

2013

# The Machete Number

David Freund

*The College of Wooster*, [dfreund5@gmail.com](mailto:dfreund5@gmail.com)

Follow this and additional works at: <https://openworks.wooster.edu/independentstudy>



Part of the [Applied Mathematics Commons](#)

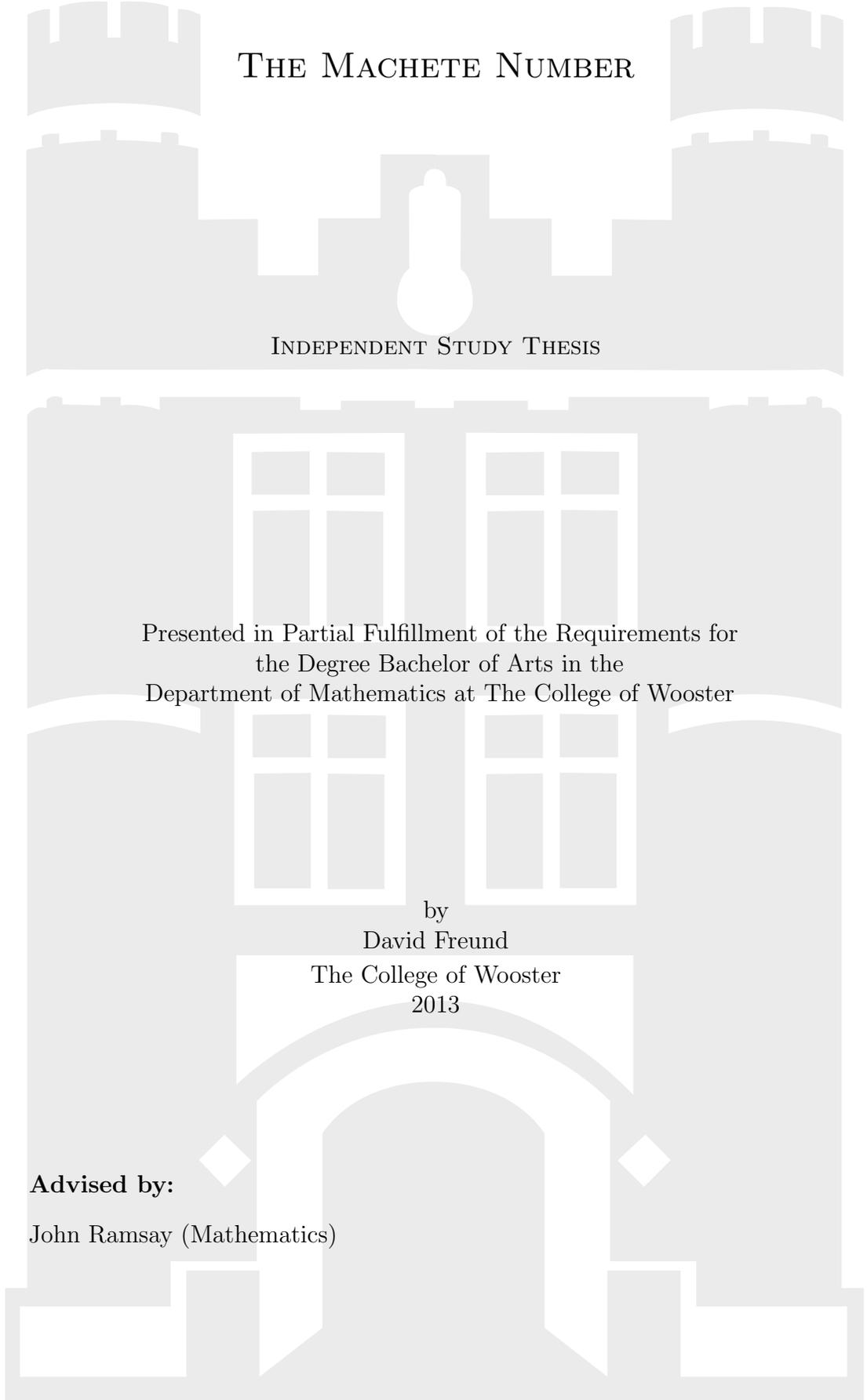
---

## Recommended Citation

Freund, David, "The Machete Number" (2013). *Senior Independent Study Theses*. Paper 961.

<https://openworks.wooster.edu/independentstudy/961>

This Senior Independent Study Thesis Exemplar is brought to you by Open Works, a service of The College of Wooster Libraries. It has been accepted for inclusion in Senior Independent Study Theses by an authorized administrator of Open Works. For more information, please contact [openworks@wooster.edu](mailto:openworks@wooster.edu).



# THE MACHETE NUMBER

INDEPENDENT STUDY THESIS

Presented in Partial Fulfillment of the Requirements for  
the Degree Bachelor of Arts in the  
Department of Mathematics at The College of Wooster

by  
David Freund  
The College of Wooster  
2013

**Advised by:**

John Ramsay (Mathematics)





---

THE COLLEGE OF

---

WOOSTER

---

© 2013 by David Freund



# ABSTRACT

Knot theory is a branch of topology that deals with the structure and properties of links. Employing a variety of tools, including surfaces, graph theory, and polynomials, we develop and explore classical link invariants. From this foundation, we define two novel link invariants, braid height and machete number, and investigate their properties and connection to classical invariants.



## ACKNOWLEDGMENTS

I am amazed by the amount of support I received trying to get this project off the ground. There are many people that I need to thank for their support, although their support may not directly relate to this project.

The first person who deserves recognition is Dr. Carl Stitz. He is undoubtedly my first mathematical father. While I was at Lakeland Community College, and even now, he has been an amazing friend and professor. It is impossible to list all that he has done for me, but I know that he has pushed me to challenge myself mathematically and that he has encouraged my fascination with and love of mathematics. Even though I declared myself a math major when I took my first class with him, I became a real math major because of him.

My family also deserves recognition. It may be a simple phrase, but they have done a great deal for me and have always supported me. For this and so much more, I cannot thank them enough. Although they may not understand me when I start “speaking in math,” they try to learn bits and pieces and they encourage me to go further.

I want to thank my advisor, Dr. John Ramsay, for all of his help and support with this project. Whenever our knot theory research group would struggle with new ideas, we would go back to trying to define a link invariant that could somehow involve machetes. Although I ultimately came up with the mathematical definition of the machete number, Dr. Ramsay is responsible for the joke that started the idea. More than anyone else, he made this project possible.

I also want to thank the College of Wooster Mathematics Department for giving me a home for the past four years. Each and every professor in the department has made an impact on my life and encouraged me mathematically. I have thoroughly enjoyed getting to know my professors and they have made me feel welcome at Wooster.

# CONTENTS

Abstract	v
Acknowledgments	vii
Contents	ix
List of Figures	xi
CHAPTER	PAGE
1 Introduction	1
2 Introductory Knot Theory	5
2.1 Link Projections . . . . .	9
2.2 Link Equivalence . . . . .	13
2.3 Link Invariants . . . . .	18
2.3.1 Crossing Number . . . . .	19
2.3.2 Linking Number . . . . .	21
2.4 Types of Links . . . . .	27
2.4.1 Amphicheiral Links . . . . .	27
2.4.2 Composite Knots . . . . .	28
3 Surfaces	31
3.1 Links on Surfaces . . . . .	34
3.2 Surfaces with Boundary . . . . .	35
3.3 Oriented Surfaces . . . . .	36
3.4 Genus and the Classification of Surfaces . . . . .	37
3.5 Euler Characteristic . . . . .	39
3.6 Seifert Surfaces . . . . .	42
3.7 Link Genus . . . . .	46
4 Graph Theory and Seifert Matrices	49
4.1 Graph Theory . . . . .	49
4.1.1 Seifert Graphs . . . . .	54
4.2 Homology of Graphs . . . . .	55
4.3 Seifert Matrices . . . . .	61
4.3.1 S-equivalence of Seifert Matrices . . . . .	67
4.3.2 Determinant and Signature of a Link . . . . .	69

5	Knot Polynomials	73
5.1	Alexander Polynomial . . . . .	73
5.1.1	Computation by Seifert Matrix . . . . .	74
5.1.2	Properties of the Alexander Polynomial . . . . .	75
5.1.3	Computation by Conway Polynomial . . . . .	79
5.2	HOMFLY-PT Polynomial . . . . .	85
6	Braids	89
6.1	Braid Representations . . . . .	89
6.1.1	Closed Braid Representation . . . . .	91
6.1.2	Braid Words . . . . .	99
6.2	Braid Equivalence . . . . .	101
6.2.1	Braid Moves . . . . .	101
6.2.2	Markov Equivalence . . . . .	103
6.3	Braid Index . . . . .	106
7	Braid Height and the Machete Number	109
7.1	Braid Height . . . . .	109
7.2	The Machete Number . . . . .	119
7.3	Bounding the Machete Number . . . . .	123
7.4	Conclusions and Future Work . . . . .	129
	References	133

# LIST OF FIGURES

Figure		Page
2.1	Example of an intuitive knot (left) and a mathematical knot (right).	5
2.2	A 0-dimensional knot (left) and a 2-dimensional knot (right).	6
2.3	Three different polygonal curves.	7
2.4	Two different ways of interweaving two unknots.	8
2.5	Changing a point of self-intersection (left) into a crossing (right).	10
2.6	An oriented knot.	10
2.7	$K_{1,i}$ .	11
2.8	$K_{2,i}$ .	11
2.9	Unknot, trefoil knot, figure-eight knot, and Hopf link (left to right)[1].	12
2.10	Unlink (trivial link) of $n$ components [1].	12
2.11	The general form of the $n$ -component Hopf link.	12
2.12	Three examples of homotopic curves.	13
2.13	Bachelor's unknotting.	14
2.14	Three projections of the unknot.	16
2.15	The three Reidemeister moves.	16
2.16	Different results from a Reidemeister I move.	16
2.17	Equivalence of two trefoil knot projections.	17
2.18	A link of crossing number 2.	20
2.19	Positive crossing (left) and negative crossing (right).	22
2.20	Hopf Link with linking number -1.	24
2.21	Link with linking number 3.	24
2.22	A 3-component link with total linking number 1.	24
2.23	Reidemeister II does not affect linking number.	25
2.24	Oriented Whitehead link.	27
2.25	The composition of $K_1$ and $K_2$ .	29
2.26	Composition of two trefoil knots whose orientations match.	30
2.27	Composition of two trefoil knots whose orientations do not match.	30
2.28	Decomposition of a composite knot.	30
3.1	$S^1$ (left) and $S^2$ (right).	32
3.2	Intersection of a ball with $S^2$ .	33
3.3	Torus.	33
3.4	Isotopic representations of $S^2$ .	34

3.5	Strand pulled around the sphere. . . . .	35
3.6	Surfaces with boundary. . . . .	36
3.7	Band (left) and Möbius band (right). . . . .	37
3.8	Double torus with non-separating loops. . . . .	38
3.9	Intersecting non-separating loops on the torus. . . . .	39
3.10	Triangulations of the sphere and the top half of the torus. . . . .	39
3.11	Non-triangulations of a surface. . . . .	40
3.12	Simpler torus triangulation. . . . .	41
3.13	Smoothing of a crossing. . . . .	42
3.14	Twisted bands corresponding to different crossings. . . . .	43
3.15	Impossible Seifert circle orientations. . . . .	44
3.16	Seifert surface of a trefoil knot. . . . .	45
3.17	Seifert surface of a 3-component Hopf link. . . . .	45
3.18	Triangulation of a band and a Seifert circle which has 3 attached bands. . . . .	47
4.1	Example graphs. . . . .	50
4.2	A tree with 6 vertices. . . . .	52
4.3	A graph and a spanning tree. . . . .	52
4.4	A directed graph. . . . .	54
4.5	Seifert graphs of Seifert surfaces from Example 3.5. . . . .	55
4.6	Two cycles in $T \cup e_i$ . . . . .	59
4.7	Right-hand orientation of a section of a thickened Seifert surface. . . . .	62
4.8	Twisted bands in a thickened Seifert surface. . . . .	62
4.9	Homology basis elements and copies in thickened Seifert surface. . . . .	63
4.10	Loop interactions and resulting crossings in thickened Seifert surface. . . . .	63
4.11	Seifert surface and Seifert graph of the trefoil knot projection for Example 4.5. . . . .	64
4.12	Trefoil knot Seifert surface with homology basis curves for Example 4.5. . . . .	65
4.13	Interaction of $\alpha_1$ and $\alpha_1^*$ in the thickened surface (Figure 4.12). . . . .	65
4.14	Intersection of $\alpha_1$ and $\alpha_2$ in the Seifert surface from Figure 4.12. . . . .	66
4.15	Links using mixed basis elements from Figure 4.12. . . . .	66
4.16	Adding or removing a tube. . . . .	69
5.1	Two surfaces joined by a band. . . . .	76
5.2	Result of attaching twisted bands between disjoint Seifert surfaces. . . . .	78
5.3	Presentation of $\beta_i$ in $\tilde{F}$ . . . . .	78
5.4	Link formed by $\beta_i$ and $\beta_i^*$ . . . . .	79
5.5	Neighborhood of skein diagrams. . . . .	80
5.6	Trefoil knot projection and corresponding skein diagrams. . . . .	81
5.7	Hopf link projection and corresponding skein diagrams. . . . .	81
5.8	Disconnected $L_0$ (left) and $L_-$ (right) for proof of Theorem 5.4. . . . .	83
5.9	Interaction of $\beta$ and $\beta^*$ in Seifert surfaces of $D_+$ and $D_-$ . . . . .	84
5.10	Skein diagrams for Example 5.3. . . . .	86
6.1	Example of a braid. . . . .	90

6.2	Oriented diagram of Figure 6.1. . . . .	91
6.3	Closure of Figure 6.1. . . . .	92
6.4	Compatible Seifert circles. . . . .	93
6.5	Incompatible Seifert circles. . . . .	94
6.6	Vogel's reducing move. . . . .	94
6.7	General form of the closure of a braid $B$ , before and after smoothing every crossing. . . . .	95
6.8	Changes in Seifert circles following a reducing move. . . . .	96
6.9	Bounded disks before and after a reducing move. . . . .	97
6.10	Isotopic braids on $S^2$ . . . . .	98
6.11	The 3-component Hopf link and its Seifert surface before and after a reducing move. . . . .	99
6.12	A braid representation of the 3-component Hopf link. . . . .	99
6.13	General form of braid generators. . . . .	100
6.14	Braid moves 1, 2 and 3. . . . .	102
6.15	Example of braid conjugation. . . . .	104
6.16	Example of braid stabilization. . . . .	105
6.17	Result of a stabilization operation. . . . .	105
7.1	Braid generators which can be presented on the same horizontal line. . . . .	110
7.2	$B_3$ , the composition of $B_1$ and $B_2$ . . . . .	113
7.3	Composition of $B_1$ and $B_2$ with $\sigma_n$ in an existing braid level. . . . .	114
7.4	Composition of a 4-component Hopf link with another Hopf link. . . . .	116
7.5	A braid representation of the 3-component Hopf link. . . . .	117
7.6	Deomposition of a braid height 3 link. . . . .	118
7.7	Effect of smoothing on strands of different components. . . . .	129



## INTRODUCTION

Knot theory is a subfield of topology that investigates mathematical knots and their properties. It is suspected that C.F. Gauss was the first individual to realize the possibility of studying knots mathematically [6]. His 1833 research in electrodynamics led to an analytic formulation of the linking number, a topic we will discuss later [6]. However, J.B. Listing, one of Gauss' students, was the one responsible for furthering the development of the theory [5].

A significant portion of the early interest in knot theory was generated by chemists rather than mathematicians. In the 1880s, it was believed that there was a substance which pervaded all of space called ether [1]. Lord Kelvin, also known as William Thomson, hypothesized that atoms were distinct knots contained in the ether [1]. Hence different knots would correspond to different atoms [1]. However, an 1887 experiment by Michelson and Morley demonstrated Lord Kelvin was quite wrong [1].

By the time that Lord Kelvin's claim was shown to be false, mathematicians had already become more invested in the theory of knots. A group of British mathematicians attempted to classify the different types of knots [6]. Since their work began before the advent of topology and algebraic topology, most of their results relied on combinatorial methods [6]. In the end, they were able to classify knots up to what they called "tenfold knottiness" [6].

For much of the 1900s, it was primarily mathematicians who were interested in

knot theory. During the early 1900s, the American mathematician J.W. Alexander was the first to show that there was a significant connection between knot theory and the study of 3-dimensional topology [5]. In the late 1920s and 1930s, German mathematicians found some connections between knot theory and algebraic geometry [5].

During the 1980s, knot theory attracted the attention of scientists from various fields [1, 5]. Physicists latched onto a discovery of V.F.R. Jones which placed knot theory solidly in the realm of mathematical physics: a relationship between knot theory and solvable models of statistical mechanics [5]. Meanwhile, biochemists observed knotting within strands of DNA and synthetic chemists realized the possibility of understanding properties of molecules through knotting [1].

This brings us to the modern study of knots, which continues to investigate the various properties of knots. Our particular focus is on a novel knot property, which we will call the machete number. However, in order to fully appreciate its definition and potential importance, we must catch up on nearly two centuries worth of research. Chapter 2 includes basic knot-theoretic concepts as well as some of the properties that are of interest to mathematicians. Chapter 3 focuses on surfaces which, although a slight divergence from knot theory, lays the foundation for knot properties discussed in Chapters 4 and 5. Chapter 6 introduces braids, which provides another way of examining knots. Finally, in Chapter 7, we formally introduce and discuss the machete number.

Given that knot theory has been built from a topological foundation, our exploration of knot theory will employ topological ideas. To avoid burdening our introduction to knot theory with a long list of definitions, we assume a familiarity with a wide variety of notions that are typically introduced during a first course in topology: topological space, interior, boundary, basis, continuous function, homeomorphism, topological invariant, Hausdorff space, connected space, and compactness. When working in

a Euclidean space, we also assume that the topology of the space is the standard topology. As a consequence, all relevant topological notions will be with respect to the standard topology.

Before we can begin a discussion of knot theory, we must establish the notation that will be used without explanation:

$\mathbb{N}$  : the set of natural numbers

$\mathbb{Z}$  : the set of integers

$\mathbb{R}$  : the set of real numbers

$\mathbb{R}^n$  :  $n$ -dimensional Euclidean space

$X \times Y$  : the Cartesian product of  $X$  and  $Y$

$\mathbb{Z}[x, y]$  : the space of polynomials in variables  $x$  and  $y$  with coefficients from  $\mathbb{Z}$

$A^T$  : the transpose of the matrix  $A$

$\det(A)$  : the determinant of the matrix  $A$

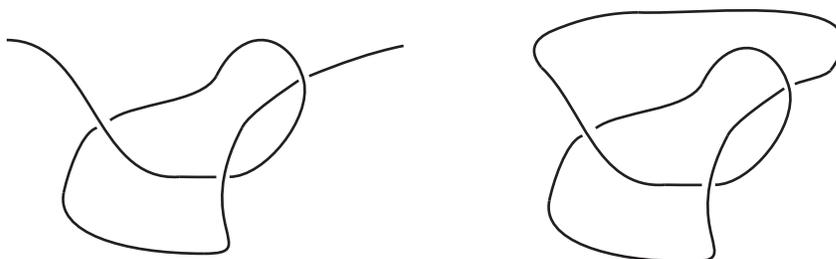
$\partial A$  : the boundary of the set  $A$

In some cases, we will encounter repeat notation which will refer to a property of a link instead of, for example, a matrix. While this may seem confusing, it will be clear which concept is being invoked by the context.



## INTRODUCTORY KNOT THEORY

To this point, we have not formally defined a knot but our entire discussion of knot theory rests on understanding their properties. An intuitive definition of a knot might be an interwoven piece of string. After all, we tie our shoes with a single shoe string. Indeed, most other knots that we use have a similar structure, although the interweaving differs significantly. Unfortunately for our intuition, this is not the type of knot we are investigating. However, we can form a mathematical knot by taking this interwoven string and gluing the ends together as in Figure 2.1 [1]. Another example involves gluing the ends of an untwisted string together. In this case we produce the simplest knot, which is known as the unknot or the trivial knot [1]. Unlike the intuitive knots, these “closed” knots cannot be unraveled by simply reversing the knotting process [2]. The only other informal requirement we will impose is that the “string” have a thickness of a single point [1].



**Figure 2.1:** Example of an intuitive knot (left) and a mathematical knot (right).

The formal definition of a knot is significantly more broad than what we discussed above. First of all, the knot we mentioned above exists in three dimensions. It is not only possible to create knots in higher (and lower) dimensions, but there are researchers who study these knots [1]. Moreover, we have assumed that we are working in Euclidean space, which is not necessary for knot theory to be studied [6].

Secondly, we can cut a knot at any point, completely unravel it, and then glue the ends back together to form a circle. Hence, any knot we created is homeomorphic to the circle and so knots are 1-dimensional. A complete definition would allow for  $n$ -dimensional knots, although our intuition would rebel at considering either of the knots in Figure 2.2 to truly be knots [6].



**Figure 2.2:** A 0-dimensional knot (left) and a 2-dimensional knot (right).

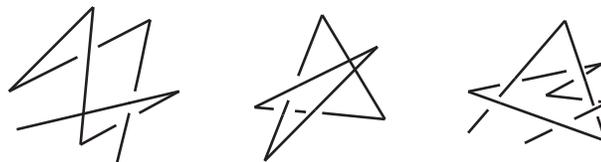
For our purposes, we will limit ourselves to a strict definition of a knot. First, we need to change how we look at knots.

**Definition 2.1** [4] *A **polygonal curve**  $P \subset \mathbb{R}^3$  generated by an ordered  $n$ -tuple of distinct points,  $(p_1, p_2, \dots, p_n)$ , is the union of line segments connecting  $p_i$  and  $p_{i+1}$ , denoted  $[p_i, p_{i+1}]$ , for  $i < n$ . Hence  $P = \bigcup_{i=1}^{n-1} [p_i, p_{i+1}]$ .*

*$P$  is a **closed polygonal curve** if  $p_n = p_1$ .  $P$  is a **simple polygonal curve** if, for all  $i < n$ ,  $[p_i, p_{i+1}]$  intersects at most two other line segments of  $P$  and the intersection points are either  $p_i$  or  $p_{i+1}$ .*

In essence, simple polygonal curves are the collection of lines in space where, away from the endpoints, lines are disjoint. Meanwhile, closed polygonal curves eventually return to the starting point. Let us consider some simple examples.

**Example 2.1** *In Figure 2.3, we have three different types of polygonal curves. The first two curves are not simple because there are two lines which intersect in the middle of a line segment. The first and last curves are not closed because there are endpoints of line segments that are only connected to one line segment. However, the middle curve is closed and the right-most curve is simple.*



**Figure 2.3:** Three different polygonal curves.

Using polygonal curves, we can formulate one definition of a knot. Using this definition, effectively reduce knots to being a complicated 3-dimensional game of connect the dots.

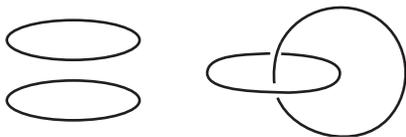
**Definition 2.2** [2, 4] A **(tame) knot**,  $K \subset \mathbb{R}^3$ , is a simple closed polygonal curve.

