the two boundary walks can be glued together forming the polygon denoted by $\mathcal{W}_1 \circ \mathcal{W}_2$. Whenever a collection of boundary walks must be glued together to construct a surface with an embedded graph, we will use this gluing method.

Using the boundary walk gluing method, we can glue the boundary walks together along the beginning and ending directed edges. Now consider $\mathcal{W}_1$ and $\mathcal{W}_2$, as given above. Notice that $\mathcal{W}_2$ starts with the relabeled edge $e_{w(1),w(p)}$ and that $\mathcal{W}_1$ ends with the relabeled edge $e_{w(p),w(1)}$. This allows for the two boundary walks to be glued together on these directed edges, as shown in Figure 4.4. The resulting

polygon is given by the following:

$$\mathcal{W}_1 \circ \mathcal{W}_2 = (e_{w(1),w_1(2)} e_{w_1(2),w_1(3)} \cdots e_{w_1(p-1),w(p)} \\ e_{w(p),w_2(3)} \cdots e_{w_2(m-1),w_2(m)} e_{w_2(m),w(1)})$$

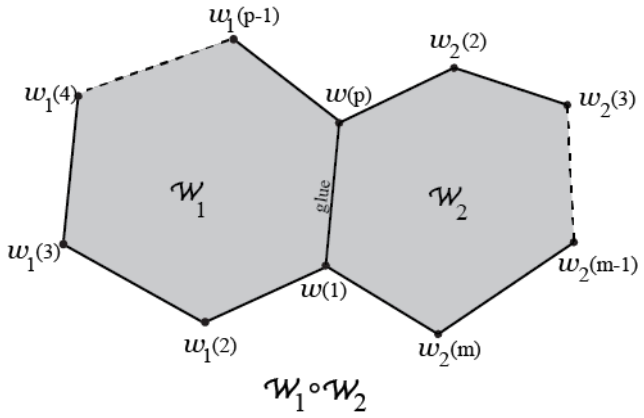

Figure 4.4. The boundary walks $\mathcal{W}_1$ and $\mathcal{W}_2$ glued together on a shared edge.

The edges $e_{w(1),w(p)}$ and $e_{w(p),w(1)}$ are not included in the collection of edges, $\mathcal{W}_1 \circ \mathcal{W}_2$, but when realizing the graph embedding in the surface, these edges will still be present as an edge of the embedded graph. In particular, when the edges $e_{w(1),w(p)}$ and $e_{w(p),w(1)}$ are identified together in $\mathcal{W}_1 \circ \mathcal{W}_2$, the two edges will form an edge of the graph embedded in the surface. Thus continuing with $\mathcal{W}_3$, $\mathcal{W}_4$ etc., the boundary walks of Π_G can be glued together to form a polygon with pairs of edges identified together.

We claim, that all of the boundary walks in $\mathscr{W}$ can be glued together to form one polygon with edges glued in pairs. To see this, suppose we cannot glue every boundary walk together. Let $\mathscr{W}^*$ be a maximal collection of boundary walks in $\mathscr{W}$ that can be glued together. Notice, this collection of boundary walks $\mathscr{W}^*$ represents a polygon with edges glued in pairs. Then there must there exist at least one boundary walk $\mathcal{W}' \in \mathscr{W}$ such that $\mathcal{W}'$ is not contained in $\mathscr{W}^*$ and further that it does not share any edges with the polygon $\mathscr{W}^*$. If such a boundary walk exists this

implies that the resulting space into which the graph can be embedded with a given rotation system must be disconnected. However, recall every graph, G, considered here is simple and connected. Then the existence of a boundary walk $\mathcal{W}'$ implies that G, a connected topological space, is embedded into a disconnected topological space, with edges of the graph in each component of the disconnected space and this is a contradiction. Therefore, no such boundary walk $\mathcal{W}'$ exists and every boundary walk in $\mathscr{W}$ can be glued together forming one polygon. We know that this polygon with edges glued in pairs will identify to a compact surface in which the graph G with rotation system Π can be embedded. From the polygon with edges identified in pairs, we can construct the surface and the actual embedding of the graph on the surface because the edges in the boundary walks prescribe the embedding of the graph in the corresponding surface.

There are a few ways in which we can identify the surface produced by the embedding algorithm. First, as previously described, we can go directly from the gluing of boundary walks into a single polygon with pairs identified together to a surface with an actual embedding of the graph. Second, we could use the Compact Surface Identification Codes presented in Chapter 3. A disadvantage of using the second method is that we lose the result of knowing what the embedding of the graph in the surface looks like and instead just recover the surface that is constructed rather than the surface with the corresponding embedding. We will see a third method for identifying the surface in Remarks section of this chapter.

To identify the surface produced by the embedding algorithm, it is first necessary define how the pairs of edges are glued. Let $e_{i,j}$ be a directed edge in the polygon. Then the edge $e_{j,i}$, in the polygon, will represent the same edge as $e_{i,j}$, only in the opposite direction. Thus, the edges $e_{i,j}$ and $e_{j,i}$ correspond to a straight pair of edges in the polygon (see Figure 3.14). Then the sphere, torus, and projective

plane identification codes can be applied to the polygon with pairs of edges glued to identify the surface.

In fact, the embedding algorithm will always result in a surface homeomorphic to either a sphere or a connected-sum of tori. To see this, recall one of the fundamental compact surfaces, the projective plane. Refer to Figure 3.2 and notice in the second picture that to create the projective plane we are identifying a twist pair of edges together. The twist made in the identification of the edges together is what causes the self-intersection characterizing the projective plane. Now, consider the polygon glued in pairs produced by the embedding algorithm. Recall, for any rotation system Π of a graph G each directed edge of G is contained exactly once in the rotation system. Specifically, the boundary walks constructed via the embedding algorithm contain each directed edge, $e_{i,j}$ and $e_{j,i}$.

Suppose the resulting polygon with edges glued in pairs contains a pair of twist edges. Then this corresponds to some edge $e_{k,l}$ that appears twice in the same direction. This means that the directed edge $e_{l,k}$ will not be contained in any boundary walk since the same edge in the opposite direction, $e_{k,l}$, is repeated twice. However, it is impossible for there to be a twist pair of edges in any boundary walk constructed via the embedding algorithm because the rotation systems are defined to contain each directed edge exactly once. This implies that every pair of edges in the $2M$-gon are glued in straight pairs therefore the algorithm will always produce a surface homeomorphic to a sphere or a connected-sum of tori.

4.2 Examples

In this section, we will go through the details of using the embedding algorithm to produce an embedding of some graphs with particular rotation systems. We will focus on the complete graph on six vertices and the complete bipartite graph, $K_{3,3}$.

4.2.1 Two Embeddings of the Complete Bipartite Graph $K_{3,3}$

In the Rotation Systems of Graphs section in the Graphs chapter, two distinct rotation systems of the complete bipartite graph $K_{3,3}$ were introduced. We will show, using the embedding algorithm, that the rotation systems will induce embeddings of the graph $K_{3,3}$ into distinct surfaces. Recall, the first rotation system of $K_{3,3}$, $\Pi_1 = \{\pi_i : i = 1, \ldots, 6\}$, where the vertex codes are given by the following:

$$\pi_1 : (e_{1,4}; e_{1,5}; e_{1,6}) \quad \pi_4 : (e_{4,3}; e_{4,2}; e_{4,1})$$
$$\pi_2 : (e_{2,4}; e_{2,5}; e_{2,6}) \quad \pi_5 : (e_{5,3}; e_{5,2}; e_{5,1})$$
$$\pi_3 : (e_{3,4}; e_{3,5}; e_{3,6}) \quad \pi_6 : (e_{6,3}; e_{6,2}; e_{6,1})$$

The rotation system Π_1 results in three boundary walks via the Boundary Walk Construction method. The collection of boundary walks, denoted $\mathscr{W}_{\Pi_1}$, is as follows:

$$\mathcal{W}_1 = (e_{1,4}e_{4,3}e_{3,5}e_{5,2}e_{2,6}e_{6,1})$$
$$\mathcal{W}_2 = (e_{1,5}e_{5,3}e_{3,6}e_{6,2}e_{2,4}e_{4,1})$$
$$\mathcal{W}_3 = (e_{1,6}e_{6,3}e_{3,4}e_{4,2}e_{2,5}e_{5,1})$$

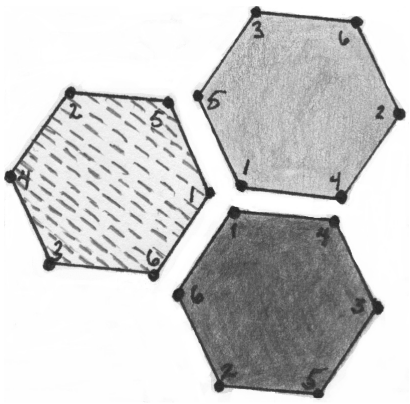

Figure 4.5. The resulting three boundary walks for $K_{3,3}$ with Π_1.

The three boundary walks are shown in Figure 4.5, where $\mathcal{W}_1$ is shaded dark gray, $\mathcal{W}_2$ is shaded light gray, and $\mathcal{W}_3$ is dashed.