For the remainder of the thesis, we will assume that all knots are tame knots. In practice, we will only be considering smooth representations of knots, but we will require that there is always “some space” between different segments of the knot. Formally this concept is known as local flatness, but that is a different approach to discussing knots [2, 3]. Local flatness avoids the issue of so-called “wild” knots. Although we will not be proving that this is equivalent to having a simple closed polygonal curve, we could approximate any smooth curve by a set of tiny straight lines [2]. Conversely, we can easily smooth out any set of lines into a curve [2].

While it may appear overly specific to consider such a limited category of knots, these knots are the basis of classical knot theory [6]. Not only are they easy to visualize, but there is a substantial body of theory that they have generated. We will also be discussing a close relative of the knot, which is called a link.

**Definition 2.3** [2] A **link**,  $L \subset \mathbb{R}^3$ , can be expressed as the disjoint union  $L = K_1 \cup K_2 \cup \dots \cup K_n$  where each  $K_i$  is a knot. Each  $K_i$  is considered to be a **component** of  $L$  and  $\mu(L) = n$  is the **multiplicity** of  $L$ .

Simply put, a link is a collection of knots that have been twisted together in some fashion. The multiplicity of the link tells us how many knots are in this collection. However, while our definition of a link tells us the knots involved in a link, it does not tell us about the interactions between the various components [2]. For instance, as in Figure 2.4, we could interweave a given set of components in different ways and generate quite different links.



**Figure 2.4:** Two different ways of interweaving two unknots.

An important distinction to note is that every knot is a link with one component, but not every link is a knot. There are some knot theoretic results which rely on the structure of a knot while others generalize to links. Given that we have already limited our discussion to 1-dimensional knots in  $\mathbb{R}^3$ , we will be proving results for links whenever possible. This may complicate some of our proofs, but having a general result is worthwhile.

Before we move on to other significant concepts in knot theory, we should introduce a specific type of link – the oriented link. Rather than attempting to formally define orientation, it is enough to provide an analogy. Suppose there is an ant on a knot and it walks around the knot in one direction. The direction the ant is traveling is the **orientation** of the knot and we would call it an **oriented knot**. Since our knots are actually one-point thick, ants cannot walk on mathematical knots, but we can still impose a direction on them. To generalize, an **oriented link** is a link where each component is an oriented knot.

It may seem unnecessary to distinguish between oriented and unoriented links, especially since it has nothing to do with the knottedness of the link. However, orientation will play a key role for some of our later concepts, such as linking number and the formation of Seifert surfaces. Thus we will distinguish between links where the only differences are the orientations placed on the components of the links.

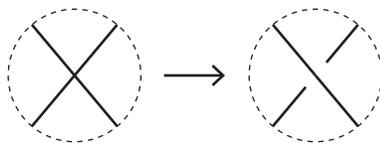
## 2.1 LINK PROJECTIONS

Although we can easily visualize links, they exist in 3-space and so attempting to graphically represent them can be difficult. At the very least, it requires a modicum of artistic ability and, for those who want to quickly represent a link, this presents a challenge. However, with a bit of projective geometry, we can fix this issue.

Let us imagine that we have a link floating in 3-space. If we were to take a sufficiently large camera some distance away from the link and snap a photo, we could obtain a 2-dimensional representation of the link [1]. Unfortunately, wherever it appears that the string overlaps itself, there will be self-intersection points in this image, despite the string not touching in 3-space.

Assume that only two projected segments of string intersect in some arbitrarily small open disk around each point of self-intersection. Moreover, assume that this is the only intersection point of the strands within this disk. We call such intersection points **crossings** [1]. We can alter the photograph so that, at every crossing, it is immediately apparent which string is closer to the camera and which is further away by altering the disk as shown in Figure 2.5. We call the closer strand the **overcrossing strand** and the further strand the **undercrossing strand**.

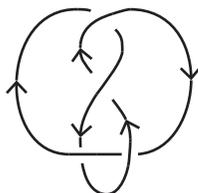
Once every crossing has been appropriately distinguished, we call the result of a projection of a link onto the plane a **knot** or **link projection** (equivalently, **knot** or



**Figure 2.5:** Changing a point of self-intersection (left) into a crossing (right).

**link diagram**) [1]. As we will see, this notion is fundamental to our understanding of knot theory.

If our link had an orientation, then the link projection will inherit the orientation from the link. We know that each component of the link was oriented, so we can visualize the imposition of an orientation on the diagram by taking a video of ants walking on each component of the link in 3-space. For our convenience, as in Figure 2.6, we will denote the orientation of a component in the link diagram by adding small arrows to it.



**Figure 2.6:** An oriented knot.

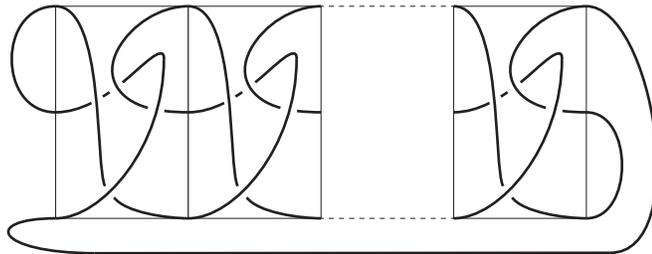
Despite the way that link projections have been introduced, it may not be immediately apparent that every link has a link projection. It is conceivable that there are complicated links such that, independent of how we photograph the link, the image will contain intersection points that involve more than two pieces of string. In the next section, we will show that this cannot happen and so every link can be represented in the plane. Even though we cannot prove this statement now, we will work with the implications to obtain a better understanding of links.

As noted earlier, we have chosen to examine tame knots. If we only required a closed simple smooth curve embedded in  $\mathbb{R}^3$ , we would be allowing erratic behavior in

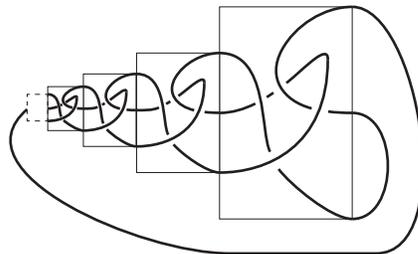
our knots and links. For a demonstration of this undesirable behavior, let us consider the projection of the “knots” in the following example.

**Example 2.2** *Let us consider the two knots in Figures 2.7 and 2.8. Both of these knots were formed iteratively by gluing  $i$  copies of strings in the prescribed pattern together. However, for the knot in Figure 2.8, the size of the bounding rectangle decreases by a constant factor [2]. For any  $i \in \mathbb{N}$ , it turns out that both  $K_{1,i}$  and  $K_{2,i}$  are the same knot [2].*

*As  $i \rightarrow \infty$ ,  $K_{1,i}$  does not change while  $K_{2,i}$  becomes complicated. One way to think about the difference is that the smaller rectangles in  $K_{2,\infty}$  allow for the formation of a limit point. However, in  $K_{1,\infty}$ , no such limit point forms and so  $K_{1,\infty}$  is the same knot as any other  $K_{1,i}$ .*



**Figure 2.7:**  $K_{1,i}$ .

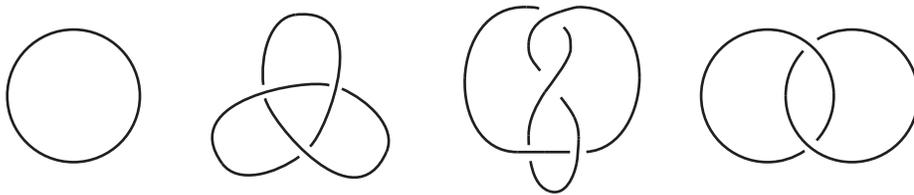


**Figure 2.8:**  $K_{2,i}$ .

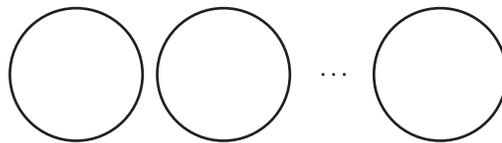
To close out this discussion of link projections, let us consider the link diagrams given in Figures 2.9, 2.10, and 2.11. We will be referring back to these links frequently

throughout the remainder of the thesis. The link in Figure 2.10 is the so-called **trivial link** – it can be projected so that there are no crossings. In general, we are interested in knots with some amount of complexity, commonly referred to as **non-trivial links**.

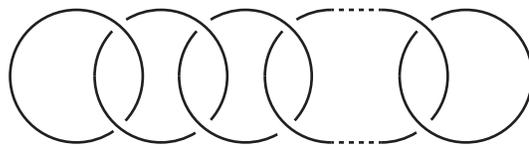
Finally, note that the  $n$ -component Hopf link is basically a chain of circles. We can form it by attaching  $n$  unknots such that successive circles are looped around the bottom of the previous circle. The  $n$ -component Hopf link is a surprisingly good example of a non-trivial link, as we will see at various points throughout our discussion of knot theory.



**Figure 2.9:** Unknot, trefoil knot, figure-eight knot, and Hopf link (left to right)[1].



**Figure 2.10:** Unlink (trivial link) of  $n$  components [1].



**Figure 2.11:** The general form of the  $n$ -component Hopf link.

## 2.2 LINK EQUIVALENCE

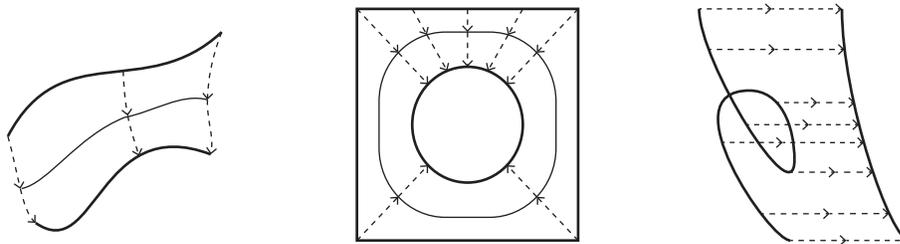
In order to understand what it means for two links to be the same, we want to define maps between the two links. To preserve the knottedness of the links, we will need a specialized type of function.

**Definition 2.4** [2] A **homotopy** of a space  $X \subset Y$  is a continuous map  $h : X \times [0, 1] \rightarrow Y$  such that  $h(X, 0) = X$  and, for all  $t \in [0, 1]$ , the restriction  $h_t : X \rightarrow Y$  is continuous where  $h_t(x) = h(x, t)$ .

The homotopy  $h$  is called an **isotopy** if, for all  $t \in [0, 1]$ ,  $h_t$  is injective.

From the definition, it is not clear what a homotopy looks like. Intuitively, they allow us to continuously deform spaces, treating the space as stretchable rubber. Thus homotopies allow us to stretch, compress, or move the space around.

**Example 2.3** In Figure 2.12, we have three different types of homotopies. While it may be confusing at first glance, let us think of the thicker curves as  $h_0$  and  $h_1$ . Then, as  $t$  changes, we follow the arrows of the thinner lines to obtain the image of a given point. In this way, we can deform a curve into another curve (as in the left-most image), shrink and curve a square into a circle, and squish a looped curve into a simple curve.

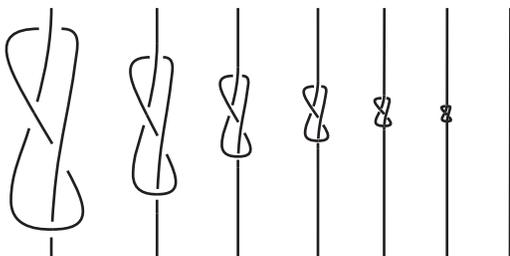


**Figure 2.12:** Three examples of homotopic curves.

An isotopy is a more restrictive form of a homotopy. For instance, the right-most homotopy in Figure 2.12 is not an isotopy because we have mapped multiple

points to the same point of the final curve to eliminate the loop. However, the other homotopies in Figure 2.12 are actually isotopies because every point has a unique image. Unfortunately, an isotopy is not strong enough to preserve knottedness of links.

**Example 2.4** [2] *Let us suppose that we are given the looped piece of string in the left-most image of Figure 2.13. Our isotopy will effectively take the ends of this string and continuously pull the string tighter and tighter. We know that this is an isotopy because every point has an obvious image as the knot becomes tighter – the compressed version of itself. However, our string is a single point thick so we can continue tightening until there is no distance between the different parts of the knot, removing the knot entirely, and leaving us with an unknotted piece of string.*



**Figure 2.13:** Bachelor's unknotting.

It may seem confusing that Figure 2.13 is an isotopy while the third example in Figure 2.12 is only a homotopy. However, notice that we are working in two different spaces: Bachelor's unknotting is a map of a simple loop in  $\mathbb{R}^3$  whereas the map of the curve in Figure 2.12 takes place in  $\mathbb{R}^2$  with a non-simple curve.

The reason that Example 2.4 does not preserve knottedness is that an isotopy only imposes restrictions on the link and not the space the link sits in. To fix this, we require a stricter type of isotopy.

**Definition 2.5** [2, 3] *Two links,  $L_1$  and  $L_2$ , are **ambient isotopic** if there is an isotopy, called an **ambient isotopy**,  $h : \mathbb{R}^3 \times [0, 1] \rightarrow \mathbb{R}^3$  such that  $h(L_1, 0) = L_1$  and  $h(L_1, 1) = L_2$ .*

Ambient isotopies ensure that the space the link is sitting in cannot vanish; while we can shrink the ambient space considerably, we cannot eliminate it. This leads to the following definition.

**Definition 2.6** [2] *Two links,  $L$  and  $L'$ , are **equivalent** if  $L$  and  $L'$  can be related by an ambient isotopy.*

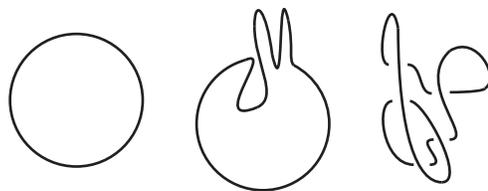
Let us now return to the issue of link projections. Recall that, in the process of projecting a link, we were unsure of whether every link had a link diagram. Specifically, we had not dealt with the possibility that there would always be some point of self-intersection that involved at least three strands of the link. Using ambient isotopies, let us deal with this issue.

Suppose that we have an intersection point of a projected link which involves more than two pieces of string. Applying an ambient isotopy which deforms the strings involved in this self-intersection slightly, we change the projection so that the strings will remain within a small disk around the intersection point. However, since our links are polygonal, there are a finite number of segments involved. Thus this isotopy can be performed such that no more than two strings intersect at a single point. This process may make other crossings apparent, but we have avoided changing the essential knottedness of the link [2, 3]. Thus, as we claimed, we can always project a link onto the plane such that the projection is a link diagram.

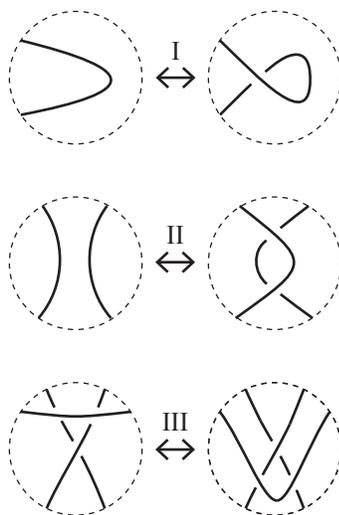
Unfortunately, it turns out that there are many link projections for a given link [1, 2, 5]. Even worse, these projections can appear dramatically different from each other and we may not recognize the link, as in Figure 2.14.

There are some ambient isotopies that are of particular interest to us. Discovered by Kurt Reidemeister in 1926, these ambient isotopies have come to be called Reidemeister moves [1].

**Definition 2.7** [1] *The **Reidemeister moves** are ambient isotopies which impose local changes on the link diagram of a link as shown in Figure 2.15.*

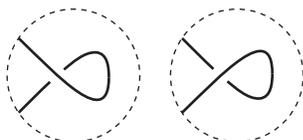


**Figure 2.14:** Three projections of the unknot.



**Figure 2.15:** The three Reidemeister moves.

To appropriately examine the Reidemeister moves, we will be extrapolating the ambient isotopy from the effect it has on the link projection. A Reidemeister I move takes a single continuous segment of string in the projection and either twists or untwists the segment. In 3-space, this corresponds to twisting or untwisting the appropriate segment of the link. Note that we have not specified which way the strand has been twisted, and so the twists in Figure 2.16 are both the result of a Reidemeister I move on a link.

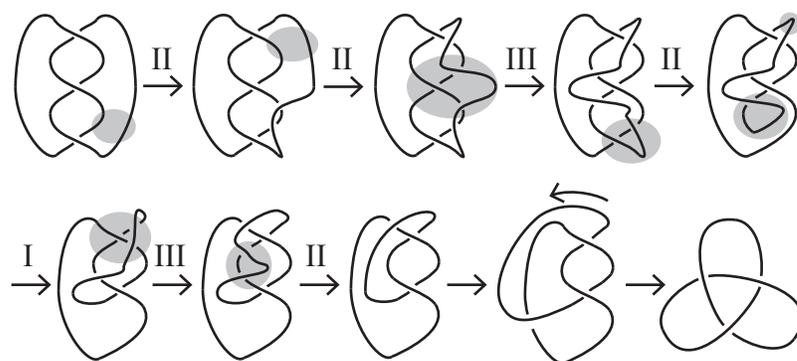


**Figure 2.16:** Different results from a Reidemeister I move.

A Reidemeister II move either takes a strand and pulls it on top of a neighboring strand or reverses this operation. In 3-space, this move corresponds to stretching or compressing the appropriate strand so that it seems to be in front of the other strand. If the strands are contained in the same plane in 3-space, then, to avoid intersecting the other strand, the move will also pull the strand closer to the camera.

The last type of move is the Reidemeister III which takes a strand overcrossing (undercrossing) both of the other involved strands and pulls it over (under) a crossing involving those strands. In 3-space, this corresponds to pushing down on the strand, leaving the other strands unaffected and ensuring that the result is still a link projection. Note that Figure 2.15 only shows a strand which is overcrossing the other strands, although our description makes it clear that there is a similar move for one strand undercrossing the other two.

As we can see in Figure 2.17, the three Reidemeister moves allow us to show the equivalence of at least some projections of the same link. In 1926, Reidemeister proved a result that is likely responsible for advancing knot theory research more than almost any other result [1].



**Figure 2.17:** Equivalence of two trefoil knot projections.

**Theorem 2.1** (*Reidemeister's Theorem*)[1] *Any two projections of a link are related*

*by a finite sequence of the three Reidemeister moves in conjunction with ambient isotopies that do not affect crossings within the projection.*

Reidemeister's Theorem is what allows us to work with the link diagrams rather than the links themselves. Basically, we only need to remember three different ambient isotopies and they will allow us to relate any two projections of a link. As a consequence, whenever we mention a link, we could be referring to either the link in 3-space or the link projection in the plane. If it is necessary, we will emphasize which of these we are using, but it will generally be clear from the context.

## 2.3 LINK INVARIANTS

Using Reidemeister's Theorem, we can concretely describe the properties of links that we alluded to earlier.

**Definition 2.8** [1, 2] *A **link invariant** is a property of a link that does not depend on the projection of the link. Equivalently, it is a property of a link projection that does not change under any of the Reidemeister moves.*

Link invariants are ultimately what allow us to prove that there is more than one type of link. If we can show that a property does not change under any of the three Reidemeister moves, then it must be a link invariant. So if we find two links which differ on this property, it follows that the two links must be different. However, the converse is not necessarily true; if two links have the same property, it does not necessarily follow that they are the same link. To date, no one has discovered an invariant that accurately distinguishes between all links and it is likely that no such universal property exists [1].

We now introduce two classical link invariants so that we can familiarize ourselves with different ways of thinking about links. Throughout the remainder of the thesis, we will be adding to our collection of link properties. Whereas each of the later

invariants requires additional tools, the following invariants are elementary and give us a chance to use our present knowledge.

### 2.3.1 CROSSING NUMBER

The first invariant we discuss is called the crossing number. We know that every link projection has some number of crossings. As it turns out, link projections necessarily have a finite number of crossings. While we can show this quite easily, it is an important observation about how we can handle links and a number of knot theory texts have seemingly assumed it.

**Proposition 2.1** *Every link projection has a finite number of crossings.*

*Proof.* Let  $L$  be a link. Then, by Definition 2.3, every component of  $L$  consists of a finite number of line segments. Since  $L$  has a finite number of components,  $L$  is formed by a finite number of line segments  $M$ . Since line segments do not cross themselves, each line segment can be involved in at most  $M - 1$  crossings. If we sum the number of crossings in which each line segment is used, then every crossing will be counted twice; once for each of the two involved line segments. Hence  $L$  can have at most  $\frac{M(M-1)}{2}$  crossings.  $\square$

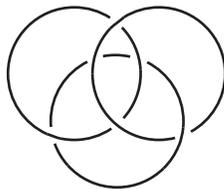
If we take the set of projections of an arbitrary link and extract the number of crossings of the link in each projection, then we obtain a non-empty subset of non-negative integers. Among these, we can easily pick the minimal number of crossings – this is the crossing number of  $L$ .

**Definition 2.9** [1, 2] *The **crossing number** of a link  $L$ , denoted  $c(L)$ , is the minimal number of crossings in any projection of  $L$ .*

We know that the Reidemeister III move does not change the number of crossings in the projection of a link. Unfortunately for us, we could always apply a Reidemeister

I or II and change the number of crossings in any projection we are examining. While one might believe that this invalidates the crossing number as a link invariant, notice that the crossing number is independent of the projection of the link. Hence, while one projection may not realize the crossing number of the link, this minimal number considers all possible projections. Thus the crossing number is truly a link invariant.

For examples of crossing minimal link projections, we can turn back to the diagrams in Figure 2.9. Each one is presented so that it realizes its crossing number. However, the link in Figure 2.18 has a projection with fewer crossings – it is simply a Hopf link and an unknot.



**Figure 2.18:** A link of crossing number 2.

The crossing number of a link is one of the most basic invariants in knot theory. When we start thinking about knottedness, counting crossings is a good place to start. So the crossing number is a wonderful way of categorizing distinct links and, indeed, it is one of the ways that links have been cataloged [1, 2].

One of the difficulties associated with the crossing number is that, even though it may appear that a link projection is as simple as it can get, there may be a series of Reidemeister moves which reduces the number of crossings even further. As such, finding a projection with a given number of crossings only places an upper bound on a link's crossing number. To prove a lower bound, and thus potentially establish the crossing number for the link, requires the use of other techniques.

While there are other interesting lower bounds, we will prove the most basic bound on the crossing number. Despite being simple, it ends up being surprisingly useful and will be referred to later.

**Proposition 2.2** [1] *For a non-trivial knot  $K$ ,  $c(K) \geq 3$ . That is, any knot with two or fewer crossings is trivial.*

*Proof.* Let  $K$  be a knot. If there is a projection of  $K$  with 1 crossing, then we can remove it with a Reidemeister I move and so  $K$  is trivial. So suppose that there is a projection of a knot  $K$  with 2 crossings.

Orient  $K$  and pick one of the crossings, say  $c_1$ . Follow the overcrossing strand in the prescribed direction around the knot until it encounters a crossing. Either this crossing is  $c_1$  or it is the second crossing,  $c_2$ .

In the first case, since  $c_1$  was the first crossing the overstrand encountered, we must have formed a loop. Hence we can remove the crossing with a Reidemeister I move. Then there is a projection of  $K$  with one crossing and, by our earlier argument, this knot is trivial.

Now suppose that the overcrossing strand encountered  $c_2$  instead. Let us instead follow the overcrossing strand in the opposite direction. We claim it cannot encounter  $c_2$  first. If it did, then either we would have formed a complete knot or we found a third crossing. Since we have only two crossings, we formed a complete knot and so there are still unused strands. Thus  $K$  would be a link, but  $K$  was a knot. Hence, in this direction, the overcrossing strand encounters  $c_1$  and we have reduced the problem to the previous case.

Thus all non-trivial knots satisfy  $c(K) \geq 3$ . □

Using the same proof, we can extend a similar result to links.

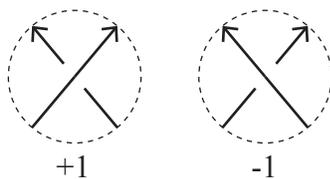
**Corollary 2.1** *Any non-trivial link  $L$  will have  $c(L) \geq 2$ .*

### 2.3.2 LINKING NUMBER

The second link invariant we discuss, the linking number, is substantially easier to compute. Similar to how the crossing number is a good measure of the knottedness of a

link, the linking number is a way of measuring how intricately the various components of a link are linked [1]. First, we need to discuss the notion of assigning a sign to crossings.

**Definition 2.10** [1, 5] *Let  $L$  be an oriented link diagram and let  $c$  be a crossing of  $L$ . Then the **sign** of  $c$  is  $\text{sign}(c) = \pm 1$ , depending on the configuration of  $c$  (Figure 2.19). If  $\text{sign}(c) = 1$  then  $c$  is a **positive crossing**. Otherwise  $c$  is a **negative crossing**.*



**Figure 2.19:** Positive crossing (left) and negative crossing (right).

The difficult part of assigning a sign to a crossing is remembering the different configurations. We can rotate a diagram so that it is clear how the strands are leaving a given crossing, but it is a waste of time to continually refer back to Figure 2.19. Instead, we can remember the configurations by a simple method [1].

Consider the overcrossing strand and mentally rotate it so that the orientation matches the undercrossing strand. We can think of the crossing being positive or negative depending on the rotation. If the rotation required a counterclockwise twist, then the crossing is positive. Otherwise the rotation involved a clockwise twist and so it is a negative crossing. Underlying this method is the assumption that we choose the rotation with the minimal rotation angle, but it is usually clear which rotation is correct.

Now we are ready to define the linking number. To simplify the definition as other knot theorists have done, we will break the concept of the linking number into two definitions. The first will deal with two components of a link while the second definition will permit links with arbitrarily-large multiplicity.

**Definition 2.11** [2, 5] Let  $L = \bigcup_{i=1}^n K_i$  be an oriented link. For  $i \neq j$ , let  $c \in K_i \cap K_j$  denote a crossing  $c$  involving a strand from both  $K_i$  and  $K_j$ . Then the **linking number** of  $K_i$  and  $K_j$ , denoted by  $lk(K_i, K_j)$ , is

$$lk(K_i, K_j) = \frac{1}{2} \sum_{c \in K_i \cap K_j} \text{sign}(c).$$

As alluded to already, the linking number takes two components of a link and counts the twists between those components [1]. Consequently, the linking number is useless when discussing knots, as they only have one component by definition. The generalization of the linking number simply compares every possible pair of components in a link.

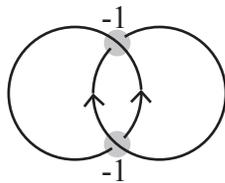
**Definition 2.12** [2, 5] Let  $L = \bigcup_{i=1}^n K_i$  be an oriented link. Then the **total linking number** of  $L$ , denoted  $lk(L)$ , is

$$lk(L) = \sum_{i < j} lk(K_i, K_j).$$

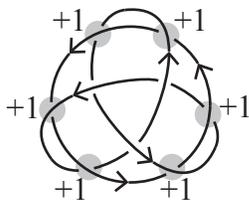
In the case that we are looking at a 2-component link, then the total linking number and the linking number are the same. To solidify our understanding of the linking number, total or otherwise, we should go through a couple of examples.

**Example 2.5** Consider the links in Figures 2.20 and 2.21. The Hopf link has two components and two negative crossings, so the linking number is simply  $\frac{1}{2}(-2) = -1$ . The second figure also has two components, but it has six positive crossings and so it has linking number  $\frac{1}{2}(6) = 3$ . Note that we have ignored the crossings which involve only a single component.

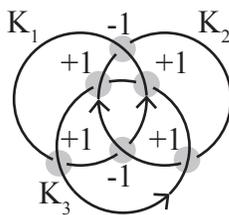
Our third example, depicted in Figure 2.22, is slightly more complicated because it has three components. Since  $lk(K_1, K_2) = -1$ ,  $lk(K_1, K_3) = 1$ , and  $lk(K_2, K_3) = 1$ , the total linking number of the link is 1.



**Figure 2.20:** Hopf Link with linking number  $-1$ .



**Figure 2.21:** Link with linking number  $3$ .



**Figure 2.22:** A 3-component link with total linking number  $1$ .

Notice that the linking number depends on prescribed orientations. If we reverse the orientation of a single component,  $K_i$ , all the inter-component crossings will flip sign, and so the linking number of  $K_i$  and any other component will flip signs as well [1]. So when we start discussing the linking number of a link, we must remember that it is dependent on the chosen orientations for the link components.

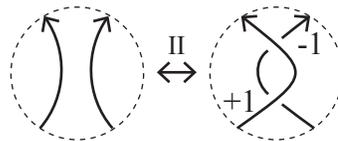
Now that we have a basic understanding of the linking number, we should prove that it is a link invariant.

**Proposition 2.3** [1, 2] *Let  $L$  be an oriented link. For any two distinct components of  $L$ ,  $K$  and  $K'$ ,  $lk(K, K')$  is a link invariant. Moreover,  $lk(L)$  is a link invariant.*

*Proof.* Let  $K$  and  $K'$  be distinct components of an oriented link  $L$ . If we can show that  $lk(K, K')$  is a link invariant then we know that, since  $lk(L)$  is simply the sum over all possible pairs of components,  $lk(L)$  is a link invariant. So let us consider the impact of each of the Reidemeister moves on the computation of  $lk(K, K')$ .

Since a Reidemeister I move involves a crossing of a link component with itself, we know that it will not affect the linking number as we only count crossings involving both  $K$  and  $K'$ . If a Reidemeister II move involves strands from the same component of the link, it will not affect the linking number for the same reason. So let us suppose that the two strands are from different link components.

A Reidemeister II move will either add or remove two crossings. These crossings will have opposite signs as shown in Figure 2.23. This figure covers all possible cases because, as discussed above, reversing the orientation of a strand would change the signs of both crossings. Hence there will be no net change in  $lk(K, K')$ .



**Figure 2.23:** Reidemeister II does not affect linking number.

Reidemeister III moves do nothing to the types of crossings within a link but rather shift them around. Thus the number of crossings of a given sign remains unchanged and thus  $lk(K, K')$  is unaffected by Reidemeister III moves. Hence  $lk(K, K')$  is invariant under the Reidemeister moves and so it is a link invariant. Consequently, as reasoned above,  $lk(L)$  is also a link invariant.  $\square$

Despite what we have shown above, it is not clear how much information the linking number gives. For instance, if there are a differing number of positive and negative signs, it is not clear that the components are necessarily bound together. It could be that the link is a so-called split link.

**Definition 2.13** [5] *A link  $L$  is **splittable** and is called a **split link** if  $\mu(L) \geq 2$  and there is some projection of  $L$  such that there is a disk  $D$  containing at least one component of  $L$  but  $\partial D \cap L = \emptyset$ . Such a projection is a **split projection** of  $L$ .*

*If  $L$  is not a split link, it is a **non-splittable link**.*

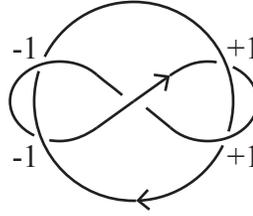
In the case of 2-component links, it is fairly easy to determine that a link is non-splittable.

**Proposition 2.4** [1, 2] *Let  $L$  be a link with  $\mu(L) = 2$ . If  $L$  is splittable then  $lk(L) = 0$ .*

*Proof.* Suppose that  $lk(L) \neq 0$ . Then, in every projection of  $L$ , there is some crossing between the different components of  $L$ . As  $\mu(L) = 2$ , this implies that  $L$  is a non-splittable link.  $\square$

Unfortunately, even for 2-component links, the total linking number being 0 is not sufficient to determine whether the link is splittable or not.

**Example 2.6** [1] *The projection of the Whitehead link in Figure 2.24 has total linking number 0. Using our current techniques, we cannot prove that the Whitehead link is non-splittable, but we can use other techniques to show it later.*



**Figure 2.24:** Oriented Whitehead link.

## 2.4 TYPES OF LINKS

While we have been talking about links in general, there are a variety of classes of links – links created through a similar process or possessing a certain quality. It is not uncommon for knot-theoretic results to be developed for specific classes of links since the similarity adds structure to the link. For our purposes, we need to discuss the notions of amphicheirality and knot composition.

### 2.4.1 AMPHICHEIRAL LINKS

By now, we have likely introduced a sufficient amount of material that it would be nice if we did not complicate matters more than absolutely necessary. Yet this is precisely what we are about to do.

**Definition 2.14** [1] *The **mirror image** of a link  $L$ , denoted  $L^*$ , replaces every crossing in a link diagram with the opposite crossing.*

Suppose we have a link diagram and take its mirror image. The result is the “same” link in some sense. So it seems that the mirror image is the result of projecting the link from a different position in 3-space. This is not the case. For instance, flipping the trefoil knot in Figure 2.9 gives us the exact same knot.

Let us use an oriented link instead. If we tried to simply project the link onto the plane from the opposite side, the orientations of the components would change. Hence we did not obtain the mirror image of the link because only the crossings should have

changed and clearly they do not always switch either. However, it is conceivable that we could use ambient isotopies to solve this problem.

**Definition 2.15** [1] *A link  $L$  is **amphicheiral** if it is ambient isotopic to  $L^*$ . Otherwise,  $L$  is **cheiral**.*

Based on our discussions to this point, it is clear that determining amphicheirality is not an easy business. It seems that the most succinct way of doing so involves the use of knot polynomials, which we will not be introducing until Chapter 5. As such, we instead assert that certain links are amphicheiral while others are cheiral.

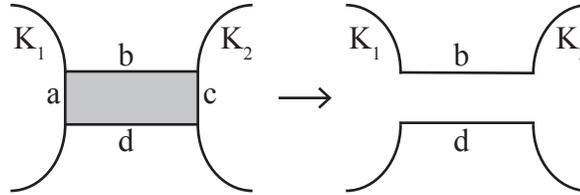
**Example 2.7** *Although we have been talking about “the” trefoil knot, it is a sad state of affairs that we must now shatter that illusion: there are two distinct trefoils. In Figure 2.17, we used the so-called “left-hand trefoil” while the “right-hand trefoil” is depicted in Figure 2.9 [1]. Thus the trefoil knot is cheiral. The only comfort we can take is that these two trefoil knots are clearly mirror images of one another. To be fair to these two knots, unlike many writers who focus on the right-hand trefoil, we will ensure that both are represented.*

**Example 2.8** *The figure-eight knot is amphicheiral [1].*

While it may be apparent that we have covered amphicheirality rather superficially, the essential component of our discussion was that a link and its mirror image are not necessarily the same thing. As such, we must be careful about whether we are using a link or its mirror image. In particular, some link invariants distinguish between the two and so give different results.

## 2.4.2 COMPOSITE KNOTS

Borrowing from the notion of adding numbers, we want a way that we can “add” knots. Notice that we are specifically dealing with knots in this case. While we could



**Figure 2.25:** The composition of  $K_1$  and  $K_2$ .

develop a notion of “adding” links, it would work on the level of a single component. Since this is basically the same situation, we will keep matters relatively simple as we introduce a new operation.

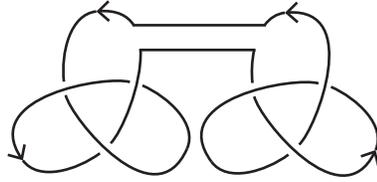
The essential idea behind knot composition is that we take two knots, remove an arc from each knot, and subsequently attach new arcs between the endpoints of the removed arcs. However, while doing so, we must be careful to make sure that we are not complicating either knot in the process. Hence, the new arcs must be close together and not add extra crossings.

**Definition 2.16** [1, 2] Let  $K_1$  and  $K_2$  be knots. A **composition** or **connected sum** of  $K_1$  and  $K_2$ , denoted  $K_1 \# K_2$ , is obtained as follows: take a rectangular disk  $D$  whose boundary is the arcs  $a, b, c, d$  and attach it to  $K_1$  and  $K_2$  such that  $K_1 \cap a = a$ ,  $K_2 \cap c = c$ , and  $(D \setminus (a \cup c)) \cap (K_1 \cup K_2) = \emptyset$  as in Figure 2.25. Then  $K_1 \# K_2 = (K_1 \setminus a) \cup (K_2 \setminus c) \cup b \cup d$ .

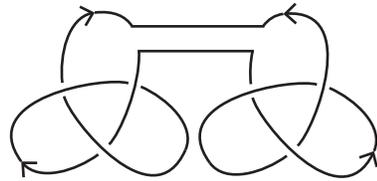
$K_1$  and  $K_2$  are called **factor knots** and  $K_1 \# K_2$  is a **composite knot**.

One important note about knot composition is that it is not necessarily unique [1]. If the knots are unoriented, then it does not matter how we compose the knots because we can imagine shrinking one knot down and having it follow the other knot to any other arc [1]. However, if the knots are oriented, then they can be composed such that their orientations match or, possibly, so that they do not match as in Figures 2.26 and 2.27.

It may be apparent that there is an inverse to the composition operation. While



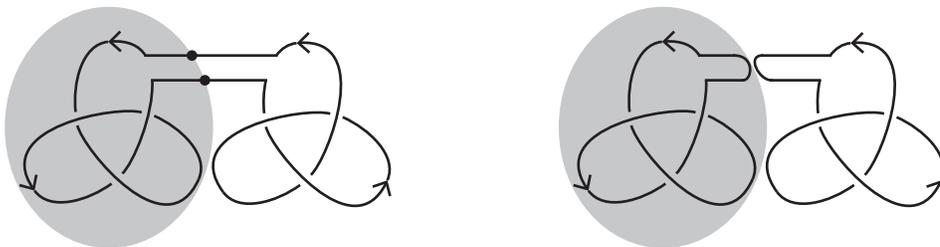
**Figure 2.26:** Composition of two trefoil knots whose orientations match.



**Figure 2.27:** Composition of two trefoil knots whose orientations do not match.

it is not always easy to see, composite knots can always be decomposed into their respective factor knots [1, 2]. In the plane, this involves finding a projection such that one factor can be contained in a disk whose boundary intersects the composite knot at precisely two points. Cutting along the disk boundary, we can reform the factor knots by attaching arcs as in Figure 2.28.

For our purposes, this is a sufficient discussion of knot composition. While there are interesting results and conjectures surrounding composite knots, we only require a basic understanding of the operation and its inverse.



**Figure 2.28:** Decomposition of a composite knot.

## SURFACES

At first glance, the present chapter may seem unrelated to knot theory. While we will briefly explore the possibility of links being projected onto spaces other than  $\mathbb{R}^2$ , this will not give us any definitively useful tools for differentiating between links. Yet discussing surfaces brings us closer to one of the original interests in knot theory: using links, usually knots, to categorize a special class of topological spaces called manifolds. We will not be investing in this application of knot theory, as we are more interested in the knots themselves, but there is certain information about knots that can be understood from surfaces.

In order to properly define a link invariant that uses surfaces, we first require an introduction to surfaces both with and without boundary. Then we will build various properties of surfaces, including genus and Euler characteristic. Finally, we will conclude with a useful connection between links and surfaces that we will employ in Chapters 4 and 5. Unlike some of our previous topics, we will apply the topology-famous “hand-waving” proof technique frequently so that we can focus on knot-theoretic results.

If we are going to make an attempt to understand surfaces, we should invest some time into a proper definition.

**Definition 3.1** [2] *An  $n$ -**manifold** is a topological space  $(X, \mathcal{T})$  such that*

1.  $X$  is Hausdorff;

2.  $\mathcal{T}$  has a countable basis;
3. For all  $x \in X$ , there is a neighborhood of  $x$  homeomorphic to  $\mathbb{R}^n$ .

A manifold is **closed** if  $X$  is compact and  $\partial X = \emptyset$ .

In essence, an  $n$ -manifold is locally similar to  $n$ -dimensional Euclidean space. As such,  $\mathbb{R}^n$  is a simple example of an  $n$ -manifold. Another commonly described  $n$ -manifold is the  $n$ -sphere,  $S^n = \{x \in \mathbb{R}^{n+1} \mid |x| = 1\}$  [2, 6]. We are actually quite familiar with the  $n$ -sphere when  $n = 1$  and  $n = 2$ , as we can see in Figure 3.1.



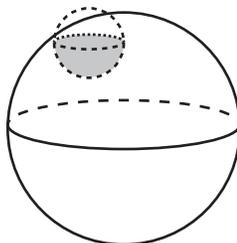
**Figure 3.1:**  $S^1$  (left) and  $S^2$  (right).

It is reasonable to question why the  $n$ -sphere is considered a manifold. If we think of the  $n$ -sphere sitting in  $\mathbb{R}^{n+1}$ ,  $S^n$  would inherit the Hausdorff property because  $S^n$  would be a subspace of  $\mathbb{R}^{n+1}$ . Similarly,  $S^n$  would also necessarily have a countable basis because we could take a countable basis for  $\mathbb{R}^{n+1}$  and intersect each element with  $S^n$ .

The final requirement in Definition 3.1 also holds for  $S^n$ . Although we can handle this situation in general, we will work through the process in the specific case of  $n = 2$ . Let us visualize what happens if we take an open ball of radius  $\varepsilon \in (0, 1)$  centered at a point on  $S^2$  and intersect it with  $S^2$ . Since the ball will touch less than half of  $S^2$ , the intersection will appear as an open disk on  $S^2$  (see Figure 3.2). Since an open disk is homeomorphic to  $\mathbb{R}^2$ , this disk in  $S^2$  will be as well. Thus  $S^n$  is a manifold.

Despite this brief and somewhat crude introduction to manifolds, we will “get on with it” and continue with a definition of surfaces.

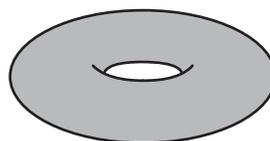
**Definition 3.2** [2] *A surface with no boundary is a closed 2-manifold.*



**Figure 3.2:** Intersection of a ball with  $S^2$ .

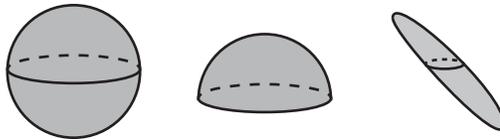
The compactness requirement imposes a sort of “finiteness” condition on the topological space [1]. For instance, one can only travel in one direction on  $S^1$  or  $S^2$  for a limited amount of time, assuming a constant rate, before returning to the point of origin and so  $S^n$  is compact. Moreover, no point of  $\mathbb{R}^{n+1} \setminus S^n$  is in the boundary of  $S^n$  because there is always a small ball that misses  $S^n$ . Hence  $\partial S^n = \emptyset$  and so  $S^n$  is a surface without boundary. Meanwhile  $\mathbb{R}^2$  is not compact and so it is not a surface by our definition.

Another standard surface example is the torus, shown in Figure 3.3. A popular way to think of it is as the frosting on a glazed doughnut [1]. While it has been referred to as a doughnut itself, this is misleading because an actual doughnut has definitive thickness (although this is also true of certain frostings).



**Figure 3.3:** Torus.

Similar to having different projections of links, equivalent surfaces are related by isotopies [1]. In this case, this visually represents stretching or collapsing portions of the surface while it sits in space. For instance, the three surfaces in Figure 3.4 are equivalent.



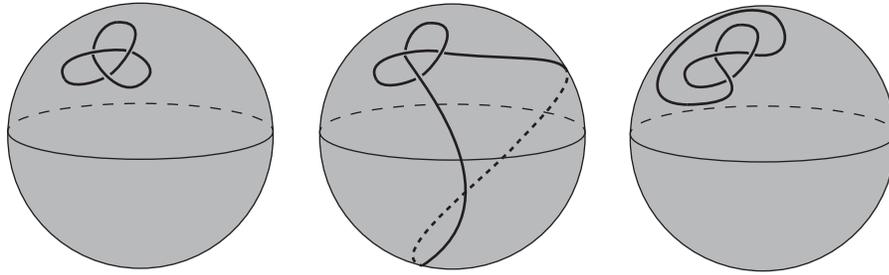
**Figure 3.4:** Isotopic representations of  $S^2$ .

### 3.1 LINKS ON SURFACES

As a short digression, it is useful to discuss how surfaces are immediately applicable to knot theory. In our definition of link projections, we assumed that we were projecting onto the plane. However, it is sufficient to project links onto spaces which are similar to the plane. For instance, since an open disk and the plane are homeomorphic, we could project a link onto the interior of the disk. While this is reasonable, it does not give us any additional information about the link nor does it simplify any problems.

Let us instead consider  $S^2$ . By projecting onto the sphere instead of the plane, the local image of the links are almost the same. The one distinction that needs to be made is that we can ambient isotope a strand around the back of  $S^2$ , as shown in Figure 3.5. Although there is a sequence of Reidemeister moves which can perform an equivalent change in the plane, it is significantly simpler to justify the equivalence on  $S^2$ . To convince ourselves of this fact, we can take an open disk neighborhood of the link on  $S^2$  and project it to the plane, maintaining the link diagram in the process. If we do this both before and after isotoping the strand, then we have two link projections of the same link. Hence, by Theorem 2.1, such a sequence of Reidemeister moves must exist in the plane.

We will be returning to the idea of placing links on  $S^2$  in Chapter 6, when it will allow us a conceptually simple proof of a very significant result about links.



**Figure 3.5:** Strand pulled around the sphere.

## 3.2 SURFACES WITH BOUNDARY

Up-to-now, we have only been looking at examples of surfaces with no boundary. To fully describe surfaces with boundary, we would have to wander deeper into manifold theory. Luckily for us, it is equivalent to start with a surface without boundary and to turn it into a surface with boundary [4]. Naturally, this is built on proven results about surfaces which we are blatantly passing over, but it speeds up the process significantly.

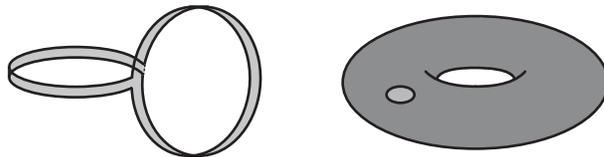
**Definition 3.3** [1] *Let  $D$  be an open disk contained in a surface  $S$ . In  $S \setminus D$ ,  $\partial D \subset \partial S$  and  $\partial D$  is called a **boundary component** of  $S \setminus D$ .*

So if we take a surface with no boundary and remove some open disk from it, we are left with a non-compact space, and so it is not precisely a surface anymore. Yet it is close to being a surface; we would only have to replace the lost disk. The boundary component then becomes a marker of where the disk used to be. Repeating this process a finite number of times, we obtain a surface with boundary.

**Definition 3.4** [1, 2] *A **surface with boundary**  $S$  is homeomorphic to a closed 2-manifold  $\tilde{S}$  with some number of disks removed. The number of boundary components of  $S$  is denoted by  $|\partial S|$  and  $\tilde{S}$  is called the **capped-off surface** of  $S$ .*

It is not always easy to tell whether a space is a surface with boundary. For instance, the spaces in Figure 3.6 are surfaces with boundary. In fact, they are isotopic.