Now we can use the Boundary Walk Gluing method to glue the three boundary walks together. The third boundary walk, $\mathcal{W}_3$, ends with edge $e_{5,1}$ and the second boundary walk, $\mathcal{W}_2$, begins with edge $e_{1,5}$, so we can glue the two boundary walks together on those edges. Similarly, we can glue the first boundary walk onto the end of $\mathcal{W}_3 \circ \mathcal{W}_2$, because $\mathcal{W}_2$ ends with edge $e_{4,1}$ and $\mathcal{W}_1$ begins with $e_{1,4}$. Thus the three boundary walks glued together form a collection of pairs of edges identified to each other, which will determine the surface of the embedding. Note that $\mathcal{W}_3 \circ \mathcal{W}_2 \circ \mathcal{W}_1$,

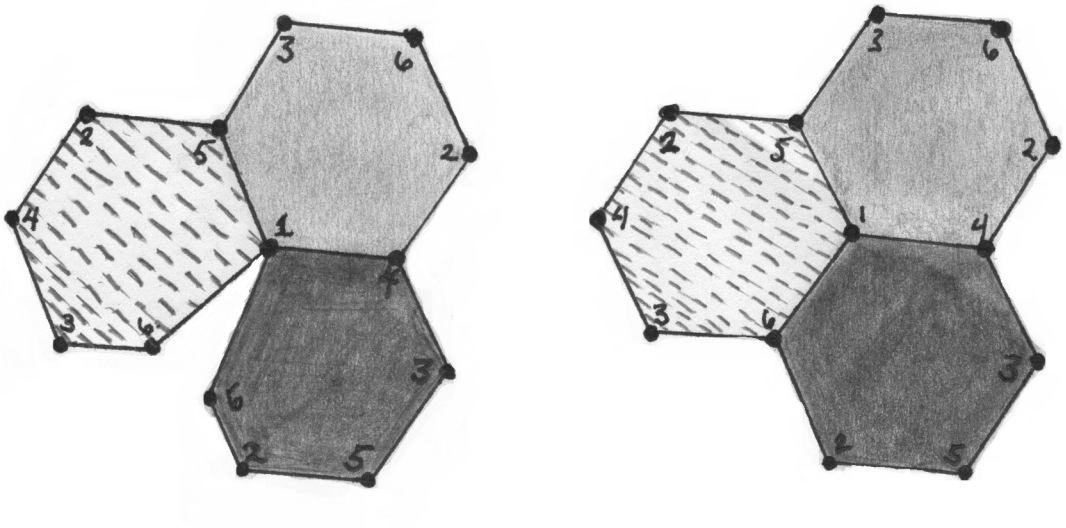

Figure 4.6. The three boundary walks for $K_{3,3}$ with Π_1 glued together forming one polygon.

as shown in Figure 4.6, represents a polygon with its edges glued together in pairs to form a surface in which the graph is embedded according to rotation system Π_1. We will identify the resulting surface by applying the torus and sphere identification codes introduced in the Compact Surface Chapter.

$$\mathcal{W}_3 \circ \mathcal{W}_2 \circ \mathcal{W}_1 = \underbrace{e_{1,6}}\, e_{6,3}e_{3,4}e_{4,2}e_{2,5}e_{5,3}e_{3,6}e_{6,2}e_{2,4}e_{4,3}e_{3,5}e_{5,2}e_{2,6}\, \underbrace{e_{6,1}}$$

$$\xrightarrow{1S^2 \quad I.D.} e_{6,3}\, \underbrace{e_{3,4}}\, e_{4,2}\, \underbrace{e_{2,5}}\, e_{5,3}e_{3,6}e_{6,2}e_{2,4}\, \underbrace{e_{4,3}}\, e_{3,5}\, \underbrace{e_{5,2}}\, e_{2,6}$$

$$\xrightarrow{T \quad I.D.} \underbrace{e_{5,3}}\, e_{3,6}e_{6,2}\, \underbrace{e_{2,4}e_{4,2}}\, e_{2,6}e_{6,3}\, \underbrace{e_{3,5}}\, \#T$$

$$\xrightarrow{2S^2 \quad I.D.} \underbrace{e_{3,6}}\, \underbrace{e_{6,2}e_{2,6}}\, \underbrace{e_{6,3}}\, \#S^2\#S^2\#T$$

$$\xrightarrow{2S^2 \quad I.D.} \#S^2\#S^2\#S^2\#S^2\#T \to T$$

Therefore, the complete bipartite graph, $K_{3,3}$, with rotation system Π_1 will embed into a torus via the embedding algorithm. We will now realize the same embedding of $K_{3,3}$ by gluing together the edges of the polygon in Figure 4.6. This process will not only identify the corresponding surface for the polygon representing the three boundary walks glued together, but also show us the actual embedding of the graph in the surface.

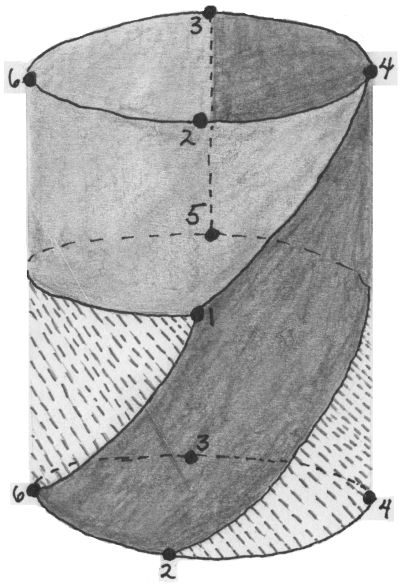

Figure 4.7. A visual construction of the embedding of $K_{3,3}$ with Π_1.

We begin this proces by first forming a cylinder by gluing together the edges $e_{3,5}$ to $e_{5,3}$ and $e_{5,2}$ to $e_{2,5}$, as shown in Figure 4.7. Next, it is necessary to glue the top and bottom edges of the cylinder together to form the torus with $K_{3,3}$ embedded according to the rotation system Π_1. The resulting embedding is shown in Figure 4.8.

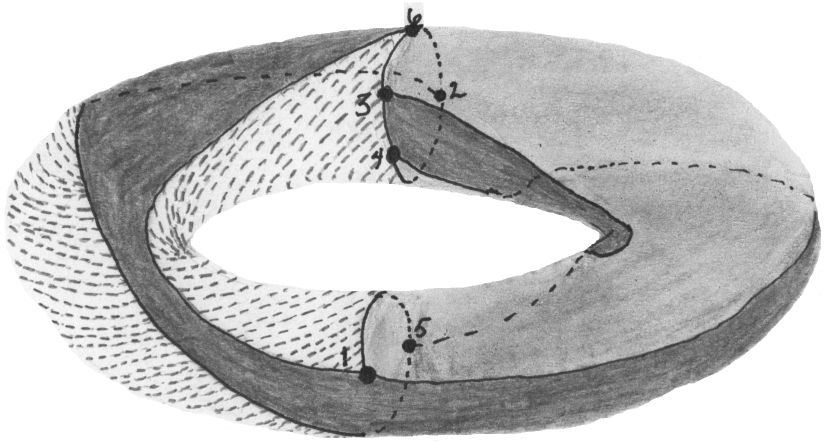

Figure 4.8. The embedding of $K_{3,3}$ with Π_1 on the torus.

Now recall the second rotation system $\Pi_2 = \{\pi_i : i = 1, \ldots, 6\}$ of the graph $K_{3,3}$. The vertex codes of this rotation system are as follows:

$$\begin{aligned} \pi_1 &: (e_{1,5}; e_{1,6}; e_{1,4}) & \pi_4 &: (e_{4,1}; e_{4,3}; e_{4,2}) \\ \pi_2 &: (e_{2,4}; e_{2,6}; e_{2,5}) & \pi_5 &: (e_{5,2}; e_{5,1}; e_{5,3}) \\ \pi_3 &: (e_{3,5}; e_{3,4}; e_{3,6}) & \pi_6 &: (e_{6,3}; e_{6,2}; e_{6,1}) \end{aligned}$$

The rotation system Π_2 of $K_{3,3}$ will induce an embedding of the graph into a different surface when compared to the rotation system Π_1. In this case, the rotation system Π_2 results in a single boundary walk for the graph. The boundary walk is given by the following:

$$\mathcal{W} : (e_{1,5}e_{5,3}e_{3,4}e_{4,2}e_{2,6}e_{6,1}e_{1,4}e_{4,3}e_{3,6}e_{6,2}e_{2,5}e_{5,1}e_{1,6}e_{6,3}e_{3,5}e_{5,2}e_{2,4}e_{4,1})$$

Since the rotation system Π_2 of $K_{3,3}$ only resulted a single boundary walk, we can immediately apply the torus and sphere identification codes to the boundary walk to identify the resulting surface.

$$\mathcal{W} = \underbrace{e_{1,5}}\, e_{5,3}e_{3,4}e_{4,2}e_{2,6}e_{6,1}\, \underbrace{e_{1,4}}\, e_{4,3}e_{3,6}e_{6,2}e_{2,5}\, \underbrace{e_{5,1}}\, e_{1,6}e_{6,3}e_{3,5}e_{5,2}e_{2,4}\, \underbrace{e_{4,1}}$$

$$\xrightarrow{T \quad I.D.} e_{4,3}e_{3,6}e_{6,2}e_{2,5}e_{5,3}e_{3,4}e_{4,2}e_{2,6}\, \underbrace{e_{6,1}e_{1,6}}\, e_{6,3}e_{3,5}e_{5,2}e_{2,4}\#T$$

$$\xrightarrow{1S^2 \quad I.D.} e_{4,3}\, \underbrace{e_{3,6}}\, e_{6,2}\, \underbrace{e_{2,5}}\, e_{5,3}e_{3,4}e_{4,2}e_{2,6}\, \underbrace{e_{6,3}}\, e_{3,5}\, \underbrace{e_{5,2}}\, e_{2,4}\#S^2\#T$$

$$\xrightarrow{T \quad I.D.} \underbrace{e_{5,3}}\, e_{3,4}e_{4,2}\, \underbrace{e_{2,6}e_{6,2}}\, e_{2,4}e_{4,3}\, \underbrace{e_{3,5}}\, \#S^2\#T\#T$$

$$\xrightarrow{2S^2 \quad I.D.} \underbrace{e_{3,4}}\, \underbrace{e_{4,2}e_{2,4}}\, \underbrace{e_{4,3}}\, \#S^2\#S^2\#S^2\#T\#T$$

$$\xrightarrow{2S^2 \quad I.D.} \#S^2\#S^2\#S^2\#S^2\#S^2\#T\#T \to T\#T$$

Thus, by the embedding algorithm the graph $K_{3,3}$ with rotation system Π_2 will embed into a connected-sum of two tori. By the Compact Surface Classification Theorem, we know that T and $T\#T$ are not homeomorphic and hence the two embeddings of $K_{3,3}$ are into distinct surfaces. In Figure 4.9, the embedding of the graph $K_{3,3}$ with rotation system Π_2 into $T\#T$ is shown.

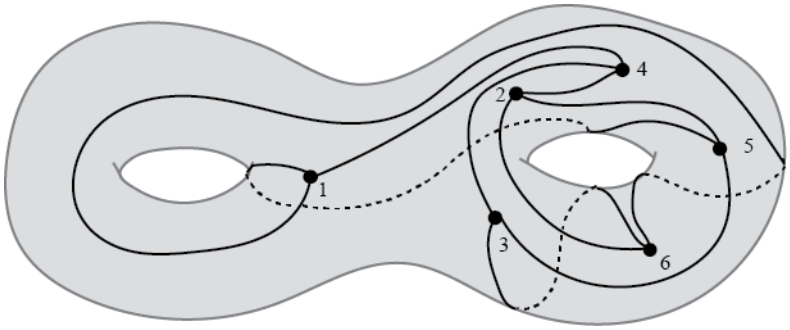

Figure 4.9. An embedding of $K_{3,3}$ with rotation system Π_2 into $T\#T$ such that the graph does not separate the surface.

4.2.2 An Embedding of the Complete Graph K_6

This section will go through the details of using the algorithm to produce an embedding of the complete graph on 6 vertices with a specific rotation system. Let $\Pi = \{\pi_i : i = 1, \ldots, 6\}$ be a rotation system of the complete graph K_6, where the specific vertex codes are given by:

$$\pi_1 : (e_{1,2}; e_{1,5}; e_{1,4}; e_{1,3}; e_{1,6}) \quad \pi_4 : (e_{4,5}; e_{4,2}; e_{4,1}; e_{4,6}; e_{4,3})$$
$$\pi_2 : (e_{2,3}; e_{2,6}; e_{2,5}; e_{2,4}; e_{2,1}) \quad \pi_5 : (e_{5,4}; e_{5,1}; e_{5,2}; e_{5,3}; e_{5,6})$$
$$\pi_3 : (e_{3,4}; e_{3,1}; e_{3,6}; e_{3,5}; e_{3,2}) \quad \pi_6 : (e_{6,1}; e_{6,4}; e_{6,3}; e_{6,2}; e_{6,5}).$$

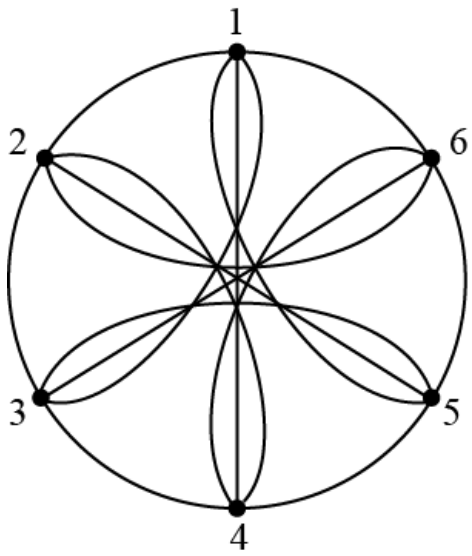

Figure 4.10. A rotation system of K_6.

This rotation system corresponds to the picture of K_6 found in Figure 4.2.2. Next, it is necessary to construct the boundary walks of the this rotation system

using the Boundary Walk Construction method. The boundary walks are as follows,

$$\mathcal{W}_1 = (e_{1,2}e_{2,3}e_{3,4}e_{4,5}e_{5,1}e_{1,4}e_{4,6}e_{6,3}e_{3,5}e_{5,6}e_{6,1})$$

$$\mathcal{W}_2 = (e_{1,5}e_{5,2}e_{2,4}e_{4,1}e_{1,3}e_{3,6}e_{6,2}e_{2,5}e_{5,3}e_{3,2}e_{2,6}e_{6,5}e_{5,4}e_{4,2}e_{2,1})$$

$$\mathcal{W}_3 = (e_{1,6}e_{6,4}e_{4,3}e_{3,1})$$

Then we can glue together the three boundary walks using the methods described previoulsy. Notice, the boundary walk $\mathcal{W}_2$ ends on edge $e_{2,1}$ and $\mathcal{W}_1$ begins on edge $e_{1,2}$, thus we can glue the two boundary walks together on that edge; denote this $\mathcal{W}_2 \circ \mathcal{W}_1$. In a similar manner, we can glue the third boundary walk to the first to obtain $\mathcal{W}_2 \circ \mathcal{W}_1 \circ \mathcal{W}_3$.

$$\begin{aligned}
\mathcal{W}_2 \circ \mathcal{W}_1 \circ \mathcal{W}_3 =& e_{1,5}e_{5,2}e_{2,4}e_{4,1}e_{1,3}e_{3,6}e_{6,2}e_{2,5}e_{5,3}e_{3,2}e_{2,6}e_{6,5}e_{5,4}e_{4,2}e_{2,3}e_{3,4}e_{4,5}e_{5,1} \\
& e_{1,4}e_{4,6}e_{6,3}e_{3,5}e_{5,6}e_{6,4}e_{4,3}e_{3,1} \\
\rightarrow & \underbrace{e_{1,5}} e_{5,2}e_{2,4}e_{4,1}e_{1,3} \underbrace{e_{3,6}} e_{6,2}e_{2,5}e_{5,3}e_{3,2}e_{2,6}e_{6,5}e_{5,4}e_{4,2}e_{2,3}e_{3,4}e_{4,5} \underbrace{e_{5,1}} \\
& e_{1,4}e_{4,6} \underbrace{e_{6,3}} e_{3,5}e_{5,6}e_{6,4}e_{4,3}e_{3,1} \\
\xrightarrow{T \quad I.D.} & \underbrace{e_{6,2}} \underbrace{e_{2,5}} e_{5,3}e_{3,2} \underbrace{e_{2,6}} e_{6,5}e_{5,4}e_{4,2}e_{2,3}e_{3,4}e_{4,5} \underbrace{e_{5,2}} e_{2,4}e_{4,1}e_{1,3} \\
& e_{3,5}e_{5,6}e_{6,4}e_{4,3}e_{3,1}e_{4,1}e_{4,6}\#T \\
\xrightarrow{T \quad I.D.} & e_{6,5} \underbrace{e_{5,4}} e_{4,2} \underbrace{e_{2,3}} e_{3,4} \underbrace{e_{4,5}} e_{5,3} \underbrace{e_{3,2}} e_{2,4}e_{4,1}e_{1,3}e_{3,5}e_{5,6}e_{6,4} \\
& e_{4,3}e_{3,1}e_{1,4}e_{4,6}\#2T \\
\xrightarrow{T \quad I.D.} & e_{5,3}e_{3,4}e_{4,2}e_{2,4}e_{4,1} \underbrace{e_{1,3}} e_{3,5} \underbrace{e_{5,6}} e_{6,4}e_{4,3} \underbrace{e_{3,1}} e_{1,4}e_{4,6} \underbrace{e_{6,5}} \#3T \\
\xrightarrow{T \quad I.D.} & \underbrace{e_{6,4}} e_{4,3} \underbrace{e_{3,5}e_{5,3}} e_{3,4} \underbrace{e_{4,1}e_{1,4}} \underbrace{e_{4,6}} \#4T \\
\xrightarrow{3S^2 \quad I.D.} & \underbrace{e_{3,4}e_{4,3}} T\#T\#T\#T \xrightarrow{1S^2 \quad I.D.} T\#T\#T\#T
\end{aligned}$$

Therefore by the embedding algorithm K_6 with rotation system Π, defined above, embeds into the connected-sum of four tori.

4.3 Remarks

Recall, the Euler Characteristic of a triangulated surface S, $\chi(S) = V - E + F$, where V is the number of vertices, E is the number of edges, and F is the number of triangles in the triangulation of S. A triangulation of a surface can be looked at as a collection of edges, vertices, and triangles that covers the surface. So if one takes the complement of the edges and vertices in a triangulated surface, the result is a collection of faces that are homeomorphic to open disks. One can think of a triangulation as a graph that is embedded in the surface such that the embedded graph segments the surface into triangles. In a similar fashion, we can define the Euler Characteristic for a graph embedded into a surface. Let G be a connected, simple graph with rotation system Π such that G has V vertices, E edges, and F boundary walks or faces. Let $\chi(\Pi)$ denote the *Euler Characteristic of the graph* Π_G embedded via the algorithm, where we define $\chi(\Pi)$ as

$$\chi(\Pi) = V - E + F.$$

Then the following lemma allows us to relate the Euler characteristic of an embedded graph to the genus of the surface in which the graph is embedded through the algorithm.

Lemma 4.3.1. *Let S be a surface and G be a connected graph that is 2-cellular embedded in S, with V vertices, E edges, and F faces. Then S is homeomorphic to the connected-sum of $k-$tori, kT, where k is defined by*

$$V - E + F = 2 - 2k.$$

If $k = 0$ then $0T$ is defined to be a sphere.

A proof of this result can be found in [10]. Every cellular embedding is uniquely determined by its rotation system, so Lemma 4.3.1 can be extended to a graph

G that is embedded via a rotation system and the Embedding Algorithm. When using the algorithm to embed a graph, the value of F is determined by the number of boundary walks produced by the specific rotation system. This leads to the definition of the *genus of an embedding of a graph G with rotation system* Π. The genus is given by

$$g(\Pi) = 1 - \tfrac{1}{2}\chi(\Pi).$$

Where the *genus of the surface, $g(S)$, of the embedded graph* is given by $g(S) := g(\Pi)$. This formulation of the genus of the embedding allows us to identify the surface produced by the Embedding Algorithm by counting the number of boundary walks, which corresponds to F. Therefore we can identify the surface without the use of the compact surface identification codes, as shown previously.

4.4 Minimal and Maximal Graph Embeddings

Consider the two embeddings of $K_{3,3}$ presented in the Examples Section, previously. Recall, the crossing number of the complete bipartite graph $K_{3,3}$ is 1 and hence by Theorem 2.2.6 the graph can be embedded into a torus. Notice, in the case of $K_{3,3}$ with rotation system Π_1, we produce an embedding of the graph into a torus. Because the crossing number of $K_{3,3}$ is $\nu(K_{3,3}) = 1$ this implies the graph cannot be embedded into a sphere and hence the torus is the smallest genus surface in which we can embed $K_{3,3}$. This observation leads to the idea of the minimum genus of a graph. The *minimum genus of a graph G*, denoted $g(G)$, is defined to be the minimum integer n such that G can embed into a connected-sum of n tori. The following proposition provides a lower bound for the genus of an arbitrary connected graph. For more details on minimum and maximum graph embeddings see [10] and [7].