However, it is significantly easier to tell from the image on the right that they are a surface with a single boundary component.



**Figure 3.6:** Surfaces with boundary.

Obtaining  $\tilde{S}$  from a surface with boundary  $S$  is relatively easy. After identifying the boundary components of  $S$ , we only need to replace the disks they bound [1]. It may seem as though we are going in circles, but this is what allows us to conclude that we can always obtain a capped-off surface if we are given a surface with boundary.

### 3.3 ORIENTED SURFACES

Similar to when we introduced links, we want to establish a notion of oriented surfaces. However the equivalent notion is not as obvious as it was for links. Unfortunately, there is more to it than simply following the surface in one direction to assign an orientation – sometimes such a thing is not possible! For the sake of simplicity, we will skip to a mathematically-imprecise but equivalent and easily applied method of determining whether a surface can be oriented.

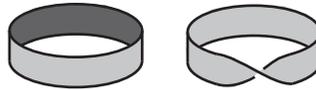
**Definition 3.5** [1] *A surface is **orientable** if, for every open disk contained in the surface, there is an assignment of color – black or white – to each side of the disk such that every point of the interior is colored and overlapping disks assign the same colors to interior points. If such an assignment is impossible, the surface is **nonorientable**.*

Despite all the rigor seemingly involved with our definition, it is imprecise because of the notion of different “sides” of a disk. Certainly, given a physical disk, it would be clear what is meant by sides, but mathematical disks do not have a thickness. Yet we

can still imagine painting mathematical disks in a similar fashion and we will pretend that it holds up mathematically.

To restate our definition in an even more informal manner: an orientable surface is one which has two sides. Most of the surfaces we will use are orientable, but we should be able to distinguish between the two types to better understand the object we are using.

**Example 3.1** *While the torus and sphere have two sides, and thus are orientable, it is harder to visually depict this orientability. In contrast, consider the surfaces with boundary shown in Figure 3.7. It is clear that we can orient the band. However, we cannot orient the second surface, which is a band with a twist in it known as a Möbius band. The Möbius band is a classical example of a non-orientable surface.*



**Figure 3.7:** Band (left) and Möbius band (right).

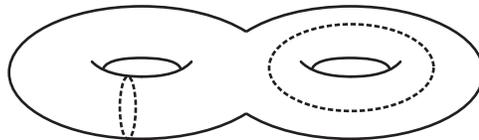
## 3.4 GENUS AND THE CLASSIFICATION OF SURFACES

Despite our apparent claims that there are different surfaces, especially having given distinct names to particular spaces, we have yet to conclusively demonstrate this fact. To do this, we want to develop a topological invariant of surfaces. First, we require another definition.

**Definition 3.6** [2] *A **non-separating loop** of a surface without boundary  $S$  is a simple closed curve  $l$  such that  $S \setminus l$  is connected.*

For an example of non-separating loops, let us consider the surface in Figure 3.8. While there are two different loops presented, we can remove one or both of them and the resulting surface remains connected. A non-example is shown in Figure 3.6 where,

instead of removing the entire disk (which is not a simple closed loop), we would only remove the boundary of the disk. In this case, the interior of the disk would be disconnected from the rest of the torus and so it would be a separating loop instead.



**Figure 3.8:** Double torus with non-separating loops.

Non-separating loops come into play with an important notion of surfaces known as genus.

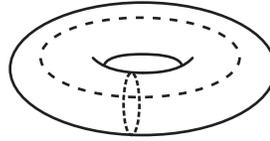
**Definition 3.7** [2] *The **genus** of a connected surface without boundary  $S$ , denoted  $g(S)$ , is the maximum number of disjoint non-separating loops in  $S$  such that their removal does not disconnect  $S$ . If  $S$  has boundary then  $g(S) = g(\tilde{S})$ .*

The notion of surface genus gets at the idea of counting “holes” in the surface. A non-separating loop is necessarily wrapped over one of these holes and, upon removal, reduces the number of holes in the resulting surface by one. We will take it for granted that genus is a topological invariant and instead focus on some examples.

**Example 3.2** *Although it takes some proof to show, it seems clear that removing any simple closed loop from  $S^2$  will disconnect the surface. Hence  $g(S^2) = 0$ . In contrast, we can remove at least two loops from the double torus in Figure 3.8 so its genus is at least 2.*

*While we can remove two loops from the torus as well, as in Figure 3.9, these loops are not disjoint. Instead, it turns out that the genus of the torus is 1.*

As should be clear from our examples above, calculating the genus of a surface can be difficult because we cannot easily determine when we have optimized the number of non-separating loops. However, using genus and tracking boundary components, we can fully describe surfaces.



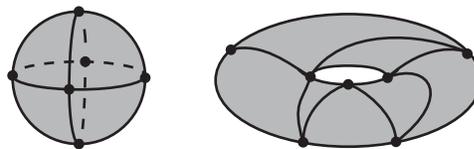
**Figure 3.9:** Intersecting non-separating loops on the torus.

**Theorem 3.1** (CLASSIFICATION OF SURFACES): [2] *Two connected surfaces are homeomorphic if and only if they have the same number of boundary components and the same genus.*

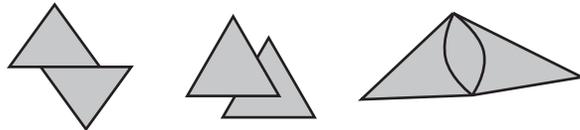
### 3.5 EULER CHARACTERISTIC

Euler characteristic is another surface invariant which, unlike genus, is relatively simple to compute. More significantly, it is easily relatable to genus and so determining genus is simply a task of computing Euler characteristic. Given that this relationship exists, it becomes a wonder that we bothered introducing genus at all. However, this approach gives genus its appropriate motivation rather than being a consequence of Euler characteristic.

Before we can define the Euler characteristic of a surface, we need to introduce surface triangulation. Simply put, a **triangulation** is a division of a surface into a finite number of triangles, including their boundaries, such that adjoining triangles meet at a vertex or share an edge [1]. Note that each triangle has an interior homeomorphic to a disk, which we will synonymously refer to as a face. Example triangulations are shown in Figure 3.10, while non-triangulations are shown in Figure 3.11.



**Figure 3.10:** Triangulations of the sphere and the top half of the torus.



**Figure 3.11:** Non-triangulations of a surface.

Supposing that we can triangulate a surface, then the Euler characteristic is nicely defined.

**Definition 3.8** [1, 2, 4] Let  $S$  be a triangulated surface with  $V$  vertices,  $E$  edges, and  $F$  faces. Then the **Euler characteristic** of  $S$  is

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

**Example 3.3** Let us compute the Euler characteristic of the surfaces in Figure 3.10. This triangulation of the sphere uses 8 faces, 6 vertices, and 12 edges. So  $\chi(S^2) = 6 - 12 + 8 = 2$ .

It is clear that we can mirror the triangulation of the top half of the torus onto the bottom of the torus. By our choice of vertices, doing so will add no additional vertices. So there are 7 vertices. The number of faces for the whole torus will double, giving us a total of 14 faces. Carefully counting the edges, double counting the 7 edges that are not shared by the top and bottom halves, we can see there are 21 edges. Thus the torus has Euler characteristic  $7 - 21 + 14 = 0$ .

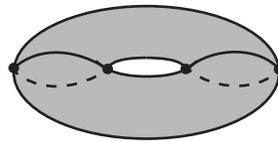
To affirm the usefulness of Euler characteristic as a measure, we require the following statements:

**Theorem 3.2** [1] Every surface can be triangulated.

**Theorem 3.3** [1, 2, 4] Euler characteristic is independent of the triangulation of a given surface and so it is a topological invariant.

As it stands, there is only one complicated aspect of the Euler characteristic computation: keeping track of all the different triangles. To alleviate this concern,

it turns out that triangulation is equivalent to dividing a surface into polygons [1]. So we can divide a surface into vertices, edges, and faces such that the faces have a polygonal boundary and are homeomorphic to a disk. Then we can compute Euler characteristic using this “triangulation.” For instance, in this way, Figure 3.12 is a triangulation of the torus. Note that it is clear that there are 4 vertices, 4 faces, and 8 edges in this triangulation – a much simpler calculation to show that the torus has Euler characteristic 0.



**Figure 3.12:** Simpler torus triangulation.

Finally, we can relate the genus of a surface to its Euler characteristic.

**Theorem 3.4** [1, 4] *For an orientable connected surface  $S$ ,*

$$2g(S) = 2 - \chi(S) - |\partial S|.$$

Note that Theorem 3.4 only applies to orientable surfaces. While a similar result is known for nonorientable surfaces, our interest is primarily directed toward orientable surfaces. Using this powerful tool, we can easily demonstrate that we were correct in our earlier assessments of genus.

**Example 3.4** *We know that the torus has Euler characteristic 0. So, by Theorem 3.4, it has genus 1 as we suggested in Example 3.2. Moreover, since  $\chi(S^2) = 2$ , we also confirm that  $g(S^2) = 0$  by Theorem 3.4.*

### 3.6 SEIFERT SURFACES

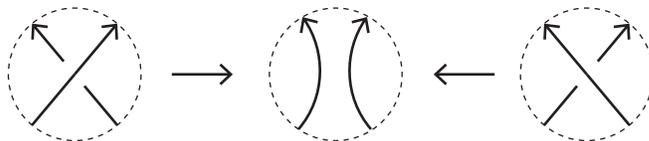
At last we are able to turn our attention back to links, although the build-up was essential. The foundational concept behind a few invariants we will be introducing is that links can serve as the boundaries of surfaces which we call Seifert surfaces.

**Definition 3.9** [1] *Let  $L$  be a link. Then a **Seifert surface** of  $L$  is a connected orientable surface with boundary  $L$ .*

At this point, we should show that every link has at least one Seifert surface. Otherwise, no link invariant based off of Seifert surfaces would be possible and so this concept would have little use. Luckily, this fact was proven by Pontrjagin and Frankl in 1930 [5]. However, in 1934, Herbert Seifert developed an algorithm, known modernly as Seifert's algorithm, that accomplishes the same goal in a simpler fashion [1]. So we will use his method instead.

**Theorem 3.5** [1, 2] *Given a link  $L$ , there is an orientable surface whose boundary is  $L$ .*

*Proof.* Consider a link diagram of  $L$  in the  $xy$ -plane of  $\mathbb{R}^3$ . If  $L$  is unoriented, orient each component of the link. For each crossing in the diagram, we will remove crossings by **smoothing** them as shown in Figure 3.13.



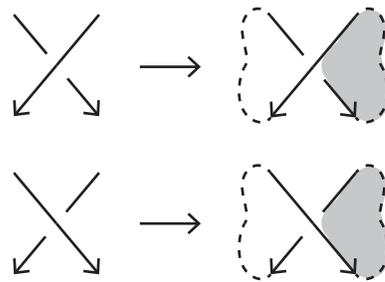
**Figure 3.13:** Smoothing of a crossing.

This operation preserves the prescribed orientation from the original link diagram. The result will be a set of disjoint oriented circles in the plane, which we call **Seifert circles**. Since these circles may be nested within other Seifert circles, we want to shift the circles so that they are no longer nested. This can be accomplished by moving the

circles up in the  $z$ -direction until they are separate from each other. For each Seifert circle, take a bounded disk and let the Seifert circle be its boundary. Now we have a surface with boundary in  $\mathbb{R}^3$ .

Viewing the surface by looking down the  $z$ -axis, we still have the same diagram that we had in the  $xy$ -plane after removing crossings. Let us color each disk such that the side facing us is gray if the disk has a counterclockwise orientation and white otherwise. Then we color the bottoms of the disks by using the opposite color. Thus our surface is orientable.

Let us consider the locations where we removed crossings. Between the two involved Seifert circles, attach a twisted band. The twisting of this band will depend on the original crossing, as shown in Figure 3.14.



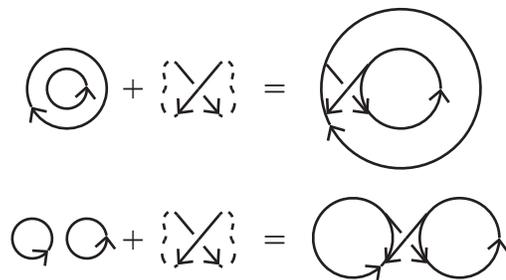
**Figure 3.14:** Twisted bands corresponding to different crossings.

We will now show that the addition of these bands will not affect the orientability of our surface. Consider any two Seifert circles which formerly shared a crossing. Either they were originally nested in the plane or they were not.

If the Seifert circles were nested in the plane, then they must have the same orientation. Otherwise there would be an inconsistency in orientation as shown in the top of Figure 3.15. So, by our earlier coloring, they have the same color on the upper face. Since these circles are nested, the twisted band will have to bend to connect the disks. Accounting for the half-twist in the band and this bend, the colors on the two

disks will match appropriately (an upcoming example will use such bands to connect disks, Figure 3.17).

Similarly, if the Seifert circles were not nested, then they have opposite orientations so as to avoid the orientation conflict demonstrated in the bottom of Figure 3.15. By our earlier coloring, this implies that the upper faces of the disks will have opposite colors. Then attaching the twisted band will not affect orientation because the color on the top of one side of the band will flip over to the other color on the other side, matching the upper face color of the other disk.



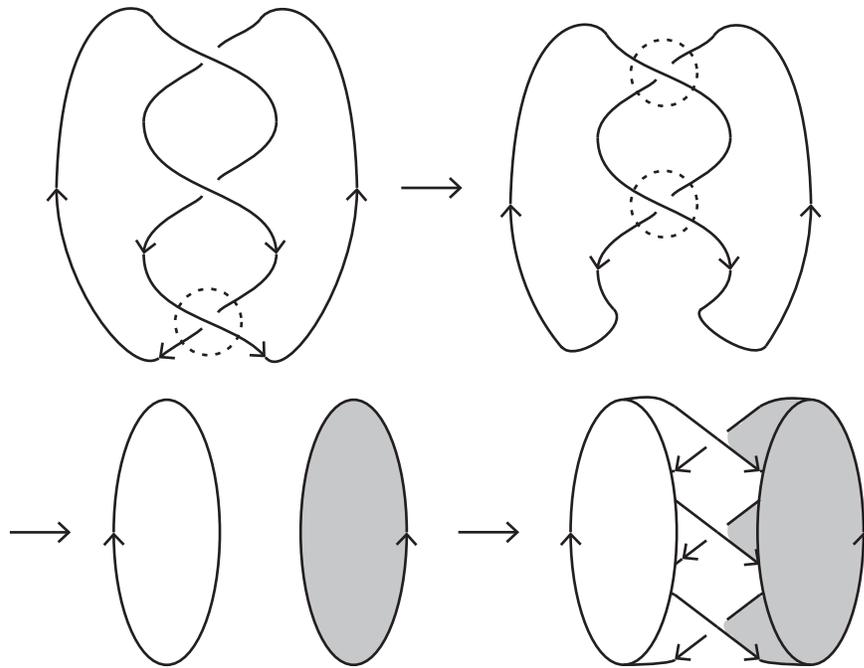
**Figure 3.15:** Impossible Seifert circle orientations.

Thus we have shown that attaching bands will not affect the orientability of the surface and so it will remain orientable. The fact that the boundary of the surface is the original link follows immediately because the twisting of the bands matches the type of crossing found on the link. Thus the surface is an orientable surface whose boundary is the link.  $\square$

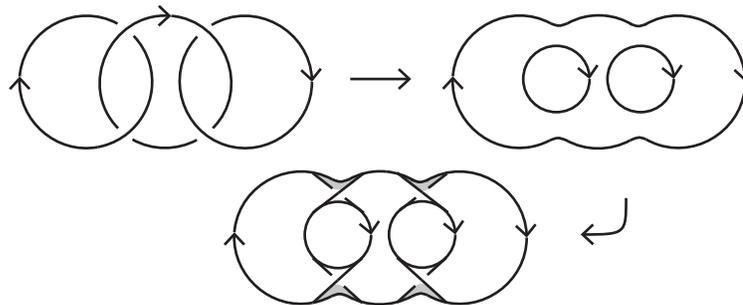
For a concrete application of Seifert's algorithm, let us consider a couple examples.

**Example 3.5** *Let us apply Seifert's algorithm to the projection of the trefoil knot in Figure 3.16 and the 3-component Hopf link in Figure 3.17. The final images in each figure correspond to a Seifert surface for the particular link.*

It is important to note that, for a general link  $L$ , Seifert's algorithm does not always produce a Seifert surface because the resulting surface is not necessarily connected.



**Figure 3.16:** Seifert surface of a trefoil knot.



**Figure 3.17:** Seifert surface of a 3-component Hopf link.

In particular, if  $L$  is a split link, then there is some projection of  $L$  such that some component of  $L$  is separated from the rest of  $L$  by a disk. Applying Seifert’s algorithm to this projection, we obtain disconnected surfaces and so the resulting surface for  $L$  is disconnected. One remedy for this problem is the following:

**Corollary 3.1** [2] *Let  $L$  be a non-split projection of a link. Then Seifert’s algorithm produces a Seifert surface of  $L$ .*

Using Seifert surfaces, we may define our first new link invariant.

### 3.7 LINK GENUS

We have previously discussed the concept of the genus of a surface. By Theorem 3.5, we know that every link has at least one Seifert surface and so the following is a well-defined link invariant:

**Definition 3.10** [2] *The **genus** of an oriented link  $L$ , denoted by  $g(L)$ , is the minimal genus of any Seifert surface of  $L$ . In the case of an unoriented link, the genus is the minimal genus of a Seifert surface over all possible orientations of that link.*

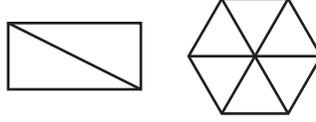
As with many link invariants that we have seen, it can be difficult to determine whether a given Seifert surface for a link  $L$  realizes  $g(L)$ . In fact, it turns out that Seifert's algorithm is not always optimal. In other words, there are links such that, given any projection, Seifert's algorithm will produce a surface of genus greater than the genus of the link [2]. However, given a Seifert surface, it is not difficult to compute its genus by appealing to the Euler characteristic of the surface as we saw in Theorem 3.4. Using a surface generated through Seifert's algorithm, we are able to prove a simple method of computing the Euler characteristic of a Seifert surface.

**Theorem 3.6** *Let  $F$  be the Seifert surface obtained, via Seifert's algorithm, from a link projection  $L$  with  $s(L)$  Seifert circles and  $k(L)$  crossings. Then the Euler characteristic of  $F$  is given by*

$$\chi(F) = s(L) - k(L).$$

*Proof.* [2] Let us divide the Seifert circles and bands into triangles. To streamline this process, let us divide each band into two pieces and each Seifert circle, on which  $n$  bands are attached, into  $2n$  triangles with a vertex in the center as is shown in Figure 3.18.

Since they replace crossings, we know that each band is twisted and subsequently



**Figure 3.18:** Triangulation of a band and a Seifert circle which has 3 attached bands.

attached to two distinct Seifert circles. As each band has two faces, they contribute  $2k(L)$  faces to the surface. We have a total of  $4k(L)$  faces within the Seifert circles since, based on our triangulation, a crossing creates two faces in each Seifert circle to which it is connected. So there are a total of  $F = 4k(L) + 2k(L) = 6k(L)$  faces.

To count vertices, let us assume that no two bands share a vertex. This is possible because of how we triangulated the Seifert circles. Then it is sufficient to count the number of vertices used in the bands and add the number of Seifert circles. This is because each Seifert circle will only use the vertices from the bands, plus the central vertex. Since each band has 4 vertices, we have  $4k(L)$  vertices plus an additional  $s(L)$  vertices from the Seifert circle center vertices. So we have a total of  $V = 4k(L) + s(L)$  vertices.

Lastly, we need to count edges. Each band contributes 5 edges for a total of  $5k(L)$  edges. Now let us consider the remaining edges in the Seifert circles. There are  $6k(L)$  such edges because the uncounted edges form triangles, as no two bands share a vertex, and there are  $2k(L)$  triangles. Thus there are a total of  $E = 11k(L)$  edges.

Hence  $\chi(F) = V - E + F = (4k(L) + s(L)) - 11k(L) + 6k(L) = s(L) - k(L)$  as desired.  $\square$

**Corollary 3.2** *The genus of a Seifert surface  $F$  constructed from  $L$ , a non-split projection of a link, satisfies*

$$2g(F) = (1 - s(L) + k(L)) + (1 - \mu(L))$$

where  $s(L)$  is the number of Seifert circles of  $L$  and  $k(L)$  is the number of crossings in  $L$ .

*Proof.* By Theorem 3.4 and Theorem 3.6,

$$\begin{aligned} 2g(F) &= 2 - \chi(F) - |\partial F| \\ &= (1 - \chi(F)) + (1 - \mu(L)) \\ &= (1 - s(L) + k(L)) + (1 - \mu(L)). \end{aligned}$$

Thus  $2g(F) = (1 - s(L) + k(L)) + (1 - \mu(L))$  as desired. □

## GRAPH THEORY AND SEIFERT MATRICES

In the following chapter, we use Seifert surfaces to generate a matrix, appropriately called the Seifert matrix. To obtain the Seifert matrix, we will need to familiarize ourselves with graphs and the homology of graphs. While this may seem to be taking us further away from studying links in any capacity, this matrix will be used to generate two new link invariants: signature and determinant. Later on, it will give us a way of developing a powerful third invariant called the Alexander polynomial.

## 4.1 GRAPH THEORY

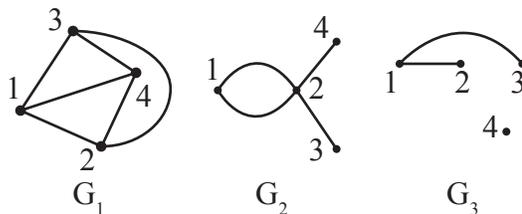
The first step on our path to a veritable wealth of link invariants is understanding the basics of graph theory. Our brief introduction will give us the necessary tools to build a graph from a Seifert surface, which is appropriately called a Seifert graph. While we could define a Seifert graph almost immediately, we will build theory surrounding a general graph first.

**Definition 4.1** [2] *A **graph**  $G$  is composed of a set of vertices  $V(G)$  and a set of edges  $E(G)$ . Every edge  $e \in E(G)$  may be represented as an unordered pair of vertices,  $e = \{v_i, v_j\}$  for some  $v_i, v_j \in V(G)$ .*

It is convention to refer to the vertices involved in an edge as **endpoints**. Note that our definition of graphs allows for the possibility that two vertices may act as

endpoints for multiple edges. From a set-theoretic perspective, this poses a problem. However, in visual representations of a graph, we tend to denote vertices by points and edges by lines between appropriate vertices. Thus, from a visual perspective, it is clear when two vertices share multiple edges.

**Example 4.1** *Let us consider the three examples of graphs shown in Figure 4.1. Each of these graphs contains 4 vertices, labelled 1, 2, 3, or 4. So for each graph,  $V(G_i) = \{1, 2, 3, 4\}$ . However, they have different edge sets. For example,  $E(G_3) = \{\{1, 2\}, \{1, 3\}\}$  and  $E(G_2) = \{\{1, 2\}, \{1, 2\}, \{2, 3\}, \{2, 4\}\}$ .  $G_1$  is a special type of graph, called a complete graph, because it contains one copy of every possible edge between the 4 vertices.*



**Figure 4.1:** Example graphs.

*In the case of  $G_2$ , note that we have a multiple edge. While listing edges in  $E(G_2)$  by the vertices they connect makes sense, we must remember that our edge sets are allowed to contain repeat elements.*

*While the other two graphs involve every vertex in at least one edge, vertex 4 in  $G_3$  does not share an edge with any other vertex in the graph. This shows that, even when a vertex is present in the graph, it is not necessary for it to be the endpoint of any edge.*

We will return to these examples later, but there are other basic graph theoretic concepts to introduce.