Proposition 4.4.1. *Let G be a connected graph with $V \geq 3$ vertices and E edges. Then a lower bound for the minimum genus of a graph G is given by*

$$g(G) \geq \lceil \tfrac{E}{6} - \tfrac{V}{2} + 1 \rceil.$$

Proof. Let Π be an arbitrary rotation system of the graph G. Suppose the embedding algorithm produces f boundary walks. The sum of the number of edges in the collection of boundary walks of Π_G will be $2E$. Since $V \geq 3$ each boundary walk of Π_G must contain at least three or more directed edges. This implies that $2E \geq 3f$. By the Euler characteristic of the embedded graph Π_G,

$$\begin{aligned} 3\chi(\Pi) &= 3V - 3E + 3f \\ &\leq 3V - 3E + 2E = 3V - E \end{aligned}$$

In this case, the genus of the embedding of G_Π is the same as the genus of the graph G. So given the genus of an embedded graph, $g(\Pi) = 1 - \frac{1}{2}\chi(\Pi)$, and the fact that the genus of a graph must be an integer, we have the following

$$\begin{aligned} g(\Pi) &= 1 - \frac{1}{2}\chi(\Pi) \\ &\geq 1 - \frac{1}{2}(V - \frac{E}{3}) \\ &= 1 - \frac{V}{2} + \frac{E}{6} \\ g(\Pi) = g(G) &\geq \lceil 1 - \frac{V}{2} + \frac{E}{6} \rceil \end{aligned}$$

□

Similarly, we could also find a the non-trivial maximal genus surface in which a graph can be embedded. Now, consider the embedding produced of $K_{3,3}$ with rotation system Π_2 (see Figure 4.9). In this case, we saw that $K_{3,3}$ embeds into a connected-sum of two tori. Coincidentally, the maximal genus surface in which $K_{3,3}$ can be embedded is $T\#T$. Of course, we could embed $K_{3,3}$ into $5T$, for example,

but such an embedding of the graph will contain faces that are not homeomorphic to the open disk. The *maximum genus of a graph* G, denoted $g_M(G)$, is the largest integer m such that G can be embedded into a connected-sum of m tori where each face of the embedded graph is homeomorphic to the open disk. It is necessary to include the requirement that each face of the embedded graph be an open disk in its maximal surface because we could always embed the graph into a surface that contains extra holes but then it would not necessarily produce a 2-cellular embedding. Further, if one were to use the Embedding Algorithm to investigate different rotation systems of a graph to create the maximal embedding, the resulting embedded graph will always have faces that are homeomorphic to the open disk due to the construction of boundary walks. Once again, consider the embedding of $K_{3,3}$ with rotation system Π_2. Notice, $K_{3,3}$ with Π_2 produced one boundary walk via the Embedding Algorithm. This implies the embedding of $K_{3,3} \to T\#T$ only contains a single face and that when one takes the complement of $K_{3,3}$ in $T\#T$ the graph does not separate the surface.

The following proposition establishes an upper bound for the maximum genus of an embedded graph G.

Proposition 4.4.2. *Let G be a connected graph with V vertices and E edges. Then an upper bound for the maximum genus of a graph G is given by the following*

$$g_M(G) \leq \lfloor \tfrac{E-V+1}{2} \rfloor.$$

Proof. Let G be a connected graph with a rotation system Π. Suppose G_Π is embedded into a surface S via the embedding algorithm. Let f denote the number of boundary walks for the embedding of G_Π. We know that for any rotation system Π of G that $f \geq 1$. By the Euler characteristic of an embedded graph we have the following,

$$\chi(\Pi) = V - E + f \geq V - E + 1.$$

Thus by the genus of the embedding of G_Π and the fact the maximum genus of a graph is an integer, the result holds.

$$\begin{aligned} g(\Pi) &= 1 - \frac{1}{2}(\chi(\Pi)) \\ &\leq 1 - \frac{1}{2}(V - E + 1) \\ g(\Pi) = g_M(G) &\leq \lfloor \frac{E - V + 1}{2} \rfloor \end{aligned}$$

□

In Proposition 4.4.2, equality holds if and only if the rotation system of G constructs, via the Embedding Algorithm, exactly one boundary walk or exactly two boundary walks. The following theorem, attributed to Richard A. Duke [4], illustrates how we can construct embeddings of a graph G for every surface between its minimum and maximum genus surfaces.

Theorem 4.4.3. *A connected graph G has embeddings in all surfaces gT where $g \in \mathbb{Z}^+$ such that*

$$g(G) \leq g \leq g_M(G).$$

Proof. Every rotation system Π' can be obtained from any other rotation system Π of a graph G. This can be achieved by successively exchanging the order of two consecutive edges in some vertex code of a rotation system of G. Such an exchange will affect at most three boundary walks when producing an embedding of G_Π via the Embedding Algorithm. Either three boundary walks will be transformed into one boundary walk, one boundary walk will be transformed into three boundary walks, or the number of boundary walks remains the same. This will change the genus of the embedded graph by at most one, regardless of the change that occurs within the boundary walks. Therefore, going from a minimum genus embedding to a maximum genus embedding by exchanging pairs of consecutive edges in a vertex code will result in embeddings into all intermediate genus surfaces. □

One can search for the minimum or maximum embeddings of a graph G by using the methods described in the proof of Theorem 4.4.3. However, it can be difficult to search all the possible rotation systems of a graph G in efforts to construct the minimum or maximum embedding since the number of rotation systems for a graph increases exponentially with the number of edges. In the following chapter, maximum embeddings of the complete bipartite graph $K_{3,n}$ are investigated using the Embedding Algorithm.

Chapter 5
CONSEQUENCES OF THE EMBEDDING ALGORITHM

The embedding algorithm allows us to draw many conclusions about the types of surfaces that various types of graphs can be embedded into.

5.1 Embedding Trees

A tree is a graph that is connected and contains no cycles. Further, trees are minimally connected, meaning that if an edge is removed then the graph will become disconnected. It is relatively straightforward to prove that a tree is planar. However, we also have the following result as a corollary to the embedding algorithm.

Theorem 5.1.1. *Let T be a tree with $n \geq 1$ edges. Let Π_T be a rotation system of the tree. Then, using the embedding algorithm, Π_T will result in a single boundary walk that corresponds to embedding T into the sphere.*

Before we can prove this corollary, we must introduce the following result about trees.

Lemma 5.1.2. *If a tree has 1 or more edges then it has at least 2 vertices of degree 1.*

Proof. Let T be a tree with $n \geq 1$ edges. The proof will proceed by induction on n, the number of edges of the graph.

Let $n = 1$. Then T is a tree with 1 edge that is incident to two vertices. Clearly, there exists two vertices of degree 1 in T.

Assume the result holds for a tree with $n \leq m - 1$ edges. Let T be a tree with $n = m$ edges. Remove an edge from T, to form a new graph T^*. There are two cases to consider for the resulting graph T^*. First consider the case where the resulting

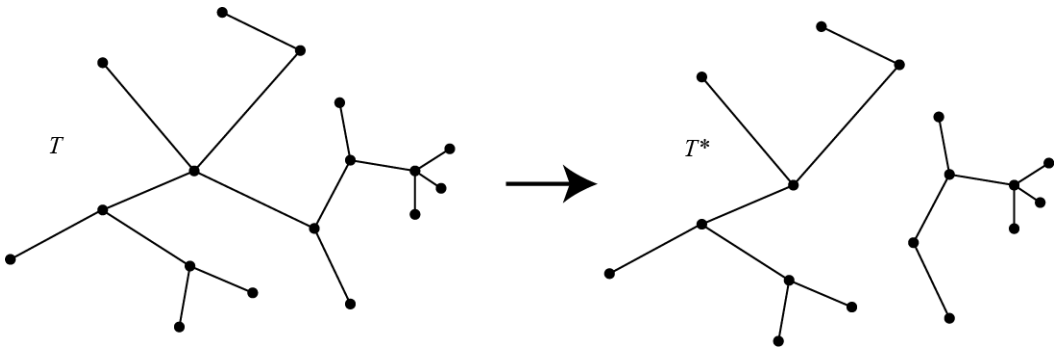

Figure 5.1. A tree is separated into two smaller tree components when an edge is removed.

graph T is split into two smaller components, both of which are trees with fewer edges than $m-1$, see Figure 5.1. So by the inductive hypothesis, in each component of the graph T^* there must be at least two vertices with degree 1. Now if we replace the removed edge, we connect the two components of T^* and recover the tree T, having at least two of the vertices of degree 1 from T^* remain.

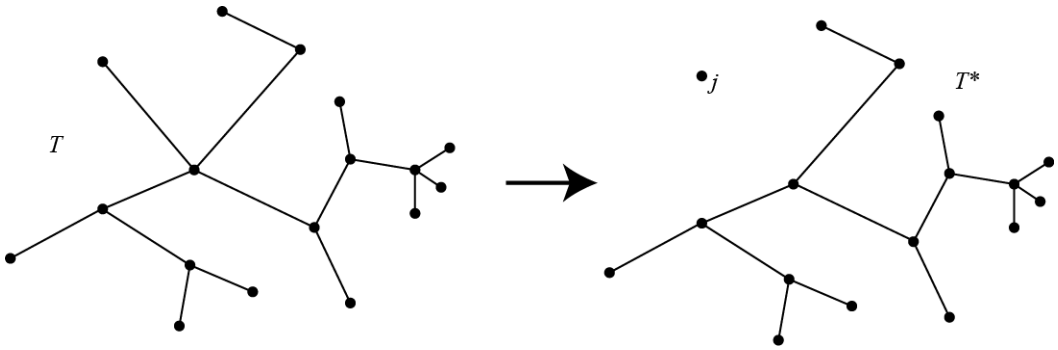

Figure 5.2. A tree is separated into a smaller tree and a singleton vertex when a edge incident to a vertex of degree 1 is removed.

Alternately, the graph T^* may consist of two components where one component is an isolated vertex, call it j, and the other component is a tree with $m-1$ edges, see Figure 5.2. The component of T^* that is a tree satisfies the inductive hypothesis. However, the only way we could have an isolated vertex j in T^* is if vertex j had degree 1 in the tree T. Therefore, if we replace the removed edge in T^* to recover

the graph T, we see there must be at least two vertices of degree 1, namely j and at least one of the vertices from T^*. □

Now we can proceed proving Theorem 5.1.1.

Proof. Let T be a tree with n edges. The proof will follow by induction on n. Let $n = 1$. Then T is a tree with one edge and two vertices. Let Π_T be a rotation system of T. The rotation system is given by $\Pi_T = \{(e_{1,2}), (e_{2,1})\}$. Use the embedding algorithm to construct a boundary walk, we will see there is only one boundary walk. The boundary walk is as follows

$$\mathcal{W} = (e_{1,2}e_{2,1}) \xrightarrow{S^2 \quad I.D.} S^2.$$

Therefore, a tree with one edge will embed into the sphere.

Assume the result holds for a tree T with $n-1$ edges with rotation system Π_T. Let G be a tree with n edges and let Π_G be a rotation system of the tree. By Lemma 5.2, we know that in G there exists at least two vertices with degree 1. Remove an edge connected to one of the vertices of degree 1, call it $e_{i,j}$ where vertex j was of degree 1. Then the result will be a graph with two components, an isolated vertex j and a component G^* that is a tree with $n-1$ edges. The graph G^* has an induced rotation system Π', which is found by removing the edges $e_{i,j}$ and $e_{j,i}$ from the rotation system Π_G. By the inductive hypothesis the rotation system Π' for G^* will result in a single boundary walk, $\mathcal{W}^*$, when the graph is embedded via the embedding algorithm, and the resulting embedding will be into the sphere. If we add the deleted edge $e_{i,j}$ back into G^* we recover the tree G; in the boundary walk $\mathcal{W}^*$ this corresponds to adding the edges $e_{i,j}e_{j,i}$ to obtain a boundary walk $\mathcal{W}$ for the tree G according to the rotation system Π_G. The addition of the edges $e_{i,j}e_{j,i}$ is the same as adding a connected-sum of a sphere to the surface obtained from $\mathcal{W}^*$ and hence the additional edges do not modify the surface in which we could embed

the tree G. Therefore, by induction, every rotation system for every tree will result in a single boundary walk and an embedding into the sphere via the embedding algorithm. □

5.1.1 Remarks

As shown in the proof of Theorem 5.1.1, every tree with any rotation system will produce, using the Embedding Algorithm, a single boundary walk. It follows, by Theorem 4.4.2, that an embedding of a tree into the sphere is a maximal embedding. Therefore, every such embedding of a tree will be both maximal and minimal. Further, because each rotation system for each tree will only produce a single boundary walk, we know the embedded tree does not separate the sphere. Also note if a graph that contains a cycle embeds into the sphere, by the Jordan Curve Theorem [1] it separates the sphere. The smallest graph that can be embedded into a sphere and

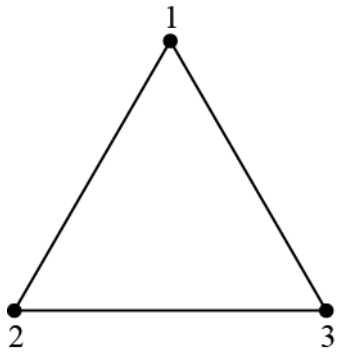

Figure 5.3. A drawing of the complete graph K_3 that corresponds to the rotation system Π_{K_3}.

separate the surface is the complete graph on three vertices, K_3. To see this, let Π_{K_3} be a rotation system of the graph given by the following vertex codes:

$$\pi_1 : (e_{1,2}; e_{1,3}) \quad \pi_2 : (e_{2,3}; e_{2,1}) \quad \pi_3 : (e_{3,1}; e_{3,2}).$$

This rotation system corresponds to the drawing of K_3 shown in Figure 5.3. The Embedding Algorithm will produce two boundary walks for the graph with rotation system Π_{K_3}.

$$\mathcal{W}_1 = (e_{1,2}e_{2,3}e_{3,1}) \quad \mathcal{W}_2 = (e_{1,3}e_{3,2}e_{2,1})$$

Then we can glue the two boundary walks together and apply the sphere identification code.

$$\mathcal{W}_2 \circ \mathcal{W}_1 = (e_{1,3}e_{3,2}e_{2,1}e_{1,2}e_{2,3}e_{3,1}) \xrightarrow{3S^2 \quad I.D.} S^2$$

The corresponding embedding of K_3 with rotation system Π_{K_3} is shown in Figure 5.4.

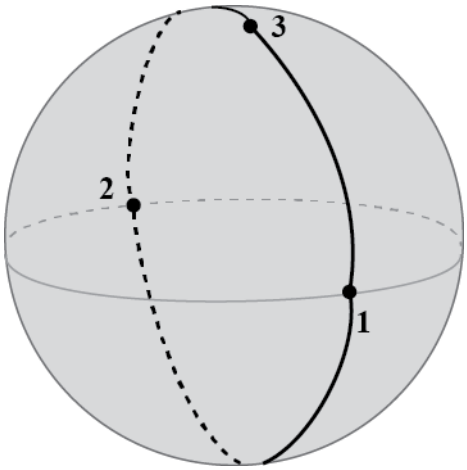

Figure 5.4. The surface-separating embedding of K_3 into the sphere via the embedding algorithm.

Again, the resulting embedding of K_3 with the embedding algorithm is both minimal and maximal. However, because the rotation system Π_{K_3} produced two boundary walks, we cannot adjust the rotation system, as described in Theorem 4.4.3, to produce a single boundary walk for K_3. If we were to switch two consecutive edges in any vertex code of Π_{K_3}, we would not change the rotation system because each vertex has degree 2. Therefore, there is no rotation system embedding of K_3 into the sphere such that the graph does not separate the surface.

5.2 Embedding the Complete Bipartite Graphs $K_{3,n}$

In this section, we will prove, using the Embedding Algorithm and a specific "standard" rotation system, the types of surfaces into which the complete bipartite graph $K_{3,n}$ embeds for any $n \in \mathbb{Z}^+$. We will define the *standard rotation system* of $K_{3,n}$ as follows

$$\Pi_{K_{3,n}} = \{\pi_I, \pi_i : I = A, B, C \text{ and } i = 1, \ldots, n\}.$$

where

$$\pi_I = (e_{I,1}; e_{I,2}; \ldots; e_{I,n}) \text{ for } I = A, B, C \text{ and}$$
$$\pi_i = (e_{i,C}; e_{i,B}; e_{i,A}) \text{ for } i = 1, \ldots, n.$$

The standard rotation system $\Pi_{K_{3,n}}$ results from the good drawing of $K_{3,n}$ shown in Figure 5.5.

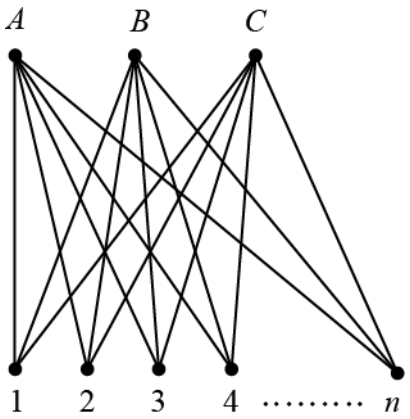

Figure 5.5. The drawing of $K_{3,n}$ that corresponds to the standard rotation system, $\Pi_{K_{3,n}}$

In Theorem 5.2.1, the only rotation system considered for $K_{3,n}$ is the standard rotation system. Recall, the vertex codes in the standard rotation system record the counter-clockwise cyclic ordering about each vertex.

Theorem 5.2.1. *For $k = 1, 2, \ldots,$ given the complete bipartite graph on 3 and n vertices, where n depends on k as below, with the standard rotation system we have the following:*

(i) For $n = 3k$, we obtain three boundary walks in the form:

$$\mathcal{W}_1 : (e_{1,C}e_{C,2}e_{2,B}\ldots e_{3k-1,B}e_{B,3k}e_{3k,A}e_{A,1})$$

$$\mathcal{W}_2 : (e_{1,B}e_{B,2}e_{2,A}\ldots e_{3k-1,A}e_{A,3k}e_{3k,C}e_{C,1})$$

$$\mathcal{W}_3 : (e_{1,A}e_{A,2}e_{2,C}\ldots e_{3k-1,C}e_{C,3k}e_{3k,B}e_{B,1})$$

and the graph $K_{3,3k}$ embeds into $(3k-2)T$.

(ii) For $n = 3k-1$, we obtain one boundary walk in the form:

$$\mathcal{W} : (e_{1,C}e_{C,2}\ldots e_{3k-2,C}e_{C,3k-1}e_{3k-1,B}e_{B,1}e_{1,A}e_{A,2}\ldots e_{3k-2,A}$$
$$e_{A,3k-1}e_{3k-1,C}e_{C,1}e_{1,B}e_{B,2}\ldots e_{3k-2,B}e_{B,3k-1}e_{3k-1,A}e_{A,1})$$

and the graph $K_{3,3k-1}$ embeds into $(3k-2)T$.

(iii) For $n = 3k-2$, we obtain one boundary walk in the form:

$$\mathcal{W} : (e_{1,C}e_{C,2}\ldots e_{3k-3,A}e_{A,3k-2}e_{3k-2,C}e_{C,1}e_{1,B}e_{B,2}\ldots e_{3k-3,C}e_{C,3k-2}$$
$$e_{3k-2,B}e_{B,1}e_{1,A}e_{A,2}\ldots e_{3k-3,B}e_{B,3k-2}e_{3k-2,A}e_{A,1})$$

and the graph $K_{3,3k-2}$ embeds into $(3k-3)T$.

Proof. The result will follow from induction on k. First let $k = 1$. For case (i), $n = 3(1) = 3$. Consider the graph $K_{3,3}$ with the standard rotation system, $\Pi_{K_{3,3}} = \{\pi_I, \pi_i | I = A, B, C \text{ and } i = 1, 2, 3\}$ given by the following vertex codes:

$$\pi_i : (e_{i,C}; e_{i,B}; e_{i,A}) \text{ for } i = 1, 2, 3 \text{ and}$$

$$\pi_I : (e_{I,1}; e_{I,2}; e_{I,3}) \text{ for } I = A, B, C.$$

Using the embedding algorithm, we obtain three boundary walks as follows:

$$\mathcal{W}_1 : (e_{1,C}e_{C,2}e_{2,B}e_{B,3}e_{3,A}e_{A,1})$$

$$\mathcal{W}_2 : (e_{1,B}e_{B,2}e_{2,A}e_{A,3}e_{3,C}e_{C,1})$$

$$\mathcal{W}_3 : (e_{1,A}e_{A,2}e_{2,C}e_{C,3}e_{3,B}e_{B,1}).$$

We can glue together the three boundary walks to obtain $\mathscr{W} = \mathcal{W}_2 \circ \mathcal{W}_1 \circ \mathcal{W}_3$ corresponding to an 18-gon with edges glued together in pairs. Using the compact surface identification codes, $\mathscr{W}$ can be manipulated to construct the surface that $K_{3,3}$ embeds into.

$$\begin{aligned}
\mathscr{W} &= \underbrace{e_{1,B}}\, e_{B,2}e_{2,A}e_{A,3}e_{3,C}e_{C,2}e_{2,B}e_{B,3}e_{3,A}e_{A,1}e_{2,C}e_{C,3}e_{3,B}\,\underbrace{e_{B,1}} \\
&\xrightarrow{S^2 \quad I.D.} \underbrace{e_{B,2}}\, e_{2,A}\,\underbrace{e_{A,3}}\, e_{3,C}e_{C,2}\,\underbrace{e_{2,B}}\, e_{B,3}\,\underbrace{e_{3,A}}\, e_{A,2}e_{2,C}e_{C,3}e_{3,B} \\
&\xrightarrow{T \quad I.D.} e_{3,C}e_{C,2}\,\underbrace{e_{2,A}e_{A,2}}\, e_{2,C}e_{C,3}\,\underbrace{e_{3,B}e_{B,3}}\,\#T \\
&\xrightarrow{2S^2 \quad I.D.} \underbrace{e_{3,C}}\,\underbrace{e_{C,2}e_{2,C}}\,\underbrace{e_{C,3}}\,\#S^2\#S^2\#T \\
&\xrightarrow{2S^2 \quad I.D.} \#S^2\#S^2\#S^2\#S^2\#T = T
\end{aligned}$$

Thus the graph $K_{3,3}$, with the standard rotation system, embeds into $(3-2)T = T$.