**Definition 4.2** [2] *A **subgraph**  $G'$  of a graph  $G$ , denoted  $G' \subseteq G$ , is a graph with  $V(G') \subseteq V(G)$  and  $E(G') \subseteq E(G)$ .*

The notion of a subgraph is a fairly intuitive extension of our definition of a graph. In this case, we start with a graph  $G$  and take only some of its vertices and then some of the edges between those vertices. One trivial example of a subgraph is the graph itself, but there are many non-trivial examples as well. For instance, using the graphs from Example 4.1, we can say that  $G_3 \subseteq G_1$  since every edge of  $G_3$  is contained in  $G_1$ . However,  $G_2 \not\subseteq G_1$  because  $E(G_1)$  has only a single copy of the edge  $\{1, 2\}$ . In general, as noted above, this poses a problem using the standard notion of a subset. So, for our purposes, we will assume that the multiplicity of an edge is also considered when comparing sets for graphs.

**Definition 4.3** [2] *Let  $G$  be a graph. Then a **path** in  $G$  is a sequence of vertices and edges  $v_1e_1v_2e_2 \dots e_{n-1}v_n$  such that  $v_i \in V(G)$  and  $e_i = \{v_i, v_{i+1}\} \in E(G)$ .*

*A **cycle** is a path such that the initial and final vertices are the same. A **simple path** uses a given vertex at most once.*

The visualization of a path is fairly intuitive. Starting at a given vertex, the path follows edges to adjacent vertices and eventually stops. If this stopping location is the same as the initial vertex, then we call the path a cycle. Note that there is nothing in the definition of a path that prohibits revisiting a vertex or using an edge multiple times. However, in simple paths, no vertex (and so no edge) can be repeated.

The next concept to introduce involves the formal definition of a connected graph.

**Definition 4.4** [2] *A graph  $G$  is **connected** if, for any two vertices in  $V(G)$ , there is a path between them. If  $G$  is not connected, then  $G$  is **disconnected**.*

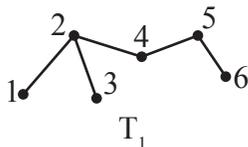
At this point, we can make sense of a special type of graph and a particular subgraph which will be essential in our later discussions.

**Definition 4.5** [2] *A graph  $T$  is called a **tree** if it is connected and it contains no cycles.*

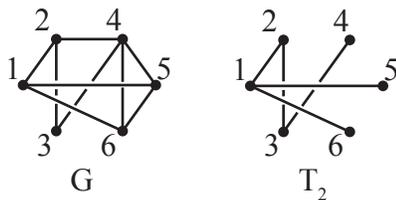
**Definition 4.6** [2] Let  $G$  be a connected graph. Then  $T \subseteq G$  is called a **spanning tree** of  $G$  if  $V(T) = V(G)$  and  $T$  is a tree.

We can think of trees as being minimally-connected graphs. There are sufficient edges in a tree so that the graph is connected, but not so many that we can form a cycle. Let us solidify these notions with a couple of examples.

**Example 4.2** The graphs  $T_1$  and  $T_2$  in Figures 4.2 and 4.3 are trees. Moreover,  $T_2$  is a spanning tree of  $G$ . Indeed,  $T_1$  is also a spanning tree of  $G$ , although this fact is less readily apparent. Thus, in general, spanning trees are not a unique subgraph of a connected graph.



**Figure 4.2:** A tree with 6 vertices.



**Figure 4.3:** A graph and a spanning tree.

While there are many more properties of trees, the following propositions are the most essential to our discussion of Seifert graphs.

**Proposition 4.1** [2] A tree on  $n$  vertices has  $n - 1$  edges.

*Proof.* Let  $T$  be a tree on  $n$  vertices. So  $T$  is connected and  $T$  contains no cycles. Thus every vertex is involved in at least one edge. Suppose we delete an edge,  $e_1 = \{v_1, v_2\}$ , and call the resulting graph  $T_1$ . We claim that  $T_1$  is disconnected.

To the contrary, suppose that  $T_1$  is connected. Then, in particular, there is a path from  $v_1$  to  $v_2$ ,  $P$ . However, in  $T$ , this would mean that  $Pe_1v_1$  is a cycle. Thus  $T_1$  is disconnected and consists of 2 connected components.

Using the same process, deleting an edge of  $T_1$  will create one more component in the graph. Deleting edges one-by-one, we start with a single component in  $T$  and, at the end of this deletion process, we will have  $n$  components because  $T$  has  $n$  vertices. Since each edge deletion adds one component to the graph, we could only have deleted  $n - 1$  edges.  $\square$

**Proposition 4.2** *Let  $T$  be a tree on  $n \geq 2$  vertices. Then there are at least two vertices such that each vertex is an endpoint of exactly one edge.*

*Proof.* Let  $T$  be a tree and let  $P = x_1e_1x_2e_2 \dots e_{n-1}x_n$  be the longest simple path in  $T$ . We claim that the endpoints of  $P$ ,  $x_1$  and  $x_n$ , are the desired vertices. Suppose, without loss of generality, that  $x_1$  is the endpoint of another edge,  $e = \{x_1, y\}$ . Either  $y$  shares an edge with a second vertex in  $P$  or it does not.

If  $y$  shares an edge with a second vertex in  $P$ , say  $x_j$  with  $j \neq 1$ , then we know that  $ye x_1 e_1 \dots e_{j-1} x_j \{x_j, y\} y$  is a cycle in  $T$ . However,  $T$  is a tree and so this is impossible.

If  $y$  does not share an edge with a second vertex in  $P$ , then  $yeP$  is a path in  $T$  which uses no vertex more than once. Moreover, it is a longer path than  $P$ . However,  $P$  was maximal and so this is impossible.

Thus  $x_1$  must be the endpoint of a single edge in  $T$ . An analogous argument applies to  $x_n$ .  $\square$

One reason for introducing these results is to determine the “fullness” of a graph.

**Definition 4.7** [2] *The **rank** of a connected graph  $G$  is given by*

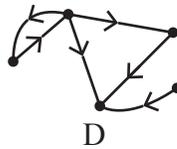
$$\text{rank}(G) = |E(G)| - |V(G)| + 1.$$

To give meaning to the rank of a graph, we know that a connected graph  $G$  has  $|E(G)|$  edges. By Proposition 4.1, a spanning tree of  $G$  has  $|V(G)| - 1$  edges. Then  $\text{rank}(G)$  is the difference of number of edges in  $G$  and the number of edges used in a spanning tree of  $G$ . Basically, as alluded to earlier, rank tells us how well  $G$  is connected.

The final concept that we should introduce before moving onto Seifert graphs and the homology of graphs is the notion of a directed graph.

**Definition 4.8** [2] A **directed graph**  $D$  is a graph in which the set of edges contains ordered pairs of vertices. Hence each  $e \in E(D)$  corresponds to  $(v_i, v_j)$ , an edge pointing from vertex  $v_i$  to vertex  $v_j$ .

**Example 4.3** The graph in Figure 4.4 is an example of a directed graph. In comparison with the representation of a standard graph, we can see that edges of directed graph are nominally assigned an orientation.



**Figure 4.4:** A directed graph.

#### 4.1.1 SEIFERT GRAPHS

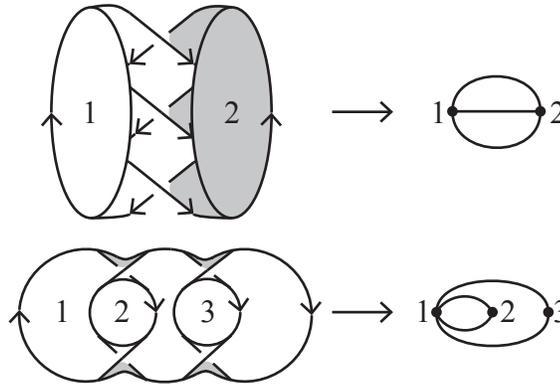
With these basics in mind, we can define the Seifert graph and use it to produce some results regarding the Seifert surface, and thus the link, from which it originated.

**Definition 4.9** [2] Fix a Seifert surface  $F$  generated by Seifert's algorithm. A graph  $G$  is called a **Seifert graph** if a vertex is uniquely associated to each Seifert circle and, for  $v_i, v_j \in V(G)$ ,  $\{v_i, v_j\} \in E(G)$  if and only if the corresponding Seifert circles are connected by a twisted band in  $F$ .

Applying Seifert's algorithm to a link projection, we quickly obtain a Seifert graph by replacing each Seifert circle with a vertex and each smoothed crossing with an edge.

**Example 4.4** *Let us consider the Seifert surfaces from Example 3.5. The trefoil knot Seifert surface has two Seifert circles, so there will be two vertices in its Seifert graph. Between these two Seifert circles, there are three bands and so the Seifert graph will contain three edges between the two vertices, as shown in the top of Figure 4.5.*

*The Seifert surface of the 3-component Hopf link appears more complicated, but we can follow the same process. Each of the three Seifert circles become vertices and each of the bands becomes an edge between the appropriate vertices, as shown in the bottom of Figure 4.5.*



**Figure 4.5:** Seifert graphs of Seifert surfaces from Example 3.5.

As a result, we can think of drawing the Seifert graph of a Seifert surface on the surface itself. More significantly, as we will see, it is useful to use the cycles of a Seifert graph in its corresponding surface.

## 4.2 HOMOLOGY OF GRAPHS

Having built up a diverse collection of graph-theoretic terms and ideas, we are now ready for the homology group of a graph. In particular, the homology group of a

Seifert graph. While the importance of this group is not immediately apparent on its own, it connects back to our discussion of surfaces.

**Theorem 4.1** [2] *The homology groups of a Seifert surface and its Seifert graph are isomorphic.*

The essence of this theorem is that anything we learn about the homology of Seifert graphs provides us with information about the Seifert surface from which it is generated. We will invoke the use of this theorem without proof, although a sketch of the proof can be found in [2].

With Theorem 4.1 in mind, let us start our discussion of graph homology. We begin with the notions of 0-chains and 1-chains.

**Definition 4.10** [2] *Let  $D$  be a connected directed graph with  $V(D) = \{v_1, v_2, \dots, v_k\}$  and  $E(D) = \{e_1, e_2, \dots, e_n\}$ . Then a **0-chain**  $c$  is a linear combination of the vertices of  $D$ ,*

$$c = \sum_{i=1}^k \lambda_i v_i \text{ where } v_i \in V(D) \text{ and } \lambda_i \in \mathbb{Z}.$$

*Similarly, a **1-chain**  $z$  is a linear combination of the edges of  $D$ ,*

$$z = \sum_{i=1}^n \mu_i e_i \text{ where } e_i \in E(D) \text{ and } \mu_i \in \mathbb{Z}.$$

Although it is not necessary for a general 1-chain to visually represent a quality of a graph, note that every path and cycle within a graph has a corresponding 1-chain [2]. We can imagine the coefficient  $\mu_i$  being the number of times we use edge  $e_i$ , with  $-e_i$  representing following the edge in the opposite direction of its orientation [2].

Let us define a binary operation on 1-chains,  $+$ , such that for 1-chains  $z_1 = \sum_{i=1}^n \lambda_i e_i$  and  $z_2 = \sum_{i=1}^n \mu_i e_i$ ,

$$z_1 + z_2 = \sum_{i=1}^n (\lambda_i + \mu_i) e_i.$$

The set of 1-chains, denoted  $C_1(D)$ , along with the operation  $+$  form an abelian group [2]. Using an analogous operation, the set of 0-chains also form an abelian group,  $C_0(D)$  [2].

The next step in our process is to define a group homomorphism between 1-chains and 0-chains.

**Definition 4.11** [2] *Let  $D$  be a directed graph and  $e = (v_i, v_j) \in E(D)$ . The **boundary operator** of  $D$  is the linear group homomorphism  $\partial : C_1(D) \rightarrow C_0(D)$  where  $\partial(e) = v_j - v_i$ .*

Since  $\partial$  is a linear function,  $\partial \left( \sum_{k=1}^n \lambda_k e_k \right) = \sum_{k=1}^n \lambda_k \partial(e_k)$ . Now we are interested in the kernel of this function.

**Definition 4.12** [2] *Let  $D$  be a connected directed graph and  $\partial$  the boundary operator of  $D$ . Then  $z \in \ker(\partial)$  is called a **1-cycle**. The subgroup of 1-cycles is denoted by  $H_1(D)$ .*

We will further distinguish between trivial and non-trivial 1-cycles. A trivial 1-cycle is simply  $z = 0$ , the cycle that involves no edges, while non-trivial 1-cycles are all other cycles in  $H_1(D)$ . We will not be going into great detail regarding homology groups, but it turns out that this group is precisely the first homology group of a graph. However, it would be helpful to have a better handle on the structure of  $H_1(D)$ .

**Definition 4.13** [2] *Let  $D$  be a connected, directed graph. Then a **basis** of  $H_1(D)$  is a set of non-trivial 1-cycles such that each 1-cycle in  $H_1(D)$  can be expressed uniquely as a linear combination of basis elements.*

So if we can find a basis for  $H_1(D)$ , we know what every 1-cycle in the graph will look like. To this end, we will show that there is an algorithm to find a basis for this subgroup. First, we should show that a tree has no 1-cycles. Note that, since trees are connected by definition, this is a legitimate consideration.

**Theorem 4.2** [2] *Let  $T$  be a directed tree. Then every 1-cycle in  $T$  is trivial.*

*Proof.* Let us suppose that  $z$  is a non-trivial 1-cycle in a directed tree  $T$ . We will show that this is impossible by inducting on  $|E(T)| = n$ .

Suppose that  $n = 1$ . Then  $z = \mu_1 e_1$ . Since  $z$  is a 1-cycle,  $\partial(z) = 0$ . However,

$$\partial(z) = \partial(\mu_1 e_1) = \mu_1 \partial(e_1) = \mu_1 (v_2 - v_1) = \mu_1 v_2 - \mu_1 v_1.$$

Thus  $0 = \lambda_1 v_2 - \lambda_1 v_1$  and so  $\lambda_1 = 0$ .

Now suppose that  $n > 1$ . Since some vertex in a tree has to be the endpoint of exactly one edge by Proposition 4.2, let us say, without loss of generality, that  $v_1$  is such a vertex. Let  $e_n$  be the unique edge. Since  $v_1$  has only this edge connecting it to the rest of  $T$ ,  $e_n$  is the only term involving it. Then  $\partial(z) = 0$  and

$$\partial(z) = \partial \left( \sum_{i=1}^n \lambda_i e_i \right) = \sum_{i=1}^n \lambda_i \partial(e_i) = \sum_{i=1}^{n-1} \lambda_i \partial(e_i) + \lambda_n \partial(e_n) = \partial \left( \sum_{i=1}^{n-1} \lambda_i e_i \right) + \lambda_n \partial(e_n).$$

For some  $v_j \in V(T)$  where  $j \neq 1$ , we know that  $\lambda_n \partial(e_n) = \pm \lambda_n (v_j - v_1)$ , depending on the orientation of  $e_n$ . Since the contribution of  $v_1$  must be zero,  $\lambda_n = 0$ . Thus  $z$  is a cycle on the subtree  $T'$  where  $V(T') = V(T) \setminus \{v_1\}$  and  $E(T') = E(T) \setminus \{e_n\}$ . Since  $T'$  is a tree with  $n - 1$  edges, we may apply the induction hypothesis to show that  $z = 0$ .

Hence, by the principle of mathematical induction, we know that any 1-cycle on a tree must be trivial. □

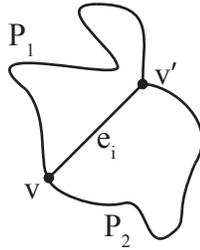
With this result in hand, we can prove that the following algorithm works.

**Theorem 4.3** [2] *Let  $D$  be a connected directed graph and  $T$  be a spanning tree of  $D$ . Label the edges of  $D$   $e_1, e_2, \dots, e_n$  so that the first  $r$  edges are not in  $T$  (i.e.,  $e_i \notin E(T)$  for  $i = 1, 2, \dots, r$ ). Then*

1. For  $i = 1, 2, \dots, r$ , the graph  $T \cup e_i$  contains a unique simple cycle. Let  $z_i$  be the corresponding 1-cycle.
2.  $\{z_1, z_2, \dots, z_r\}$  is a basis for  $H_1(D)$ .

*Proof.* First we will show that  $T \cup e_i$  contains a unique simple cycle for each  $e_i \notin T$ . As  $T$  is a tree, adding any edge will create a cycle. In particular, adding  $e_i = (v, v')$  to the graph creates a cycle. Now suppose that  $T \cup e_i$  contains at least two simple cycles,  $c_1$  and  $c_2$ . Since both cycles use  $e_i$ , both  $c_1$  and  $c_2$  must pass through the endpoints of  $e_i$ . Moreover, since  $c_1$  and  $c_2$  are simple, they use  $e_i$  exactly once.

Let  $c_1 = P_1 e_i v$  and  $c_2 = v e_i P_2$ , where  $P_1$  is a path starting at  $v$  and ending at  $v'$  and  $P_2$  is a path starting at  $v'$  and ending at  $v$  (see Figure 4.6). Then  $P_1 P_2$  is a cycle in  $T$  that does not use  $e_i$ . However  $T$  is a tree and cannot have any cycles by Theorem 4.2. Hence there must be a unique simple cycle in  $T \cup e_i$ . Let us call the corresponding 1-cycle  $z_i$ .



**Figure 4.6:** Two cycles in  $T \cup e_i$ .

If  $D$  is a tree, then a spanning tree of  $D$  is trivially  $D$  itself. So  $r = 0$ . By Theorem 4.2,  $H_1(D)$  is trivial. Thus there is an empty basis for  $H_1(D)$ , so the result is vacuously true.

Now assume that  $D$  is not a tree, implying  $r > 0$ . Let  $z \in H_1(D)$ . To show that  $\{z_1, z_2, \dots, z_r\}$  is a basis for  $H_1(D)$ , we must show that  $z$  can be written uniquely as a linear combination of  $\{z_1, z_2, \dots, z_r\}$ .

Since  $z \in H_1(D)$ , it is a 1-cycle and thus a 1-chain. Hence  $z = \sum_{i=1}^n \lambda_i e_i$ . We claim

that we can write  $z = \sum_{i=1}^r \lambda_i e_i$ .

Consider  $c = z - \sum_{i=1}^r \lambda_i e_i$ . Since the sum of two cycles is a cycle,  $c$  is a cycle in which every edge is in  $E(T)$  by our construction. Hence by Theorem 4.2,  $c$  must be trivial and so  $\lambda_i = 0$  for  $i = r + 1, r + 2, \dots, n$ . Hence  $z = \sum_{i=1}^r \lambda_i e_i$  as desired. Moreover, the coefficients are unique because  $z_i$  is the unique cycle that contains  $e_i$ . Therefore  $\{z_1, z_2, \dots, z_r\}$  is a basis for  $H_1(D)$ .  $\square$

Theorem 4.3 is the most significant result we have involving graph homology. Restating it, we can use the edges not required for a spanning tree of the graph to form a basis for the first homology group of the graph. In other words, for a connected directed graph  $D$ , we know that a basis for  $H_1(D)$  has  $\text{rank}(D)$  elements.

Notice that our entire discussion has rested on directed graphs. However, when we discussed Seifert graphs, we made no mention of how to direct edges. To bring these two concepts together, we can arbitrarily assign an orientation to an edge of the Seifert graph. Visually, the basis cycles will appear the same and so it makes no difference for our purposes. Thus we will assume that our Seifert graphs are directed graphs, even if we do not assign specific orientations.

Applying Theorem 4.1, finding a basis for the homology group of a Seifert surface simply requires a basis for the first homology group of its Seifert graph. In the surface, the 1-cycles from the graph become loops in the surface [2]. The next two results will demonstrate that this basis will contain a predictable number of cycles.

**Theorem 4.4** [2] *Let  $G$  be a Seifert graph of a Seifert surface  $F$  constructed from a link projection  $L$  with  $s(L)$  Seifert circles and  $k(L)$  crossings. Then*

$$\text{rank}(G) = 1 - s(L) + k(L) = 2g(F) + \mu(L) - 1.$$

*Proof.* By Definition 4.7,  $\text{rank}(G) = |E(G)| - |V(G)| + 1$ . Since a Seifert graph has  $s(L)$  vertices and  $k(L)$  edges,  $\text{rank}(G) = k(L) - s(L) + 1$ . By Corollary 3.2, we know that

$$2g(F) = 1 - s(L) + k(L) + 1 - \mu(L) = \text{rank}(G) + 1 - \mu(L).$$

Thus  $\text{rank}(G) = 2g(F) + \mu(L) - 1$ . □

Based on our discussion following Theorem 4.3, the following corollary is immediate.

**Corollary 4.1** [2] *A basis for the first homology group of  $F$ , a Seifert surface of a link  $L$ , will have  $2g(F) + \mu(L) - 1$  loops.*

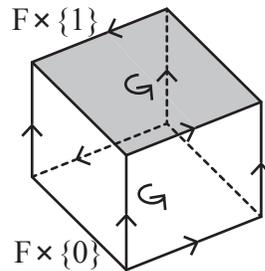
### 4.3 SEIFERT MATRICES

Now we are ready to construct our primary tool for generating new link invariants – Seifert matrices. The algorithm used to create the Seifert matrix of a given link projection is rather extensive and involved, but we will show that it is well worth the investment.

Let us fix  $F$ , a Seifert surface of a link  $L$  created from a projection  $D$ , which has Seifert graph  $G$ . The idea underlying our approach is that we will be thickening  $F$ , creating “duplicates” of the surface’s homology basis elements, and then creating a matrix using the linking numbers between the basis elements and the various duplicates.

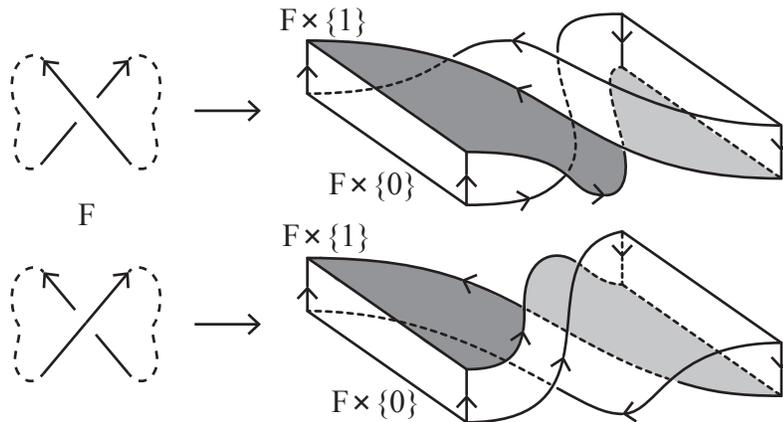
Based on our discussion in the previous section (see Theorem 4.3), we can create a homology basis for  $F$  by way of a spanning tree of  $G$ . Furthermore, by Corollary 4.1, we know that there are  $m = 2g(F) + \mu(L) - 1$  elements in this basis. Assign an orientation to each loop,  $\alpha_i$ . Let us thicken  $F$  by considering  $F \times [0, 1]$ . In this process, we need to ensure that both  $F$  and  $[0, 1]$  have orientations that obey the right-hand

rule as shown in Figure 4.7 [2]. Using our earlier choice of colors (see Theorem 3.5), we color the “top” of the thickened surface,  $F \times \{1\}$ , gray and the “bottom” white.