Next, consider case (ii) where $n = 3(1) - 1 = 2$. We are considering the graph $K_{3,2}$. The standard rotation system for $K_{3,2}$ is given by $\Pi_{K_{3,2}} = \{\pi_I, \pi_i : I = A, B, C \text{ and } i = 1, 2\}$. Where

$$\pi_1 = (e_{1,C}; e_{1,B}; e_{1,A}) \quad \pi_2 = (e_{2,C}; e_{2,B}; e_{2,A})$$
$$\pi_A = (e_{A,1}; e_{A,2}) \quad \pi_B = (e_{B,1}; e_{B,2}) \quad \pi_C = (e_{C,1}; e_{C,2}).$$

The Embedding Algorithm produces a single corresponding boundary walk,

$$\mathcal{W}(e_{1,C}\,\underbrace{e_{C,2}}\, e_{2,B}e_{B,1}\,\underbrace{e_{1,A}}\, e_{A,2}\,\underbrace{e_{2,C}}\, e_{C,1}e_{1,B}e_{B,2}e_{2,A}\,\underbrace{e_{A,1}})$$

for $\Pi_{K_{3,2}}$. Because there is one boundary walk in this case, $\mathscr{W} = \mathcal{W}$ corresponding to a 12−gon with edges glued in pairs. The torus and sphere identification codes applied to $\mathscr{W}$ identify the corresponding surface, as follows:

$$\begin{aligned}\mathscr{W} =&(e_{1,C}\,\underbrace{e_{C,2}}\,e_{2,B}e_{B,1}\,\underbrace{e_{1,A}}\,e_{A,2}\,\underbrace{e_{2,C}}\,e_{C,1}e_{1,B}e_{B,2}e_{2,A}\,\underbrace{e_{A,1}}) \\ &\xrightarrow{1T \quad I.D.} (e_{1,A}e_{A,2}\,\underbrace{e_{2,B}e_{B,2}}\,e_{2,A}e_{A,1}\,\underbrace{e_{1,C}e_{C,1}})\#T \\ &\xrightarrow{2S^2 \quad I.D.} (e_{1,A}e_{A,2}e_{2,A}e_{A,1})\#S^2\#S^2\#T \\ &\xrightarrow{2S^2 \quad I.D.} S^2\#S^2\#S^2\#S^2\#T = T.\end{aligned}$$

Therefore, $K_{3,2}$ with the standard rotation system embeds into $(2-1)T = T$.

Finally, consider case (iii) where $n = 3(1) - 2 = 1$. Here, we consider the graph $K_{3,1}$. The standard rotation system of $K_{3,1}$ is given by

$$\Pi_{K_{3,1}} = \{(e_{1,C}; e_{1,B}; e_{1,A}), (e_{A,1}), (e_{B,1}), (e_{C,1})\}.$$

The embedding algorithm produces one corresponding boundary walk for $\Pi_{K_{3,1}}$,

$$\mathcal{W} = (e_{1,C}e_{C,1}e_{1,B}e_{B,1}e_{1,A}e_{A,1})$$

Applying the sphere identification code, we see that $K_{3,1}$ embeds in the sphere.

$$\mathcal{W} = (\underbrace{e_{1,C}e_{C,1}}\,\underbrace{e_{1,B}e_{B,1}}\,\underbrace{e_{1,A}e_{A,1}}) \xrightarrow{3S^2 \quad I.D.} S^2.$$

Therefore, $K_{3,1}$ with the standard rotation system embeds into $(1-1)T = S^2$.

Now suppose the result holds for $k = 1, 2, \ldots, m$. We prove that the result holds for $k = m+1$. First, consider case (i). So $n = 3(m+1)$, and we are considering the graph $K_{3,3(m+1)}$. It is necessary to show that we obtain 3 boundary walks from the Embedding Algorithm and that $K_{3,3m+3}$ embeds into $((3m+3)-2)T = (3m+1)T$. The Embedding Algorithm results in the following boundary walks:

$$\begin{aligned}\mathcal{W}_1 : (&e_{1,C}e_{C,2}e_{2,B}\ldots e_{3m-1,B}e_{B,3m}e_{3m,A}e_{A,3m+1}e_{3m+1,C} \\ &e_{C,3m+2}e_{3m+2,B}e_{B,3m+3}e_{3m+3,A}e_{A,1})\end{aligned}$$

$$\mathcal{W}_2 : (e_{1,B}e_{B,2}e_{2,A}\ldots e_{3m-1,A}e_{A,3m}e_{3m,C}e_{C,3m+1}e_{3m+1,B}$$
$$e_{B,3m+2}e_{3m+2,A}e_{A,3m+3}e_{3m+3,C}e_{C,1})$$

$$\mathcal{W}_3 : (e_{1,A}e_{A,2}e_{2,C}\ldots e_{3m-1,C}e_{C,3m}e_{3m,B}e_{B,3m+1}e_{3m+1,A}$$
$$e_{A,3m+2}e_{3m+2,C}e_{C,3m+3}e_{3m+3,B}e_{B,1})$$

Next, we will glue together the three boundary walks such that $\mathscr{W}_{3m+3} = \mathcal{W}_3 \circ \mathcal{W}_2 \circ \mathcal{W}_1$. We apply the sphere and torus identification codes, along with the induction hypothesis, to identify the surface into which $K_{3,3(m+1)}$ embeds in this case.

$$\mathscr{W}_{3m+3} = (\underbrace{e_{1,A}}\, e_{A,2}e_{2,C}\ldots e_{3m-1,C}e_{C,3m}e_{3m,B}e_{B,3m+1}e_{3m+1,A}e_{A,3m+2}e_{3m+2,C}$$
$$e_{C,3m+3}e_{3m+3,B}e_{B,2}e_{2,A}\ldots e_{3m-1,A}e_{A,3m}e_{3m,C}e_{C,3m+1}e_{3m+1,B}e_{B,3m+2}$$
$$e_{3m+2,A}e_{A,3m+3}e_{3m+3,C}e_{C,2}e_{2,B}\ldots e_{3m-1,B}e_{B,3m}e_{3m,A}e_{A,3m+1}$$
$$e_{3m+1,C}e_{C,3m+2}e_{3m+2,B}e_{B,3m+3}e_{3m+3,A}\,\underbrace{e_{A,1}})$$

$$\xrightarrow{1S^2 \quad I.D.} (e_{A,2}e_{2,C}\ldots e_{3m-1,C}e_{C,3m}e_{3m,B}\,\underbrace{e_{B,3m+1}}\,e_{3m+1,A}\,\underbrace{e_{A,3m+2}}\,e_{3m+2,C}e_{C,3m+3}$$
$$e_{3m+3,B}e_{B,2}e_{2,A}\ldots e_{3m-1,A}e_{A,3m}e_{3m,C}e_{C,3m+1}\,\underbrace{e_{3m+1,B}}\,e_{B,3m+2}\,\underbrace{e_{3m+2,A}}$$
$$e_{A,3m+3}e_{3m+3,C}e_{C,2}e_{2,B}\ldots e_{3m-1,B}e_{B,3m}e_{3m,A}e_{A,3m+1}e_{3m+1,C}$$
$$e_{C,3m+2}e_{3m+2,B}e_{B,3m+3}e_{3m+3,A})\#S^2$$

$$\xrightarrow{1T \quad I.D.} (e_{3m+2,C}\,\underbrace{e_{C,3m+3}}\,e_{3m+3,B}e_{B,2}e_{2,A}\ldots e_{3m-1,A}e_{A,3m}e_{3m,C}\,\underbrace{e_{C,3m+1}}\,e_{3m+1,A}$$
$$e_{A,3m+3}\,\underbrace{e_{3m+3,C}}\,e_{C,2}e_{2,B}\ldots e_{3m-1,B}e_{B,3m}e_{3m,A}e_{A,3m+1}\,\underbrace{e_{3m+1,C}}\,e_{C,3m+2}$$
$$e_{3m+2,B}e_{B,3m+3}e_{3m+3,A}e_{A,2}e_{2,C}\ldots e_{3m-1,C}e_{C,3m}e_{3m,B}e_{B,3m+2})\#T\#S^2$$

$$\xrightarrow{1T \quad I.D.} (e_{3m+1,A}\,\underbrace{e_{A,3m+3}}\,e_{3m+3,B}e_{B,2}e_{2,A}e_{A,3}\ldots e_{3m-1,A}e_{A,3m}e_{3m,C}\,\underbrace{e_{C,3m+2}}$$
$$e_{3m+2,B}e_{B,3m+3}\,\underbrace{e_{3m+3,A}}\,e_{A,2}e_{2,C}\ldots e_{3m-1,C}e_{C,3m}e_{3m,B}e_{B,3m+2}\,\underbrace{e_{3m+2,C}}$$
$$e_{C,2}e_{2,B}\ldots e_{3m-1,B}e_{B,3m}e_{3m,A}e_{A,3m+1})\#T\#T\#S^2$$

$$\xrightarrow{1T \quad I.D.} (\underbrace{e_{3m+2,B}}\underbrace{e_{B,3m+3}e_{3m+3,B}}\, e_{B,2}e_{2,A}e_{A,3}\ldots e_{3m-1,A}e_{A,3m}e_{3m,C}e_{C,2}$$

$$e_{2,B}\ldots e_{3m-1,B}e_{B,3m}e_{3m,A}\,\underbrace{e_{A,3m+1}e_{3m+1,A}}\,e_{A,2}e_{2,C}\ldots e_{3m-1,C}$$

$$e_{C,3m}e_{3m,B}\,\underbrace{e_{B,3m+2}})\#T\#T\#T\#S^2$$

$$\xrightarrow{3S^2 \quad I.D.} (e_{B,2}e_{2,A}e_{A,3}\ldots e_{3m-1,A}e_{A,3m}e_{3m,C}e_{C,2}e_{2,B}\ldots e_{3m-1,B}e_{B,3m}$$

$$e_{3m,A}e_{A,2}e_{2,C}\ldots e_{3m-1,C}e_{C,3m}e_{3m,B})\#(3T)\#(4S^2)$$

Notice, the edges remaining to be identified in $\mathscr{W}_{3m+3}$ correspond to the edges in the boundary walks produced by the Embedding Algorithm for the graph $K_{3,3m}$. Thus by the inductive hypothesis we have the following:

$$\mathscr{W}_{3m+3} = \mathscr{W}_{3m}\#T\#T\#T = (3m-2)T\#(3T) = (3m+1)T$$

Therefore, with the standard rotation system we obtain 3 boundary walks, and the graph $K_{3,3m+3}$ embeds into $(3m+3-2)T = (3m+1)T$, as needed.

Next, consider case (ii), where $n = 3(m+1)-1 = 3m+2$ and we are considering the graph $K_{3,3(m+1)-1}$ with the standard rotation system. Here, it is necessary to show that the Embedding Algorithm produces one boundary walk, $\mathcal{W}$, and that $K_{3,3m+2}$ will embed into $(3m+1)T$. It is not difficult to see that the embedding algorithm results in one boundary walk. We denote it by $\mathscr{W}_{3m+2}$ and present it below. Also, to $\mathscr{W}_{3m+2}$ we apply the torus and sphere identification codes and the induction hypothesis to show that $\mathscr{W}_{3m+2} = \mathscr{W}_{3m-1}\#3T$.

$$\mathscr{W}_{3m+2} = (e_{1,C}e_{C,2}\ldots e_{3m-2,C}e_{C,3m-1}e_{3m-1,B}\,\underbrace{e_{B,3m}}\,\underbrace{e_{3m,A}}\,e_{A,3m+1}e_{3m+1,C}e_{C,3m+2}$$

$$e_{3m+2,B}e_{B,1}e_{1,A}e_{A,2}\ldots e_{3m-2,A}e_{A,3m-1}e_{3m-1,C}e_{C,3m}\,\underbrace{e_{3m,B}}\,e_{B,3m+1}$$

$$e_{3m+1,A}e_{A,3m+2}e_{3m+2,C}e_{C,1}e_{1,B}e_{B,2}\ldots e_{3m-2,B}e_{B,3m-1}e_{3m-1,A}\,\underbrace{e_{A,3m}}$$

$$e_{3m,C}e_{C,3m+1}e_{3m+1,B}e_{B,3m+2}e_{3m+2,A}e_{A,1})$$

$$\xrightarrow{1T \quad I.D.} (\underbrace{e_{A,3m+1}}\, e_{3m+1,C}e_{C,3m+2}e_{3m+2,B}e_{B,1}e_{1,A}e_{A,2}\cdots e_{3m-2,A}e_{A,3m-1}e_{3m-1,C}$$

$$\underbrace{e_{C,3m}e_{3m,C}}\, e_{C,3m+1}e_{3m+1,B}e_{B,3m+2}\,\underbrace{e_{3m+2,A}}\, e_{A,1}e_{1,C}e_{C,2}\cdots e_{3m-2,C}$$

$$e_{C,3m-1}e_{3m-1,B}e_{B,3m+1}\,\underbrace{e_{3m+1,A}}\,\underbrace{e_{A,3m+2}}\, e_{3m+2,C}e_{C,1}e_{1,B}e_{B,2}\cdots e_{3m-2,B}$$

$$e_{B,3m-1}e_{3m-1,A})\#T$$

$$\xrightarrow{1S^2\&1T \quad I.D.} (e_{A,1}e_{1,C}e_{C,2}\cdots e_{3m-2,C}e_{C,3m-1}e_{3m-1,B}\,\underbrace{e_{B,3m+1}}\, e_{3m+1,C}e_{C,3m+2}\,\underbrace{e_{3m+2,B}}$$

$$e_{B,1}e_{1,A}e_{A,2}\cdots e_{3m-2,A}e_{A,3m-1}e_{3m-1,C}e_{C,3m+1}\,\underbrace{e_{3m+1,B}}\,\underbrace{e_{B,3m+2}}$$

$$e_{3m+2,C}e_{C,1}e_{1,B}e_{B,2}\cdots e_{3m-2,B}e_{B,3m-1}e_{3m-1,A})\#T\#T\#S^2$$

$$\xrightarrow{1T \quad I.D.} (e_{B,1}e_{1,A}e_{A,2}\cdots e_{3m-2,A}e_{A,3m-1}e_{3m-1,C}\,\underbrace{e_{C,3m+1}e_{3m+1,C}}$$

$$\underbrace{e_{C,3m+2}e_{3m+2,C}}\, e_{C,1}e_{1,B}e_{B,2}\cdots e_{3m-2,B}e_{B,3m-1}e_{3m-1,A})\#T\#T\#T\#S^2$$

$$\xrightarrow{2S^2 \quad I.D.} (e_{B,1}e_{1,A}e_{A,2}\cdots e_{3m-2,A}e_{A,3m-1}e_{3m-1,C}e_{C,1}$$

$$e_{1,B}e_{B,2}\cdots e_{3m-2,B}e_{B,3m-1}e_{3m-1,A})\#T\#T\#T\#3S^2$$

As before, the edges that remain to be identified together correspond to the edges in the boundary walk for the graph $K_{3,3m-1}$ which identify to $(3m-2)T$ by the inductive hypothesis. Therefore we have the following:

$$\mathscr{W}_{3m+2} = \mathscr{W}_{3m-1}\#(3T)\#(3S^2) = (3m-2)T\#(3T) = (3m+1)T.$$

Finally, consider case (iii). Here, $n = 3(m+1)-2 = 3m+1$ and the graph $K_{3,3(m+1)-2}$ with the standard rotation system is considered. The Embedding Algorithm will produce a single boundary walk, denoted $\mathscr{W}_{3m+1}$. We show, using the Emedding Algorithm and compact surface identification codes, that $\mathscr{W}_{3m+1}$ identifies

to $(3(m+1)-3)T$.

$$\begin{aligned}
\mathscr{W}_{3m+1} = (&e_{1,C}e_{C,2}e_{2,B}\cdots e_{3m-3,A}e_{A,3m-2}e_{3m-2,C}\,\underbrace{e_{C,3m-1}}\,e_{3m-1,B}e_{B,3m}e_{3m,A}\\
&e_{A,3m+1}e_{3m+1,C}e_{C,1}e_{1,B}\cdots e_{3m-3,C}e_{C,3m-2}e_{3m-2,B}e_{B,3m-1}e_{3m-1,A}e_{A,3m}e_{3m,C}\\
&e_{C,3m+1}\,\underbrace{e_{3m+1,B}}\,e_{B,1}e_{1,A}e_{A,2}\cdots e_{3m-3,B}e_{B,3m-2}e_{3m-2,A}e_{A,3m-1}\,\underbrace{e_{3m-1,C}}\\
&e_{C,3m}e_{3m,B}\,\underbrace{e_{B,3m+1}}\,e_{3m+1,A}e_{A,1})\\
\xrightarrow{1T\quad I.D.}(&e_{B,1}e_{1,A}e_{A,2}\cdots e_{3m-3,B}e_{B,3m-2}e_{3m-2,A}\,\underbrace{e_{A,3m-1}}\,e_{3m-1,B}\,\underbrace{e_{B,3m}}\,e_{3m,A}\\
&e_{A,3m+1}e_{3m+1,C}e_{C,1}e_{1,B}\cdots e_{3m-3,C}e_{C,3m-2}e_{3m-2,B}e_{B,3m-1}\,\underbrace{e_{3m-1,A}}\,e_{A,3m}e_{3m,C}\\
&e_{C,3m+1}e_{3m+1,A}e_{A,1}e_{1,C}e_{C,2}e_{2,B}\cdots e_{3m-3,A}e_{A,3m-2}e_{3m-2,C}e_{C,3m}e_{3m,B})\#T\\
\xrightarrow{1T\quad I.D.}(&e_{3m,A}e_{A,3m+1}\,\underbrace{e_{3m+1,C}}\,e_{C,1}e_{1,B}\cdots e_{3m-3,C}e_{C,3m-2}e_{3m-2,B}\,\underbrace{e_{B,3m-1}e_{3m-1,B}}\\
&e_{B,1}e_{1,A}e_{A,2}\cdots e_{3m-3,B}e_{B,3m-2}e_{3m-2,A}e_{A,3m}\,\underbrace{e_{3m,C}}\,\underbrace{e_{C,3m+1}}\,e_{3m+1,A}\\
&e_{A,1}e_{1,C}e_{C,2}e_{2,B}\cdots e_{3m-3,A}e_{A,3m-2}e_{3m-2,C}\,\underbrace{e_{C,3m}})\#T\#T\\
\xrightarrow{1S^2\&1T\quad I.D.}(&e_{C,1}e_{1,B}e_{B,2}\cdots e_{3m-3,C}e_{C,3m-2}e_{3m-2,B}e_{B,1}e_{1,A}e_{A,2}\cdots e_{3m-3,B}e_{B,3m-2}\\
&e_{3m-2,A}\,\underbrace{e_{A,3m}e_{3m,A}}\,\underbrace{e_{A,3m+1}e_{3m+1,A}}\,e_{A,1}e_{1,C}e_{C,2}\\
&e_{2,B}\cdots e_{3m-3,A}e_{A,3m-2}e_{3m-2,C})\#T\#T\#T\#S^2\\
\xrightarrow{2S^2\quad I.D.}(&e_{C,1}e_{1,B}e_{B,2}\cdots e_{3m-3,C}e_{C,3m-2}e_{3m-2,B}e_{B,1}e_{1,A}e_{A,2}\cdots e_{3m-3,B}\\
&e_{B,3m-2}e_{3m-2,A}e_{A,1}e_{1,C}e_{C,2}e_{2,B}\cdots e_{3m-3,A}\\
&e_{A,3m-2}e_{3m-2,C})\#T\#T\#T\#S^2\#S^2\#S^2
\end{aligned}$$

Once again, $\mathscr{W}_{3m+1}$ has been reduced to the connected-sum of tori, spheres, and the collection of pairs of edges that correspond to the edges in the boundary walk $K_{3,3m-2}$. By the inductive hypothesis, we know that the remaining edges identify to a connected-sum of $3m-3$ tori. Thus, $\mathscr{W}_{3m+1} = \mathscr{W}_{3m-2}\#(3T) = (3m-3)T\#(3T) = 3T$, as needed. □

5.2.1 Remarks

As shown in the proof of Theorem 5.2.1, the surface in which $K_{3,n}$, with the standard rotation system, can be embedded depends on whether 3 divides n. In the case where $3 \nmid n$, the Embedding Algorithm only produces a single boundary walk for $\Pi_{K_{3,n}}$. By Theorem 4.4.2 we know that the embedding of $K_{3,n}$ is maximal and $K_{3,n}$ does not separate the surface. However, in the case where $3|n$, we see that the Embedding Algorithm produces three boundary walks for $\Pi_{K_{3,n}}$ and hence the resulting embedding is not maximal.