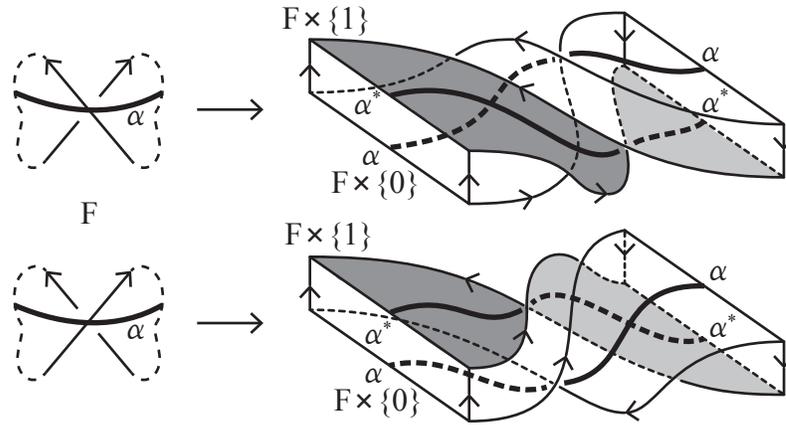
**Figure 4.7:** Right-hand orientation of a section of a thickened Seifert surface.

One of the trickier aspects of this thickening is considering what happens around a crossing in  $D$ . In  $F$ , we know that the crossing becomes a band with a half-twist, but it becomes more complicated when considering  $F \times [0, 1]$ , as shown in Figure 4.8.



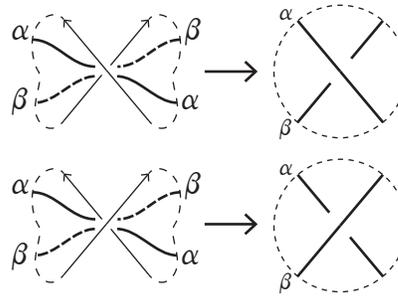
**Figure 4.8:** Twisted bands in a thickened Seifert surface.

In this new context, we may think of  $F$  as  $F \times \{0\}$ . Although homology basis elements of  $F$  should properly be referred to as  $\alpha_i \times \{0\}$ , we will continue to use the original notation for convenience. Now we create a copy of each homology basis element,  $\alpha_i^* = \alpha_i \times \{1\}$ , maintaining the original orientation of  $\alpha_i$ . Visually,  $\alpha_i^*$  lies just above  $\alpha_i$ . With this in mind, let us revisit the picture of the thickened bands, adding in homology basis elements (Figure 4.9).



**Figure 4.9:** Homology basis elements and copies in thickened Seifert surface.

It can take some time to properly visualize how various loops interact within these crossings. Let us take a step back and look at how general basis loops interact in the thickened surface. Consider two basis loops,  $\alpha$  and  $\beta$ , where one of the loops lies on  $F \times \{1\}$ . Then we can characterize the interactions of these two loops, as links, according to the twist in the surface, as in Figure 4.10 [5].



**Figure 4.10:** Loop interactions and resulting crossings in thickened Seifert surface.

To understand this characterization, let us walk through the first diagram in Figure 4.10. We have the two loops,  $\alpha$  and  $\beta$ , coming into the twisted band. As we approach the twist,  $\alpha$  will be twisted over  $\beta$ . So  $\alpha$  will cross over  $\beta$ . A similar process makes sense of the second diagram in Figure 4.10. In forming the link, we should always be mindful of which strand should be considered  $\alpha$  and which should be  $\beta$ .

This simplification holds even when we specifically consider  $\alpha$  and  $\beta = \alpha^*$ . Although, at the outset, the two loops should be positioned with one directly above the

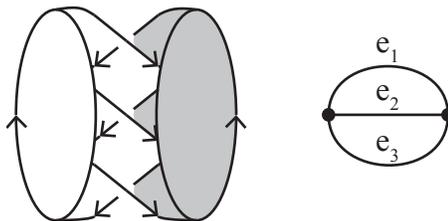
other, we should deform one within the surface so that we can see both clearly with no overlap. Additionally, Figure 4.9 is suggestive about how  $\alpha$  and  $\alpha^*$  interact.

With these interactions between basis loops in mind, we compute the linking number between the homology basis elements and the lifted loops,  $\text{lk}(\alpha_i, \alpha_j^*)$ . Finally we define the Seifert matrix of  $F$ .

**Definition 4.14** *Let  $F$  be a Seifert surface of a link  $L$  constructed from a projection  $D$  and let  $m = 2g(F) + \mu(L) - 1$ . Then a **Seifert matrix** of  $L$  is the  $m \times m$  matrix  $M = (\text{lk}(\alpha_i, \alpha_j^*))_{i,j=1,2,\dots,m}$  where  $\alpha_i$  is a homology basis element and  $\alpha_j^*$  is the duplicate of  $\alpha_j$  in the thickened surface  $F \times [0, 1]$ .*

In order to improve our understanding of this construction, let us consider a detailed example.

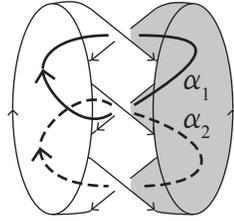
**Example 4.5** *Let us use the projection of the trefoil knot from Example 3.5. In Example 4.4, we found the Seifert surface and Seifert graph of this projection. For convenience, we have summarized this information in Figure 4.11.*



**Figure 4.11:** Seifert surface and Seifert graph of the trefoil knot projection for Example 4.5.

Since a spanning tree of the Seifert graph has a single edge, but the graph itself has three edges, we know there will be two cycles in the homology basis of the graph. Let us choose the cycles  $\alpha_1 = e_1e_2$  and  $\alpha_2 = e_2e_3$  as our homology basis. For convenience, looking down on the basis elements, let us orient them both clockwise. Mapping these cycles to the Seifert surface, we obtain the presentation in Figure 4.12.

Now let us consider the interactions of our cycles in the thickened version of our surface. Instead of attempting to depict this visually, we will consider the linking

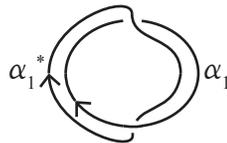


**Figure 4.12:** Trefoil knot Seifert surface with homology basis curves for Example 4.5.

of the cycles individually. First note that the Seifert circle on the right is oriented counterclockwise, while the one on the left is oriented clockwise. To maintain the correct orientation, our thickened surface will extend above our projection for the right circle and below the projection for the left circle.

Let us compute  $lk(\alpha_1, \alpha_1^*)$ . As we must follow the curve  $\alpha_1$  clockwise, let us start on the right Seifert circle. To understand the link formed by  $\alpha_1$  and  $\alpha_1^*$ , we may use a pen to represent how the components twist through a band, where the head of the pen represents  $\alpha_1^*$  and the bottom of the pen is  $\alpha_1$ .

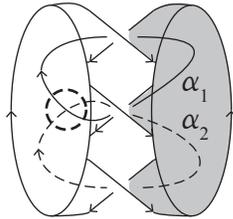
At the beginning of the curve, since the Seifert circle is oriented counterclockwise, our pen begins with the head pointing up. As we move along the curve and approach our first twist, the bottom of the pen starts moving toward us, signifying that  $\alpha_1$  is crossing over  $\alpha_1^*$ . Continuing into the second Seifert circle, the bottom of our pen is facing up. Approaching the second twist in the surface, the head of the pen starts moving away from us and so  $\alpha_1^*$  is crossing over  $\alpha_1$ . From there, we return to our starting point. Thus the link formed is the Hopf link (Figure 4.13).



**Figure 4.13:** Interaction of  $\alpha_1$  and  $\alpha_1^*$  in the thickened surface (Figure 4.12).

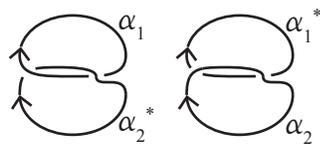
Since each crossing is positive,  $lk(\alpha_1, \alpha_1^*) = 1$ . This same process shows that  $lk(\alpha_2, \alpha_2^*) = 1$ .

Now we want to find  $lk(\alpha_1, \alpha_2^*)$ . While these curves share only one twisted band, we also need to consider the locations where the curves  $\alpha_1$  and  $\alpha_2$  intersect. As shown in Figure 4.14, there is only one such location in this projection.



**Figure 4.14:** Intersection of  $\alpha_1$  and  $\alpha_2$  in the Seifert surface from Figure 4.12.

Let us start from the perspective of  $\alpha_1$  on the right Seifert circle. Continuing along the curve, we will pass under  $\alpha_2^*$ . Now the only other consideration to make is the interaction of  $\alpha_1$  and  $\alpha_2^*$  at the point of intersection. Since the Seifert circle in question is oriented clockwise, we know that  $\alpha_2^*$  will lie below  $\alpha_1$ . Hence  $\alpha_1$  will cross over  $\alpha_2^*$ . Thus the link formed will again be a Hopf link (see Figure 4.15). Since we have two negative crossings in this Hopf link, we know that  $lk(\alpha_1, \alpha_2^*) = -1$ .



**Figure 4.15:** Links using mixed basis elements from Figure 4.12.

Lastly, we need to compute  $lk(\alpha_2, \alpha_1^*)$ . Based on our previous calculation, we already know that  $\alpha_2$  will cross over  $\alpha_1^*$  at the shared twist in the surface. At the intersection point between  $\alpha_1$  and  $\alpha_2$ ,  $\alpha_1^*$  will lie below  $\alpha_2$  because the Seifert circle is oriented clockwise. Thus  $\alpha_2$  crosses over  $\alpha_1^*$  and the resulting link will be the trivial link of two components (see Figure 4.15). Hence  $lk(\alpha_2, \alpha_1^*) = 0$ .

Finally, the Seifert matrix we have constructed is

$$M = \begin{bmatrix} lk(\alpha_1, \alpha_1^*) & lk(\alpha_1, \alpha_2^*) \\ lk(\alpha_2, \alpha_1^*) & lk(\alpha_2, \alpha_2^*) \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}.$$

As should be clear from the process we used to create the Seifert matrix, it is not a link invariant. By changing the projection of the link, changing orientations on the homology basis, or even selecting a different basis, we will change the Seifert matrix. This is unfortunate, but it is not the end of the story.

### 4.3.1 S-EQUIVALENCE OF SEIFERT MATRICES

Despite the fact that Seifert matrices are not link invariants themselves, we can relate any two Seifert surfaces of a link [2, 5]. Unfortunately, the method of showing this relationship is not as simple as applying Reidemeister moves. Instead, we require two new operations.

**Definition 4.15** [2, 5] *Let  $M$  be a Seifert matrix for some link. Then*

$\Lambda_1 : M \mapsto PMP^T$  where  $P$  is an invertible integer matrix with  $\det(P) = \pm 1$ ;

$$\Lambda_2 : M \mapsto M' = \begin{bmatrix} & * & 0 \\ & M & \vdots & \vdots \\ & & * & 0 \\ 0 & \cdots & 0 & 0 & 1 \\ 0 & \cdots & 0 & 0 & 0 \end{bmatrix} \text{ or } \begin{bmatrix} & & & 0 & 0 \\ & M & \vdots & \vdots \\ & & & 0 & 0 \\ * & \cdots & * & 0 & 0 \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix}$$

where  $*$  denotes an unknown integer.

While these operations may seem arbitrary and unconnected to the link we are interested in, it turns out that these represent very natural connections between the

surfaces.  $\Lambda_1$  is the transformation that arises from a change of basis [2]. In essence,  $\Lambda_1$  is the operation that accounts for changing the orientation of basis elements on the surface as well as handling a change in the order of basis elements in the matrix [5]. Thus, when applied to a given matrix  $M$ ,  $\Lambda_1$  has one of the two following effects:

1. Interchanges row  $i$  and row  $j$  and then interchanges column  $i$  and column  $j$ .
2. Adds  $k$  times row  $i$  to row  $j$  and then adds  $k$  times column  $i$  to column  $j$ .

Meanwhile,  $\Lambda_2$  and its inverse represent a change in genus of the Seifert surface, either adding or removing a tube from the surface [2, 5]. The addition of the tube creates an additional two basis elements that have to be considered in the Seifert matrix [2]. The resulting matrix from applying  $\Lambda_2$  depends on how the tube is attached to the surface. We obtain the first matrix if the outside of the tube is a part of the positive side of the surface (the gray-colored side, according to our algorithm in Theorem 3.5) and the second matrix otherwise [2].

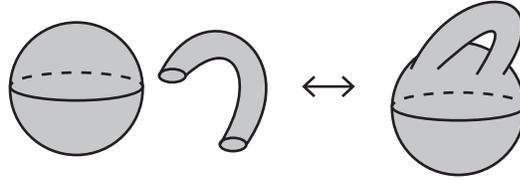
Using these new operations, we can define the notion of S-equivalence.

**Definition 4.16** [2, 5] *Two Seifert matrices  $M_1$  and  $M_2$  are said to be **S-equivalent**, denoted  $M_1 \sim_S M_2$ , if one can be obtained from the other by a finite sequence of operations  $\Lambda_1$ ,  $\Lambda_2$ , and  $\Lambda_2^{-1}$ .*

Any property of a link, derived from a Seifert matrix, is a link invariant if it is invariant under S-equivalence [2]. We will not go into the formal details surrounding this idea, but we will state the required facts to prove it.

**Theorem 4.5** [2] *Let  $L$  be a link. If  $F_1$  and  $F_2$  are surfaces such that each surface has boundary  $L$ , then there is a finite sequence of tubing and compressing operations such that  $F_1$  can be turned into  $F_2$ .*

For full disclosure, we should note that tubing and compressing operations correspond to adding or removing a tube respectively [2]. An example of this is shown in Figure 4.16.



**Figure 4.16:** Adding or removing a tube.

**Theorem 4.6** [2] *If two surfaces are related by a series of tubing and compressing operations, then their Seifert matrices are S-equivalent.*

Using these recent results, we conclude our discussion of S-equivalence with a significant and practical application of the notion: it tells us when a property of a Seifert matrix is a link invariant.

**Corollary 4.2** *Let  $L$  be a link and  $\gamma_L(M)$  a property of  $L$  derived from a Seifert matrix of  $L$ ,  $M$ . If  $\gamma_L(M)$  is invariant under S-equivalent Seifert matrices, then it is a link invariant.*

*Proof.* Suppose that  $\gamma_L$  is invariant under S-equivalent Seifert matrices. Let  $F_1$  and  $F_2$  be Seifert surfaces of  $L$ . Then we know that they are related under tubing and compressing operations by Theorem 4.5. By Theorem 4.6, the corresponding Seifert matrices,  $M_1$  and  $M_2$ , are S-equivalent. Then  $\gamma_L(M_1) = \gamma_L(M_2)$ . Thus  $\gamma_L$  is independent of the Seifert surface of  $L$  and so it is a link invariant.  $\square$

Restated, Corollary 4.2 says that there is a single condition to prove that a property of a Seifert matrix is a link invariant: it must be invariant under both  $\Lambda_1$  and  $\Lambda_2$  operations.

### 4.3.2 DETERMINANT AND SIGNATURE OF A LINK

Finally we can introduce two link invariants, the determinant and the signature of a link.

**Definition 4.17** [2, 5] Let  $M$  be a Seifert matrix of a link  $L$ . The **determinant** of  $L$  is  $\det(L) = |\det(M + M^T)|$ .

While the definition of the determinant of a matrix is rather self-explanatory, we must recall the notion of the signature of a matrix for the second invariant.

**Definition 4.18** [2] Let  $M$  be similar to a diagonal matrix  $D$  which has  $p$  positive terms and  $n$  negative terms. The **signature** of  $M$  is  $p - n$ .

**Definition 4.19** [2, 5] Let  $M$  be a Seifert matrix of a link  $L$ . The **signature** of  $L$ , denoted by  $\sigma(L)$ , is the signature of  $M + M^T$ .

Let us consider a brief example of these invariants to illustrate the computations involved.

**Example 4.6** Recall the Seifert matrix of the trefoil knot from Example 4.5,  $M = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}$ . Then  $M + M^T = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$ . So the determinant of the trefoil knot is

$$\det(M + M^T) = 3.$$

The characteristic polynomial of  $M + M^T$  is  $(\lambda - 3)(\lambda - 1)$ . Thus  $M + M^T$  is similar to the diagonal matrix  $D = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$ . Since the signature of  $D$  is 2, the signature of the trefoil knot is also 2.

The fact that the signature exists for any Seifert matrix is a result of  $M + M^T$  being, necessarily, a symmetric matrix. Applying the Spectral Theorem, any symmetric matrix is diagonalizable [2]. Even so, it is not clear that these “invariants” are well-defined at this point. After all, it seems not only plausible, but likely, that these values depend entirely on the Seifert matrix rather than the link. So we must show that they are truly link invariants.

**Theorem 4.7** [2, 5] *Let  $M$  be the Seifert matrix of a Seifert surface of a link  $L$ . Then  $\det(L)$  and  $\sigma(L)$  are link invariants (i.e., they depend only on  $L$  and not on  $M$ ).*

*Proof.* In order to show that  $\det(L)$  and  $\sigma(L)$  are link invariants, we will appeal to Corollary 4.2. We will first show that  $\det(L)$  is an invariant and then show that  $\sigma(L)$  is a link invariant.

We know that  $\Lambda_1(M) = P^T M P$  for some invertible integer matrix  $P$  with  $\det(P) = \det(P^T) = \pm 1$ . Thus

$$\begin{aligned} \det(\Lambda_1(M) + \Lambda_1(M)^T) &= \det(P^T M P + (P^T M P)^T) \\ &= \det(P^T M P + P^T M^T P) \\ &= \det(P^T (M + M^T) P) \\ &= \det(P^T) \det(M + M^T) \det(P) = \det(M + M^T). \end{aligned}$$

So  $\det(L)$  is invariant under the  $\Lambda_1$  operation.

Now we will show that  $\det(L)$  is invariant under  $\Lambda_2$ . We know that

$$\Lambda_2(M) + \Lambda_2(M)^T = \begin{bmatrix} & & * & 0 \\ & M + M^T & \vdots & \vdots \\ & & * & 0 \\ * & \cdots & * & 0 & 1 \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix}$$

and we can expand the determinant of this matrix using the lower-right  $2 \times 2$  matrix,

$$B = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \text{ to obtain } -\det(M + M^T). \text{ Since}$$

$$|\det(\Lambda_2(M) + \Lambda_2(M)^T)| = |-\det(M + M^T)| = |\det(M + M^T)|,$$

we have shown the invariance of  $\det(L)$  under the  $\Lambda_2$  operation.

Let us now consider the effect of a  $\Lambda_1$  operation on the signature of  $M + M^T$ . We may apply a result from linear algebra which states that the signature of a symmetric matrix is preserved by similarity [2]. As  $\Lambda_1$  is a specific similarity operation, the signature of  $M + M^T$  is preserved accordingly.

To show that  $\sigma(L)$  is invariant under the  $\Lambda_2$  operation, let us note that  $\Lambda_2(M) + \Lambda_2(M)^T$  is a symmetric matrix. So, by the Spectral Theorem, we know that it is diagonalizable. Thus there is a sequence of  $\Lambda_1$  operations that will change each unknown integer  $*$  to 0. Thus it is enough to compute the signature of  $B$ .

Since -1 and 1 are eigenvalues of  $B$ , there is an orthogonal matrix  $Q$  such that  $QBQ^T = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$ . As this matrix has signature 0,  $B$  must have signature 0. Therefore  $\sigma(\Lambda_2(M) + \Lambda_2(M)^T) = \sigma(M + M^T)$  as desired.

Thus we have shown that both  $\det(L)$  and  $\sigma(L)$  are invariant under S-equivalent Seifert matrices. Therefore, by Corollary 4.2, they are link invariants.  $\square$

Although there is a great deal of theory supporting the use of a link's determinant and signature, they are not our final goal for the Seifert matrix. However, the machinery we used to show their invariance will now allow us to develop a polynomial using the Seifert matrix.

## KNOT POLYNOMIALS

In the following chapter, we introduce one of the most successful types of link invariants – knot polynomials. We will develop three ways of taking a link diagram and computing a polynomial from it. We start by introducing one of the first knot polynomials, the Alexander polynomial, and then move on to the more recent HOMFLY-PT polynomial.

### 5.1 ALEXANDER POLYNOMIAL

There are two ways of computing the Alexander polynomial of a link, a link invariant discovered in 1928 [1]. The traditional approach uses Seifert matrices, which is the reason behind developing that body of theory in the previous chapter. The second method was developed by John Conway in 1969 and uses two rules to compute the polynomial directly from the link diagram [1]. Although the definitions will seem different, we will show that these methods produce equivalent polynomials. While the original method is more mathematically complex, the derivation makes it clear how the polynomial is related to the link.

### 5.1.1 COMPUTATION BY SEIFERT MATRIX

As suggested, our first approach to computing the Alexander polynomial of a link requires the use of a Seifert matrix of the link.

**Definition 5.1** [2] *Let  $L$  be a link and  $M$  any Seifert matrix for  $L$ . Then the **Alexander polynomial** of  $L$ , denoted by either  $\Delta(L)$  or  $\Delta_L(x)$ , is  $\det(xM - x^{-1}M^T) \in \mathbb{Z}[x, x^{-1}]$ .*

This definition of the Alexander polynomial is perhaps less used than another which can be obtained by making the change of variables  $x = t^{1/2}$ , with  $x^{-1} = t^{-1/2}$  [2]. However, our definition above uses whole powers of the variable and this seems somewhat more friendly [2]. As with the determinant and signature of a link, it is not immediately clear that this definition is independent of our choice of  $M$  and so we should prove it.

**Theorem 5.1** [2] *Let  $L$  be a link. Then  $\Delta(L)$  is a link invariant.*

*Proof.* Let  $M$  be a Seifert matrix of  $L$ . By Corollary 4.2, it is sufficient to show that any Seifert matrix  $S$ -equivalent to  $M$  gives the same determinant.

Since  $\Lambda_1(M) = PMP^T$  for some invertible, integer matrix  $P$  with  $\det(P) = \pm 1$ , we know that

$$\begin{aligned} \det(x\Lambda_1(M) - x^{-1}\Lambda_1(M)^T) &= \det(xPMP^T - x^{-1}(PMP^T)^T) \\ &= \det(xPMP^T - x^{-1}PM^T P^T) \\ &= \det(P(xM - x^{-1}M^T)P^T) \\ &= \det(P) \det(xM - x^{-1}M^T) \det(P^T) \\ &= \det(xM - x^{-1}M^T). \end{aligned}$$

Thus  $\det(xM - x^{-1}M^T)$  is preserved under  $\Lambda_1$ .

To show that this determinant is preserved under  $\Lambda_2$ , let us first note that  $x\Lambda_2(M) - x^{-1}\Lambda_2(M)^T$  has the matrix  $xM - x^{-1}M^T$  in the upper-left corner of the matrix and either  $B = \begin{bmatrix} 0 & x \\ -x^{-1} & 0 \end{bmatrix}$  or  $B^T$  in the bottom-right corner. Since  $\det(B) = \det(B^T) = 1$ , expanding the determinant of  $x\Lambda_2(M) - x^{-1}\Lambda_2(M)^T$  by  $B$  (or  $B^T$ ) yields  $\det(xM - x^{-1}M^T)$  and we have shown invariance under the  $\Lambda_2$  operation.

Hence  $\det(xM - x^{-1}M^T)$  is preserved under S-equivalence of  $M$  and thus it is independent of our choice of  $M$ . Therefore  $\Delta(L)$  is an invariant of  $L$ .  $\square$

It would be useful at this point to compute the Alexander polynomial of our knot from Example 4.5 to see how this computation plays out.

**Example 5.1** *We know from Example 4.5 that the Seifert matrix of our projection of the trefoil knot  $K$  (Figure 3.16) is  $M = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}$ . Then*

$$\begin{aligned} \Delta(K) &= \det(xM - x^{-1}M^T) = \det \left( \begin{bmatrix} x - x^{-1} & -x \\ x^{-1} & x - x^{-1} \end{bmatrix} \right) \\ &= (x - x^{-1})^2 - x^{-1}(-x) \\ &= x^2 - 2xx^{-1} + x^{-2} + 1 \\ &= x^2 - 1 + x^{-2}. \end{aligned}$$

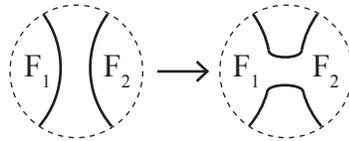
So  $\Delta(K) = x^2 - 1 + x^{-2}$ .

### 5.1.2 PROPERTIES OF THE ALEXANDER POLYNOMIAL

One of the more useful properties of the Alexander polynomial is that it behaves nicely under composition operations.

**Theorem 5.2** [5] *Let  $L_1$  and  $L_2$  be links. Then  $\Delta(L_1 \# L_2) = \Delta(L_1)\Delta(L_2)$ .*

*Proof.* For  $i = 1, 2$ , let  $F_i$  be a Seifert surface of  $L_i$  and  $M_i$  the corresponding Seifert matrix. Then a surface for  $L_1 \# L_2$  is created by joining  $F_1$  and  $F_2$  by a band as shown in Figure 5.1. To confirm that this new surface is indeed a Seifert surface for  $L_1 \# L_2$ , we should show that it is connected, orientable, and that its boundary is  $L_1 \# L_2$ . These first two properties follow immediately from  $F_1$  and  $F_2$  being connected orientable surfaces and the band itself being orientable. By the definition of  $L_1 \# L_2$ , we are essentially joining two Seifert circles, and so the new surface has boundary  $L_1 \# L_2$ .



**Figure 5.1:** Two surfaces joined by a band.

Now a Seifert matrix for this surface can be generated by using the basis elements of  $F_1$  and  $F_2$ . The Seifert graphs of the surfaces will be joined at a single vertex, so we will not need any additional cycles to form a basis. Moreover, the basis elements of  $F_1$  will not link with any basis elements of  $F_2$  because they are disjoint in space. In the Seifert matrix of this surface,  $M$ , this corresponds to having 0 entries whenever we consider basis elements from different surfaces. Additionally, joining the two surfaces will not impact the interactions of the basis elements of a single surface. Thus

$$M = \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} \text{ is a Seifert matrix for } L_1 \# L_2.$$

Therefore,

$$\begin{aligned}
\Delta_{L_1 \# L_2}(x) &= \det(xM - x^{-1}M^T) \\
&= \det \left( \begin{bmatrix} xM_1 - x^{-1}M_1^T & 0 \\ 0 & xM_2 - x^{-1}M_2^T \end{bmatrix} \right) \\
&= \det(xM_1 - x^{-1}M_1^T) \det(xM_2 - x^{-1}M_2^T) \\
&= \Delta_{L_1}(x) \Delta_{L_2}(x).
\end{aligned}$$

Hence  $\Delta_{L_1 \# L_2}(x) = \Delta_{L_1}(x) \Delta_{L_2}(x)$  as desired.  $\square$

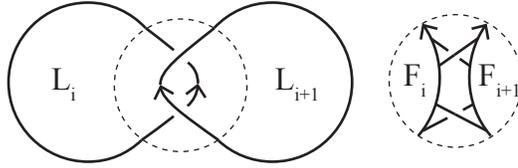
One of the less fortunate properties of the Alexander polynomial is that it does not detect split links. Recall that Seifert's algorithm does not produce a Seifert surface for a split projection of a split link. However, split links still have Seifert matrices and the following lemma gives us one such form for that matrix.

**Lemma 5.1** [5] *Let  $L$  be a split link projection where  $L = L_1 \sqcup L_2 \sqcup \cdots \sqcup L_n$  such that  $L_i$  is a non-split link. Choose a Seifert surface  $F_i$  for  $L_i$ , let  $M_i$  be the Seifert matrix of  $F_i$ , and denote the  $(n-1) \times (n-1)$  zero matrix by  $O_{n-1}$ . Then there is a Seifert matrix for  $L$  that is  $S$ -equivalent to*

$$M = \begin{bmatrix} M_1 & & & & \\ & M_2 & & 0 & \\ & & \ddots & & \\ & & & M_n & \\ & & & & O_{n-1} \end{bmatrix}.$$

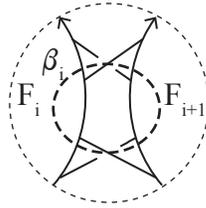
*Proof.* First let us create a connected surface,  $\tilde{F}$ , from all the  $F_i$ 's. For  $i = 1, 2, \dots, n-1$ , connect  $F_i$  and  $F_{i+1}$  by two bands with opposite twists as shown in Figure 5.2. This is equivalent to applying a Reidemeister II move to link  $L_i$  and  $L_{i+1}$  together, if the

relevant strands have opposite orientation (which can be assumed because we could change the projection otherwise).



**Figure 5.2:** Result of attaching twisted bands between disjoint Seifert surfaces.

Take a basis for the first homology group of  $F_i$ ,  $\mathcal{B}_i = \{\alpha_{i,1}, \alpha_{i,2}, \dots, \alpha_{i,k_i}\}$ , where  $i = 1, 2, \dots, n$ . Then the union of these bases is almost a basis for  $H_1(\tilde{F})$ . We only need to add the newly created cycles between  $F_i$  and  $F_{i+1}$ , which we will call  $\beta_i$  for  $i = 1, 2, \dots, n - 1$  and we will assume are oriented clockwise as we look at the surface. These new 1-cycles are shown in Figure 5.3.



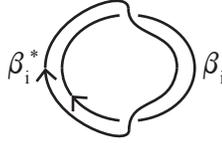
**Figure 5.3:** Presentation of  $\beta_i$  in  $\tilde{F}$ .

Now the Seifert matrix of  $\tilde{F}$  is  $\tilde{M}$  where the first  $k_1$  rows and columns involve the linking number of the elements of  $\mathcal{B}_1$  elements, the second  $k_2$  rows and columns involve the elements of  $\mathcal{B}_2$ , and so on, while the final  $(n - 1)$  rows and columns involve the  $\beta_i$ s.

Let us now consider the entries of  $\tilde{M}$ . We already know that  $\text{lk}(\alpha_{i,j}, \alpha_{k,l}^*) = 0$ , so long as  $i \neq k$ , because they correspond to different surfaces. So  $\tilde{M}'$ , the submatrix of  $\tilde{M}$  obtained by removing the last  $(n - 1)$  rows and columns, is simply a matrix with  $M_i$ 's on the diagonal and 0 entries otherwise.

Finally let us consider  $\text{lk}(\beta_i, \alpha_{j,k}^*)$ ,  $\text{lk}(\alpha_{j,k}^*)$ , and  $\text{lk}(\beta_i, \beta_j)$ . The first two linking numbers are easily computed to be 0 because  $\alpha_{j,k}$  and  $\beta_i$  do not share any twisted

bands and so they cannot be linked together. Similarly, for  $i \neq j$ ,  $\text{lk}(\beta_i, \beta_j^*) = 0$  because the cycles are disjoint. Lastly,  $\text{lk}(\beta_i, \beta_i^*) = 0$  as shown in Figure 5.4.



**Figure 5.4:** Link formed by  $\beta_i$  and  $\beta_i^*$ .

Thus  $\tilde{M} = M$ . Since  $\tilde{F}$  was created by a link projection equivalent to  $L$ ,  $\tilde{F}$  is a Seifert surface of  $L$  and so  $L$  has a Seifert matrix of the form  $M$ .  $\square$

**Theorem 5.3** [2, 5] *If  $L$  is a split link then  $\Delta(L) = 0$ .*

*Proof.* If  $L$  is a split link then we know by Lemma 5.1 that there is a Seifert matrix of  $L$ , which we will call  $M$ , with a zero matrix in the bottom-right corner. When computing the Alexander polynomial, this zero matrix will be preserved and so expanding by it to compute the determinant will give us  $\Delta_L(x) = \det(xM - x^{-1}M^T) = 0$ .  $\square$

To demonstrate the usefulness of this theorem, let us recall that the Whitehead link (Figure 2.24) has linking number 0. We claimed without proof that this link is also non-splittable, thus showing that not all linking number 0 links were splittable. However, if we go through the computation of the Alexander polynomial for the Whitehead link, we obtain  $x^3 - 3x + 3x^{-1} - x^{-3}$ . Since this is not the zero polynomial, we can apply Theorem 5.3 to conclude that the Whitehead link is non-splittable.

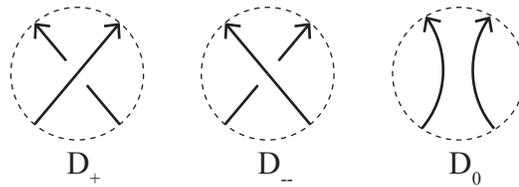
### 5.1.3 COMPUTATION BY CONWAY POLYNOMIAL

The second approach to finding the Alexander polynomial employs a recursive method. Although the result will be the same as the Alexander polynomial, we should give credit to John Conway and appropriately call this derivation the Conway polynomial.

**Definition 5.2** [1, 2, 5] Let  $L$  be an oriented link. Also let  $L_+$ ,  $L_-$ , and  $L_0$  be oriented links with projections  $D_+$ ,  $D_-$ ,  $D_0$  respectively such that

1. One of  $L_+$ ,  $L_-$ , and  $L_0$  corresponds to  $L$ ;
2.  $D_+$ ,  $D_-$ ,  $D_0$  are identical outside of the neighborhood a single crossing. Within this neighborhood, the projections differ as in Figure 5.5.

Then we call  $D_+$ ,  $D_-$ , and  $D_0$  **skein diagrams** of  $L$ . Any relation of these three diagrams is called a **skein relation**.



**Figure 5.5:** Neighborhood of skein diagrams.

Using these skein diagrams, which we will be seeing examples of soon, we can compute the Conway polynomial.

**Definition 5.3** [1, 2, 5] Let  $L$  be an oriented link. The **Conway polynomial**, denoted  $\nabla_L(z)$ , is determined using the following two axioms:

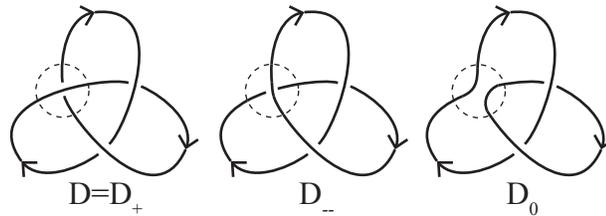
1. If  $L$  is the trivial knot then  $\nabla_L(z) = 1$ .
2. Let  $D_+$ ,  $D_-$ ,  $D_0$  be skein diagrams of  $L$ . Then the following skein relation holds:

$$\nabla_{L_+}(z) - \nabla_{L_-}(z) = z\nabla_{L_0}(z).$$

The proof that the Conway polynomial of a link is well-defined and unique is beyond what we will be covering in this section and so we will take it as a given [5]. Note that this means that the distinction between the skein diagrams  $D_+$ ,  $D_-$ ,  $D_0$  and

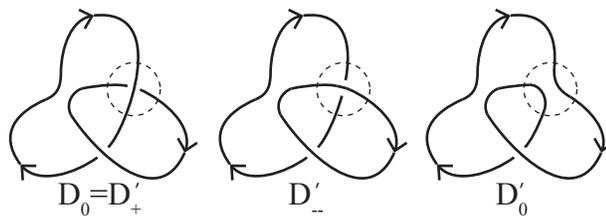
the actual links  $L_+, L_-, L_0$  is rather minimal. The Conway polynomial of a certain diagram is going to result in the polynomial for the link it represents and so we can equivalently discuss  $\nabla_L(z)$  or  $\nabla_D(z)$  where  $D$  is a projection of  $L$ . The interesting twist to using this approach to knot polynomial computation is that you end up with a tree of progressively simpler links. We do not want to go into too much detail, but we should at least consider an example of this method.

**Example 5.2** *Let us consider a projection of the trefoil knot,  $D$ , as shown in Figure 5.6 and select a crossing of it. Since we chose a positive crossing in Figure 5.6, we must consider  $D$  to be the  $D_+$  diagram. So the  $D_-$  and  $D_0$  diagrams are as shown in Figure 5.6.*



**Figure 5.6:** Trefoil knot projection and corresponding skein diagrams.

*We can see that  $D_-$  is trivial by the application of a Reidemeister II move followed by a Reidemeister I move. So  $\nabla(D_-) = 1$ . However,  $D_0$  is a Hopf link and so we must repeat this process of selecting a crossing and considering skein diagrams. To avoid confusing notation, we will now consider  $D_0$  to be  $D'_+$ , as shown in Figure 5.7.*



**Figure 5.7:** Hopf link projection and corresponding skein diagrams.

*In this case,  $D'_0$  is the unknot so  $\nabla(D'_0) = 1$ . Since  $D'_-$  is the unlink of two components, it will have Conway polynomial 0. This fact can be derived from only*

appealing to the skein relation for the Conway polynomial, but we will not go through that process and instead appeal to the equivalence of the Conway polynomial and the Alexander polynomial and use Theorem 5.3. Then, according to the skein relation,

$$\nabla_{D_0}(z) = \nabla_{D'_+}(z) = z\nabla_{D'_0}(z) + \nabla_{D'_-}(z) = z.$$

Thus the Conway polynomial of the trefoil knot is

$$\nabla_D(z) = \nabla_{D_+}(z) = z\nabla_{D_0}(z) + \nabla_{D_-}(z) = z^2 + 1.$$

Finally, let us show that the Conway polynomial is the Alexander polynomial in disguise. To do this, we first show that a similar skein relation holds for the Alexander polynomial.

**Theorem 5.4** [2] *Let  $D_+$ ,  $D_-$  and  $D_0$  be skein diagrams of a link  $L$ . Then*

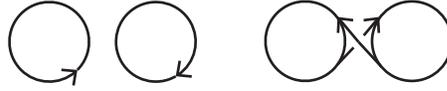
$$\Delta(L_+) - \Delta(L_-) = (x^{-1} - x)\Delta(L_0).$$

*Proof.* Let  $F_+$ ,  $F_-$ , and  $F_0$  be the Seifert surfaces constructed from  $D_+$ ,  $D_-$ , and  $D_0$  respectively and let  $M_0$  be the Seifert matrix of  $F_0$ . We will consider a number of cases that depend on the connectedness of the various projections.

If  $D_+$  is disconnected then  $D_-$  and  $D_0$  must also be disconnected. Certainly  $D_0$  will be disconnected because smoothing a crossing cannot connect disconnected components. Similarly,  $D_-$  only changes the crossing and so the disconnected component will not be affected. Hence  $L_+$ ,  $L_-$ , and  $L_0$  are all split links and so their Alexander polynomials are 0 by Theorem 5.3. The skein relation holds trivially in this situation. An analogous argument follows if we begin by assuming that  $D_-$  is disconnected.

If  $D_0$  is disconnected but  $D_+$  and  $D_-$  are connected (if one is, the other must be) then  $L_0$  is a split link. As we can see in Figure 5.8, the links for  $L_+$  and  $L_-$  are going

to be very similar. In fact, if we twist the right portion of  $L_-$  by one full rotation, we obtain  $L_+$ . Hence  $L_+$  and  $L_-$  are isotopic. So  $\Delta_{L_+}(x) - \Delta_{L_-}(x) = \Delta_{L_+}(x) - \Delta_{L_+}(x) = 0 = (x - x^{-1})\Delta_{L_0}(x)$  as desired.



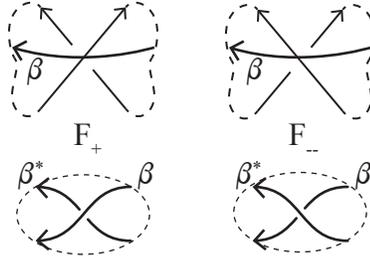
**Figure 5.8:** Disconnected  $L_0$  (left) and  $L_-$  (right) for proof of Theorem 5.4.

Lastly, suppose that all of the projections are connected. Let  $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$  be a basis for the first homology group of  $F_0$ . Each of these 1-cycles are also present in  $F_+$  and  $F_-$ . We can complete a basis for  $F_+$  by adding a loop  $\beta$  which passes once through the twisted band that came from the distinguished neighborhood and then travels through the rest of the surface. Assuming that the linking numbers involving  $\beta$  and  $\beta^*$  are counted in the last row and column, the Seifert matrix for  $F_+$  has the form

$$M_+ = \begin{bmatrix} & & & a_1 \\ & M_0 & & \vdots \\ & & & a_n \\ b_1 & \cdots & b_n & c \end{bmatrix} \text{ for } a_j, b_j \in \mathbb{Z}.$$

Using the same loop  $\beta$  in  $F_-$  gives a Seifert matrix  $M_-$  that is identical to  $M_+$  except in the final entry where, in  $F_-$ ,  $\text{lk}(\beta, \beta^*) = c + 1$ . This is because, as shown in Figure 5.9, the crossing of  $\beta$  and  $\beta^*$  in  $D_+$  will contribute -1 to the computation of the linking number while, in  $D_-$ , it contributes +1. So, in moving from  $D_+$  to  $D_-$ , we change the sum by 2 and thus add 1 to  $\text{lk}(\beta, \beta^*)$ . As a point of interest, note that this does not depend on the orientation of  $\beta$  as it does not affect whether the crossing is positive or negative.

Lastly let us compute the difference of the Alexander polynomials generated by



**Figure 5.9:** Interaction of  $\beta$  and  $\beta^*$  in Seifert surfaces of  $D_+$  and  $D_-$ .

expanding by the last column of the matrices. Note that the only terms which differ between  $\Delta_{L_+}(x)$  and  $\Delta_{L_-}(x)$  are those with coefficient either  $c$  or  $c + 1$  and so

$$\begin{aligned}
 \Delta_{L_+}(x) - \Delta_{L_-}(x) &= \det(xM_+ - x^{-1}M_+^T) - \det(xM_- - x^{-1}M_-^T) \\
 &= c(x - x^{-1}) \det(xM_0 - x^{-1}M_0^T) \\
 &\quad - (c + 1)(x - x^{-1}) \det(xM_0 - x^{-1}M_0^T) \\
 &= -(x - x^{-1})\Delta_{L_0}(x) = (x^{-1} - x)\Delta_{L_0}(x).
 \end{aligned}$$

Thus we have shown that the skein relation holds in all cases.  $\square$

Using Theorem 5.4, the equivalence of the Alexander polynomial and the Conway polynomial requires a simple substitution.

**Corollary 5.1** *The Conway polynomial gives the same result as the Alexander polynomial.*

*Proof.* Let  $L$  be a link and define  $z = x^{-1} - x$ . Then  $\nabla_L(z) = \nabla_L(x^{-1} - x)$ . Then, according to the skein relation used to compute the Conway polynomial and Theorem 5.4,  $\nabla_L(x^{-1} - x) = \Delta_L(x)$ .  $\square$

Let us take the Conway polynomial of the trefoil knot from Example 5.2,  $\nabla_D(z) = z^2 + 1$ . Making the substitution in Corollary 5.1, we get  $\Delta_D(x) = \nabla_D(x - x^{-1}) = (x - x^{-1})^2 + 1 = x^2 - 1 + x^{-2}$ . This matches our result from Example 5.1, verifying the result.

## 5.2 HOMFLY-PT POLYNOMIAL

The other knot polynomial we consider is called the HOMFLY-PT polynomial. Following mathematical tradition, this polynomial has been named after the researchers who discovered it. However, since Hoste, Ocneanu, Millett, Freyd, Lickorish, Yetter, Przytycki, and Traczyk were responsible, and it would be clumsy to refer to this polynomial as the “Hoste-Ocneanu-Millett-Freyd-Lickorish-Yetter-Przytycki-Traczyk polynomial,” we get to call it the HOMFLY-PT polynomial instead [3, 4].

Similar to the Conway polynomial, the HOMFLY-PT polynomial is computed recursively using a skein relation. In fact, the HOMFLY-PT polynomial is a generalization of the Conway polynomial, as we will see later [1].

**Definition 5.4** [1, 2, 4] *The **HOMFLY-PT polynomial** of an oriented link  $L$ , denoted  $P_L(l, m)$  or  $P(L)$ , is computed by applying the following axioms:*

1. *If  $L$  is the trivial knot then  $P_L(l, m) = 1$ .*
2. *Suppose that  $D_+$ ,  $D_-$ ,  $D_0$  are skein diagrams of  $L$ . Then the following skein relation holds:*

$$lP(L_+) + l^{-1}P(L_-) = -mP(L_0).$$

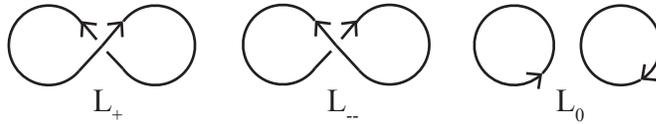
Once again, we will pass on the opportunity to prove that the HOMFLY-PT polynomial is a link invariant and take the result for granted [1, 2]. Despite the apparent similarity to the Conway polynomial, we should still go through a couple of examples to differentiate between the two skein relations.

**Example 5.3** *We can determine the HOMFLY-PT polynomial of the unlink with two components by using Figure 5.10. Since  $L_+$  and  $L_-$  are both the trivial knot, we know that  $P(L_+) = P(L_-) = 1$ . It is also clear that  $L_0$  is the unlink of two components. By the skein relation in Definition 5.4,*

$$lP(L_+) + l^{-1}P(L_-) = -mP(L_0)$$

$$l + l^{-1} = -mP(L_0)$$

Thus  $P(L_0) = -m^{-1}(l + l^{-1})$ .



**Figure 5.10:** Skein diagrams for Example 5.3.

As we can see in the above example, the HOMFLY-PT polynomial is able to distinguish between split links with some amount of success. This is a significant advantage compared to the Alexander polynomial.