We can adjust the standard rotation system of $K_{3,n}$, via the methods described in the proof of Theorem 4.4.3, to obtain all embeddings between the minimal and maximal embeddings of the graph $K_{3,n}$. Define a new rotation system, denoted $\Pi^*_{K_{3,n}}$, obtained from the standard rotation system by switching two consecutive edges in the vertex code π_A and having all other vertex codes remain the same. The new rotation system is given by the following vertex codes:

$$\pi_A : (e_{A,2}; e_{A,1}; e_{A,3}; \ldots; e_{A,n}) \quad \pi_B : (e_{B,1}; e_{B,2}; \ldots; e_{B,n}) \quad \pi_C : (e_{C,1}; e_{C,2}; \ldots; e_{C,n})$$
$$\pi_i : (e_{i,C}; e_{i,B}; e_{i,A}) \text{ where } i = 1, \ldots, n.$$

In the case where $3|n$ the rotation system $\Pi^*_{K_{3,n}}$, using the Embedding Algorithm, will construct one boundary walk.

$$\mathcal{W} = (e_{A,2}e_{2,C}e_{C,3}e_{3,B} \cdots e_{C,n}e_{n,B}e_{B,1}e_{1,A}e_{A,3}e_{3,C} \cdots e_{B,n}$$
$$e_{n,C}e_{C,1}e_{1,B}e_{B,2}e_{2,A}e_{A,1}e_{1,C}e_{2,B}e_{B,3} \cdots e_{B,n}e_{n,A})$$

In fact, the embedding of $K_{3,n}$ with rotation system $\Pi^*_{K_{3,n}}$ when $3|n$ will always be a maximal embedding. To see this, recall that equality held in Theorem 4.4.2 when the embedding algorithm produced one boundary walk. Using the genus of an embedding, $g(\Pi) = 1 - \frac{1}{2}(\chi(\Pi))$, we can determine which surface $K_{3,n}$ embeds into. Here $K_{3,n}$ has $3 + n$ vertices, $3n$ edges, and 1 boundary walk. Thus, $g(\Pi^*_{K_{3,n}}) =$

$1 - \frac{1}{2}((3+n) - 3n + 1) = n - 1$. Therefore, when $3|n$, the maximal surface into which $K_{3,n}$ embeds is a connected-sum of $n-1$ tori.

Alternatively, we can apply the new rotation system $\Pi^*_{K_{3,n}}$ to the graph $K_{3,n}$ when $3 \nmid n$. In this case, the graph $K_{3,n}$ with the standard rotation system embedded into a surface homeomorphic to a connected-sum of $n-1$ tori and this was the maximal embedding of the graph. In the case where $3|(n-1)$, we will see that $K_{3,n}$ with $\Pi^*_{K_{3,n}}$ will embed into a surface that is between the minimal and maximal embeddings. To see this, we must first construct boundary walks that correspond to the new rotation system. In this case, $K_{3,n}$ will have 3 boundary walks, as shown below.

$$\mathcal{W}_1 :(e_{A,2}e_{2,C}2_{C,3}e_{3,B}e_{B,4}\cdots e_{B,n}e_{n,A})$$
$$\mathcal{W}_2 :(e_{C,1}e_{1,B}e_{B,2}e_{2,A}e_{A,1}e_{1,C}e_{C,2}e_{2,B}\cdots e_{A,n}e_{n,C})$$
$$\mathcal{W}_3 :(e_{B,1}e_{1,A}e_{A,3}e_{3,C}e_{C,4}\cdots e_{C,n}e_{n,B})$$

Using the genus of an embedding, we can calculate the surface into which we can embed the graph $K_{3,n}$ with rotation system $\Pi^*_{K_{3,n}}$ when $3|(n-1)$. Thus, $g(\Pi^*_{K_{3,n}}) = 1 - \frac{1}{2}((3+n) - 3n + 3) = n - 2$. Therefore, in this case the surface into which $K_{3,n}$ can be embedded is homeomorphic to a connected-sum of $n-2$ tori.

Interestingly, in the case where $3|(n+1)$, the rotation system $\Pi^*_{K_{3,n}}$ does not change the number of boundary walks produced via the Embedding Algorithm. So, here the graph $K_{3,n}$ will still embed into a connected-sum of $n-1$ tori. In both cases, if we were to find the minimal surface via changing the rotation system, we would have to switch another pair of consecutive edges in $\Pi^*_{K_{3,n}}$ to continue the search.

5.3 Future Work

The proof of Theorem 5.2.1 brought to light many interesting characteristics of the complete bipartite graph embedded with the standard rotation system. For

instance, if 3 divides n, then $\Pi_{K_{3,n}}$ produces three boundary walks using the embedding algorithm. Similarly, if 3 does not divide n, then $\Pi_{K_{3,n}}$ produces one boundary walk. To see if this pattern continues with the complete bipartite graph $K_{p,n}$, where p is a prime, the graphs $K_{5,n}$ and $K_{7,n}$ are considered.

$K_{5,n}$ n	Number of Boundary Walks	Surface
1	1	S^2
2	1	$2T$
3	1	$4T$
4	1	$6T$
5	5	$6T$
6	1	$10T$
7	1	$12T$
8	1	$14T$
9	1	$16T$
10	5	$16T$

Table 5.1. Investigating embedding $K_{5,n}$ with the standard rotation system.

Applying the Embedding Algorithm to the graph $K_{5,n}$ with the standard rotation system, we see that this boundary walk pattern continues. Particularly, for the cases explored, when 5 divides n there are 5 associated boundary walks and when 5 does not divide n there is one boundary walk for $K_{5,n}$. The resulting embedding surfaces are identified using the compact surface identification codes. To continue the exploration into $K_{p,n}$ with the standard rotation system, the complete bipartite graph $K_{7,n}$ is investigated (see Table 5.2)

As expected, in the cases explored of $K_{7,n}$, we see that when 7 divides n the Embedding Algorithm produces 7 boundary walks and that when 7 does not divide n there is one boundary walk. Thus, the following conjecture is made given the patterns observed in the cases of $K_{3,n}$, $K_{5,n}$, and $K_{7,n}$ with the standard rotation system.

$K_{7,n}$ n	Number of Boundary Walks	Surface
1	1	S^2
2	1	$3T$
3	1	$6T$
4	1	$9T$
5	1	$12T$
6	1	$15T$
7	7	$15T$
8	1	$21T$

Table 5.2. Investigating embedding $K_{7,n}$ with the standard rotation system.

Conjecture 5.3.1. *Let p be a positive prime and n any positive integer. The complete bipartite graph $K_{p,n}$ with the standard rotation system will embed into $\frac{1}{2}(2-2p-n+pn)$ tori if $p|n$ or into $\frac{1}{2}(1-p-n+pn)$ tori if $p \nmid n$.*

$K_{4,n}$ n	Number of Boundary Walks	Surface
1	1	S^2
2	2	T
3	1	$3T$
4	4	$3T$
5	1	$6T$
6	2	$7T$
7	1	$9T$
8	4	$9T$
9	1	$12T$
10	2	$13T$
11	1	$15T$
12	4	$15T$

Table 5.3. Investigating embedding $K_{4,n}$ with the standard rotation system.

Next, the graph $K_{4,n}$ with the standard rotation system is explored in efforts to investigate the relationship between the number of boundary walks associated with $K_{m,n}$ in the standard rotation system and $gcd(m,n)$ (see Table 5.3).

As shown in Table 5.3, in the cases considered, we see that when $gcd(4, n) = 4$ the Embedding Algorithm produces 4 boundary walks. Similarly, when $gcd(4, n) = 2$ there are 2 associated boundary walks and when $gcd(4, n) = 1$ there is 1 associated boundary walk for the graph $K_{4,n}$. Clearly, for the complete bipartite graph $K_{m,n}$ with the standard rotation system the number of boundary walks produced by the embedding algorithm is closely related to $gcd(m, n)$.

Conjecture 5.3.2. *Let m and n be any positive integers, then via the Embedding Algorithm $K_{m,n}$ with standard rotation system will result in $gcd(m, n)$ boundary walks.*

In future work, the $K_{p,n}$ conjecture may be proved similar to the proof of Theorem 5.2.1. The $K_{m,n}$ boundary walk conjecture may be a result of having each vertex code of the vertices in V_m having the same cyclic permutation of the edges; likewise for the vertex codes of the vertices in V_n.

REFERENCES

[1] C.C. Adams and R.D. Franzosa. *Introduction to Topology: Pure and Applied.* Pearson Prentice Hall Upper Saddle River, 2008.

[2] R. Diestel. *Graph Theory.* Graduate Texts in Mathematics. Springer, New York, 3 edition, 2005.

[3] H. Dudeney. *Amusements in Mathematics.* Dover Recreational Math. Dover, Mineola, New York, 1958.

[4] R.A. Duke. The genus, regional number, and betti number of a graph. *Canadian Journal of Mathematics*, 18(4):817–822, 1966.

[5] J. Edmonds. A combinatorial representation for polyhedral surfaces. *Notices of the American Mathematical Society*, 7, 1960.

[6] J. Gallier and D. Xu. *A Guide to the Classification Theorem for Compact Surfaces.* Geometry and Computing. Springer, New York, 2010.

[7] J.L. Gross and T.W. Tucker. *Topological Graph Theory.* Dover Publications, Mineola, New York, 2001.

[8] L. Heffter. Uber das problem der nachbargebiete. *Mathematische Annalen*, 38(4):477–508, 1891.

[9] K. Kuratowski. Sur le probléme des courbes gauches en topologie. *Fundamenta Matematicae*, 15:271–283, 1930.

[10] B. Mohar and C Thomassen. *Graphs on Surfaces.* Johns Hopkins Studies in the Mathematical Sciences. The Johns Hopkins University Press, Baltimore, Marylad, 2001.

[11] T. Radoó. Uber den begriff der riemannsche flache. *Acta Matematica Szeged*, 2:101–121, 1925.

[12] C. Thomassen. Kuratowski's theorem. *Journal of Graph Theory*, 5:225–241, 1981.

[13] Paul Turán. A note of welcome. *Journal of Graph Theory*, 1:7–9, 1977.

BIOGRAPHY OF THE AUTHOR

Sophia Potoczak was born in Birmingham, Michigan to Julieann Schroeter and Dennis Potoczak on April 19, 1990. In June of 2008, she graduated from Seaholm High School in Birmingham, Michigan. In the following fall, Sophia began attending Albion College, in Albion, Michigan, where she first began her studies in mathematics. She graduated from Albion College in May 2012 with a Bachelor of Arts in Applied Mathematics and a minor in geology. Immediately after graduation, Sophia decided to continue her studies of mathematics at the University of Maine.

Sophia N. Potoczak is a candidate for the Master of Arts degree in Mathematics from The University of Maine in August 2014.