**Example 5.4** Let us reuse Figures 5.6 and 5.7. As before,  $P(D_-) = 1$  and  $P(D'_0) = 1$ . Using the result from Example 5.3, we know that  $P(D'_-) = -m^{-1}(l + l^{-1})$ . Hence

$$lP(D'_+) + l^{-1}P(D'_-) = -mP(D'_0)$$

$$lP(D'_+) + l^{-1}(-m^{-1}(l + l^{-1})) = -m$$

$$lP(D'_+) - m^{-1}(1 + l^{-2}) = -m$$

$$lP(D'_+) = -m + m^{-1}(1 + l^{-2})$$

$$P(D'_+) = -ml^{-1} + m^{-1}(l^{-1} + l^{-3})$$

So  $P(D_0) = P(D'_+) = -ml^{-1} + m^{-1}(l^{-1} + l^{-3})$ . Thus

$$\begin{aligned} lP(D_+) + l^{-1}P(D_-) &= -mP(D_0) \\ lP(D_+) + l^{-1} &= -m(-ml^{-1} + m^{-1}(l^{-1} + l^{-3})) \\ lP(D_+) &= m^2l^{-1} - l^{-1} - l^{-3} - l^{-1} \\ lP(D_+) &= m^2l^{-1} - 2l^{-1} - l^{-3} \\ P(D_+) &= m^2l^{-2} - 2l^{-2} - l^{-4} \end{aligned}$$

Therefore the HOMFLY-PT polynomial of the trefoil knot is  $m^2l^{-2} - 2l^{-2} - l^{-4}$ .

While not immediately apparent, all of the information that we could glean from the Alexander polynomial is encoded into the HOMFLY-PT polynomial.

**Theorem 5.5** [1, 2, 4] For an oriented link  $L$ ,  $P_L(i, i(x - x^{-1})) = \Delta_L(x)$ .

*Proof.* If  $L$  is the trivial knot, then  $P_L(i, i(x - x^{-1})) = 1 = \Delta_L(x)$ . So suppose that  $L$  is not the trivial knot. Then

$$\begin{aligned} lP(L_+) + l^{-1}P(L_-) &= -mP(L_0) \Rightarrow iP(L_+) + \frac{1}{i}P(L_-) = -i(x - x^{-1})P(L_0) \\ &\Rightarrow iP(L_+) - iP(L_-) = i(x^{-1} - x)P(L_0) \\ &\Rightarrow P(L_+) - P(L_-) = (x^{-1} - x)P(L_0). \end{aligned}$$

Since the above skein relation is precisely that of the Alexander polynomial, it will necessarily give the same result.  $\square$

Given this clear relationship to our other polynomial, while providing even more information, it may not be obvious why we bothered to develop the tools required for the Alexander polynomial. Instead, we could have started off with a simple skein relation and obtained more powerful results. However, the development of the Alexander polynomial gave us a way to understand the reasoning behind the final

result. With the HOMFLY-PT polynomial, the result is more magical because it was derived recursively.

## BRAIDS

Throughout our introduction to knot theory, it may have been apparent that we avoided discussing the formation of links. While we noted that links are formed by interweaving string and gluing the appropriate ends, this is rather ambiguous and imprecise; it provides no systematic way to create links. Rather, we have waved our hands and a diverse range of links magically appeared. However, there are a number of techniques for producing links that have been researched since the inception of knot theory, including Dowker Notation, Conway Notation, and braids [1].

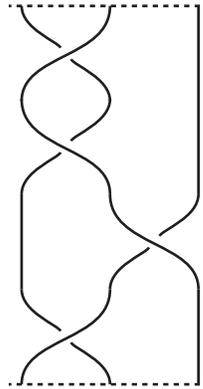
The following chapter will examine the creation of links via braids. First, we will define braids and establish their connection to links. Then we will explore the meaning of equivalent braids. Once that relationship has been determined, we will identify a well-researched invariant of links called the braid index.

## 6.1 BRAID REPRESENTATIONS

There are two major interpretations of braids, focusing on either the Artin braid group or braid diagrams [3]. Despite the attraction of exploring braids using the Artin braid group, we will emphasize the geometric structure of braids as the visualizations it provides will be of greater use. However, we will still employ the Artin braid group.

Intuitively, a braid, such as the one in Figure 6.1, is a collection interwoven strings

held in place by two bars. We can form a braid by using a simple process [1]. Take a horizontal bar and attach  $n$  strings to it. Using pairs of adjacent strings, create crossings by twisting the appropriate strings around each other. Once finished, attach the  $n$  strings to a second horizontal bar.



**Figure 6.1:** Example of a braid.

We can use this intuitive notion to help with our formal definition of a braid.

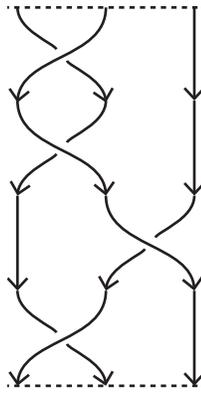
**Definition 6.1** [3] A *(geometric) braid* on  $n$  strings is a set  $D \subset \mathbb{R} \times [0, 1]$  which is the union of  $n$  topological intervals, known as the **strings** of  $D$ , such that

1. The projection  $\mathbb{R} \times [0, 1] \rightarrow [0, 1]$  maps each string homeomorphically onto  $[0, 1]$ ;
2. Every point of  $\{1, 2, \dots, n\} \times \{0, 1\}$  is the endpoint of a unique string;
3. Every point of  $\mathbb{R} \times [0, 1]$  belongs to at most two strings;
4. For each **crossing** of  $D$ , defined to be points belonging to exactly two strings, the strings must meet transversely and one must be distinguished as undercrossing and one overcrossing.

Unlike the other conditions, the second requirement for a braid is just a simplification that standardizes braid diagrams. It allows us to assume that  $\mathbb{R} \times \{1\}$  is the top of the braid and that  $\mathbb{R} \times \{0\}$  is the bottom of the braid. The remainder of the

conditions ensure that a braid is similar to a link projection. The topological intervals are supposed to act like strings whose ends have been fixed. Thus we want them to be homeomorphic to  $[0, 1]$ , as given by the first condition. We want to understand how the topological intervals interact, so we want at most two to be involved in any intersection (condition 3). Finally, we want to distinguish between the possible ways that strings could overlap each other, giving us condition 4.

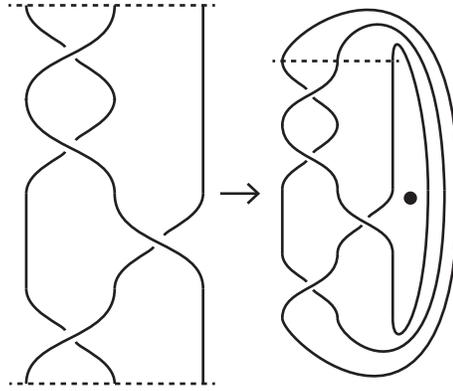
If it is necessary to impose an orientation on the strings, we will orient the strings from the top of the braid to the bottom as shown in Figure 6.2. Additionally, although we have displayed bars in many braid diagrams, we will generally not use them explicitly; they are simply a marker of where endpoints of strings have been fixed.



**Figure 6.2:** Oriented diagram of Figure 6.1.

### 6.1.1 CLOSED BRAID REPRESENTATION

The relationship between braids and links may not be immediately apparent. While both conveniently have to do with strings and crossings, that is not enough to say that they are related concepts. However, by pulling the bottom bar of an  $n$ -string braid around the point  $(n + 1, \frac{1}{2})$  and identifying it with the top bar (and subsequently deleting the bar), we get a link such as the one in Figure 6.3 [1]. Thus every braid diagram corresponds to some link.



**Figure 6.3:** Closure of Figure 6.1.

**Example 6.1** *Back in Figure 2.17, the initial projection of a trefoil knot is a closed braid representation where one strand has been isotoped to the other side of the braid.*

The natural follow-up question is whether every link can be represented by a braid. As it turns out, it is possible and this was first proved by James Alexander in 1923 [1, 3, 2].

**Theorem 6.1** (ALEXANDER'S THEOREM): [1, 2, 3] *Every link can be represented by a closed braid.*

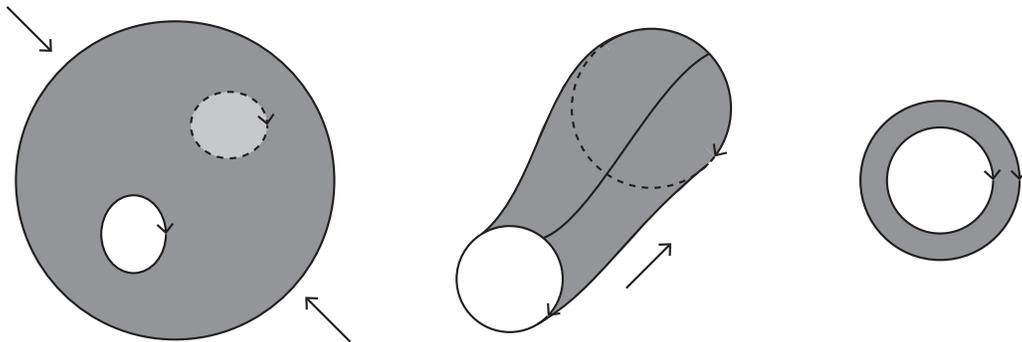
While Alexander's proof is described in [1], the process is not very useful outside of the proof as it can be difficult to replicate the steps concretely [2, 3]. A modern approach is to take an oriented link diagram and manipulate it until the corresponding Seifert circles are nested and have identical orientations [2]. Every braid has this configuration because the strings share an orientation and so the Seifert circles will as well.

Shuji Yamada showed that nesting Seifert circles could be done without increasing the number of Seifert circles compared to the original link projection [2]. Pierre Vogel improved on Yamada's result by using a single type of move to obtain the result [2]. We will be following a different proof of Alexander's Theorem, discovered by Pawel Traczyk, which improves on Vogel's method [2]. Out of the four proofs we have

mentioned, it is the most concise and elegant proof. While Alexander's Theorem is perhaps more elementary, this approach requires additional definitions and lemmas.

**Definition 6.2** [2, 3] *Two Seifert circles in  $S^2$  are **compatible** if they bound an annulus and, with respect to that annulus, they have parallel orientations. Otherwise, the Seifert circles are **incompatible**.*

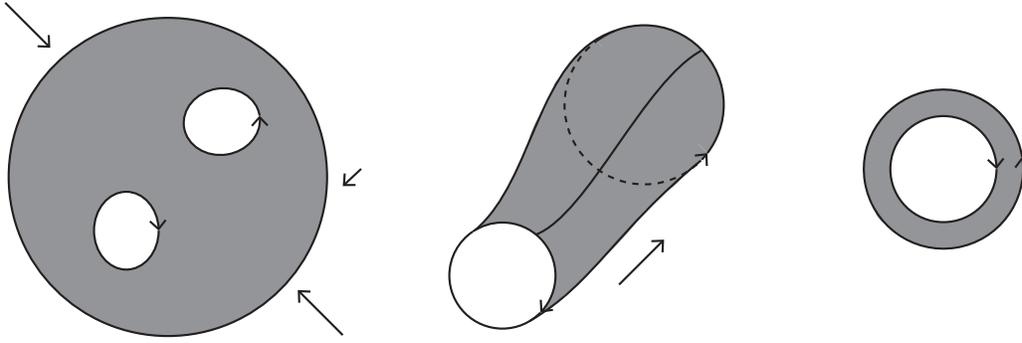
It turns out that any two disjoint topological circles in  $S^2$  bound an annulus, so this definition applies to any two Seifert circles [2, 3]. We know that if the two Seifert circles are nested, then they bound an annulus simply because they would bound an annulus in  $\mathbb{R}^2$  as well. It is less obvious why this is true if the Seifert circles are not apparently nested. However, if we consider the isotopies in Figures 6.4 and 6.5, then we can see that the result is still true. Another visualization is to isotope one Seifert circle around the sphere until it is clear that the Seifert circles are nested.



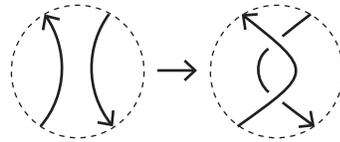
**Figure 6.4:** Compatible Seifert circles.

**Definition 6.3** [2, 3] *A **reducing move** on a link  $L$  is a Reidemeister II move such that the involved strands have opposing orientations and the strands belong to different Seifert circles, as shown in Figure 6.6.*

**Lemma 6.1** *Let  $L$  be a link projection. Then there are no pairs of incompatible Seifert circles if and only if all the Seifert circles of  $L$  are nested with parallel orientations.*



**Figure 6.5:** Incompatible Seifert circles.

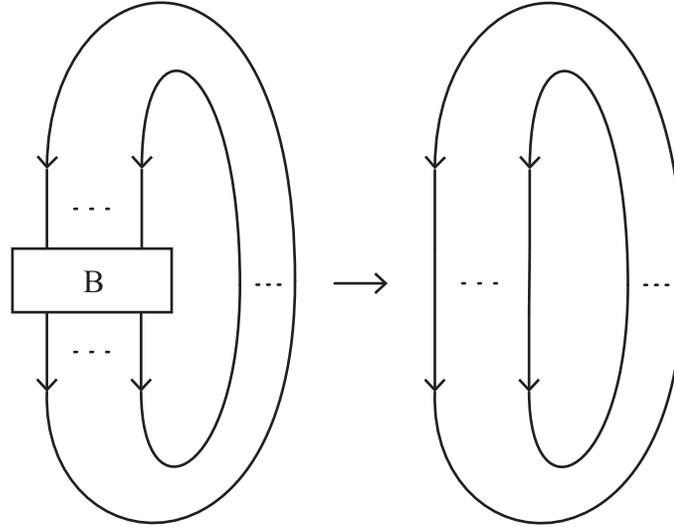


**Figure 6.6:** Vogel's reducing move.

*Proof.* Suppose that no two Seifert circles are incompatible. Then, using our earlier discussion, any pair can be considered either nested with parallel orientations or nested with opposing orientations. If we had a nested pair of Seifert circles with opposing orientations, they would be incompatible by Definition 6.2. Thus each pair must be nested with parallel orientations.

Conversely, if all Seifert circles are nested with parallel orientations then any two will be compatible.  $\square$

While we have proven Lemma 6.1 rather trivially, it is nonetheless essential to showing that every link has a braid representation. Let us consider the closure of some oriented  $n$ -string braid. Smoothing a crossing, as in Theorem 3.5, will always result in simply removing that crossing from the braid, leaving the strings otherwise unaffected. Hence, after smoothing all crossings, we will have nested Seifert circles with parallel orientations, as shown in Figure 6.7. Thus a braid representation of a link has no pairs of incompatible Seifert circles by Lemma 6.1.



**Figure 6.7:** General form of the closure of a braid  $B$ , before and after smoothing every crossing.

Conversely, suppose a link has nested Seifert circles with parallel orientations. Then we can choose the center point of the centermost Seifert circle and choose a line that goes through each Seifert circle in exactly one point. Cutting along this line, we obtain a deformed braid. Thus, if we can obtain a representation of a link with no pairs of incompatible Seifert circles, then it is necessarily a braid.

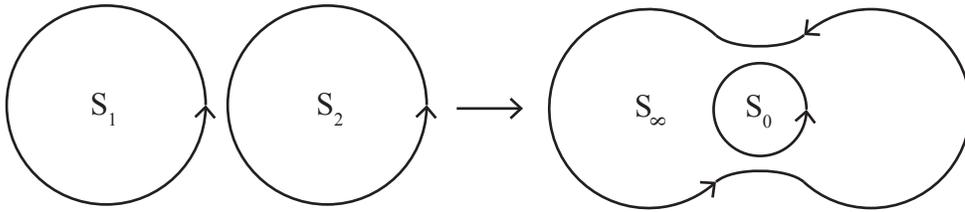
**Lemma 6.2** *Let  $L$  be a link projection with at least one incompatible pair of Seifert circles. Then a reducing move is possible.*

*Proof.* We will consider the link projection in  $S^2$  as a graph, where every crossing is a vertex and the edges are the strands between crossings. In this way, we can see that our graph divides  $S^2$  into a set of distinct regions bound by the various edges of the graph. It follows that every edge belongs to some Seifert circle of  $L$ .

We will call a region **reducible** if two edges bounding the region come from incompatible Seifert circles. Within a reducible region, we can apply a reducing move because there are no edges to interfere. We know that a reducible region exists because we have assumed that there is at least one pair of incompatible Seifert circles. Hence a reducing move is possible.  $\square$

**Lemma 6.3** *Let  $L$  be a link projection with  $\chi_L > 0$  pairs of incompatible Seifert circles,  $n$  Seifert circles, and  $L'$  the result of applying a reducing move. Then  $L'$  has  $n$  Seifert circles and  $\chi_{L'} = \chi_L - 1$  pairs of incompatible Seifert circles.*

*Proof.* Since  $\chi_L > 0$ , we know there is at least one pair of incompatible Seifert circles and so, by Lemma 6.2, a reducing move is possible. Let  $S_1$  and  $S_2$  be incompatible Seifert circles. In  $L'$ , the strands that form  $S_1$  and  $S_2$  give rise to a different pair of Seifert circles,  $S_0$  and  $S_\infty$ , as shown in Figure 6.8.



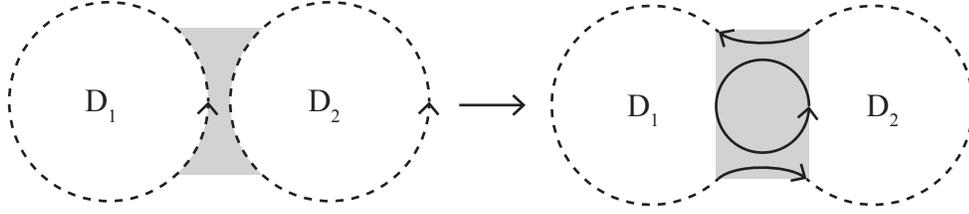
**Figure 6.8:** Changes in Seifert circles following a reducing move.

Since reducing moves are local changes in the link, the remainder of  $L$  is untouched and so all other Seifert circles are unchanged. Hence, the transformation of  $S_1$  and  $S_2$  into  $S_0$  and  $S_\infty$  creates no change in the number of Seifert circles as desired. Moreover, this means that the pairs of incompatible Seifert circles which involve neither  $S_1$  nor  $S_2$  will be unchanged after a reducing move, relative to  $S_0$  and  $S_\infty$ . Let us say there are  $h$  such incompatible pairs.

We know that  $S_1$  and  $S_2$  bound disjoint disks in  $S^2$ ,  $D_1$  and  $D_2$  respectively, as shown in Figure 6.9. Let  $d$  be the number of Seifert circles lying in  $S^2 \setminus (D_1 \cup D_2)$  that are incompatible with  $S_1$ . Equivalently, these Seifert circles will be incompatible with  $S_2$  because they are in the exterior of  $D_2$  and  $S_1$  and  $S_2$  are incompatible. Finally, for  $i = 1, 2$ , let  $d_i$  be the number of Seifert circles lying in  $D_i$ . We claim that

$$\chi_L = h + 2d + d_1 + d_2 + 1 \text{ and } \chi_{L'} = h + 2d + d_1 + d_2.$$

To substantiate this claim, we will consider each potential case: a given Seifert



**Figure 6.9:** Bounded disks before and after a reducing move.

circle of  $L$  is either compatible or incompatible with either, both, or neither of  $S_1$  and  $S_2$ ; similarly for a Seifert circle of  $L'$  related to  $S_0$  or  $S_\infty$ . We have already shown that there are  $h$  incompatible pairs of Seifert circles that involve neither  $S_1$  nor  $S_2$  in  $L$  and neither  $S_0$  nor  $S_\infty$  in  $L'$ .

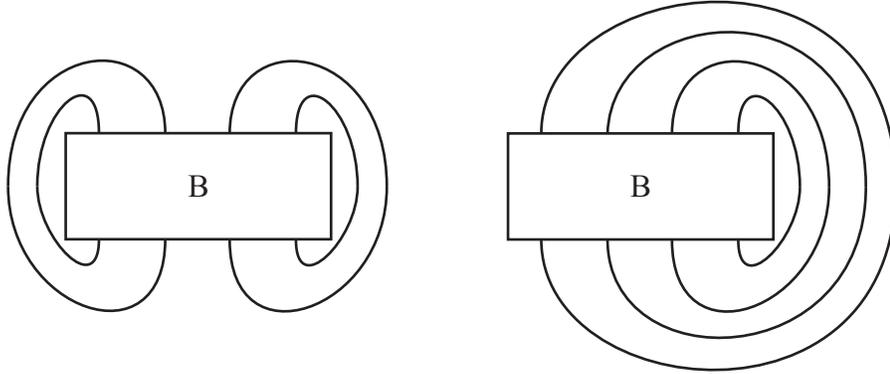
Since a Seifert circle of  $L$  in  $S^2 \setminus (D_1 \cup D_2)$  is incompatible with  $S_1$  if and only if it is incompatible with  $S_2$ , these Seifert circles contribute  $d$  pairs to  $\chi_L$  for  $S_1$  and another  $d$  pairs for  $S_2$  – a total of  $2d$  pairs. Similarly, a Seifert circle of  $L'$  in  $S^2 \setminus (D_1 \cup D_2)$  is incompatible with  $S_0$  if and only if it is incompatible with  $S_\infty$ . Hence we must add  $2d$  to  $\chi_{L'}$  as well.

If a Seifert circle of  $L$  is contained in  $D_i$ , then it must be compatible with either  $S_1$  or  $S_2$ . Hence every such Seifert circle must be counted, contributing  $d_1 + d_2$  incompatible pairs to  $\chi_L$ . Analogously, a Seifert circle of  $L'$  contained in  $D_i$  must be compatible with either  $S_0$  or  $S_\infty$  and so we must count every Seifert circle contained in  $D_1 \cup D_2$ ; contributing  $d_1 + d_2$  pairs to  $\chi_{L'}$ .

Finally, we know that  $S_1$  and  $S_2$  are incompatible, while  $S_0$  and  $S_\infty$  are compatible. Hence  $\chi_L = h + 2d + d_1 + d_2 + 1$  and  $\chi_{L'} = h + 2d + d_1 + d_2$ . Thus  $\chi_L = \chi_{L'} + 1$  or, equivalently,  $\chi_{L'} = \chi_L - 1$ .  $\square$

We are finally ready to prove Alexander's Theorem.

*Proof (Alexander's Theorem).* Let  $L$  be an oriented link projection with  $n$  Seifert circles lying in  $S^2$ . We will not concern ourselves with the difference between the closed braids in Figure 6.10 because they are isotopic on  $S^2$ .



**Figure 6.10:** Isotopic braids on  $S^2$ .

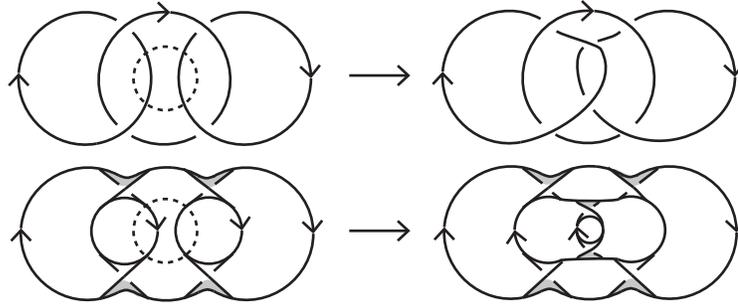
Let  $\chi_L$  be the number of pairs of incompatible Seifert circles. If  $\chi_L = 0$  then, by Lemma 6.1, we know that  $L$  is a closed representation of a  $n$ -string braid and we are done. So suppose that  $\chi_L \neq 0$ . Since  $L$  has  $n$  Seifert circles,  $\chi_L \leq \frac{n(n-1)}{2}$ .

By Lemma 6.2, we know that we can apply a reducing move to  $L$  to obtain  $L'$ . Applying Lemma 6.3 to  $L$ , we know that  $L'$  will have  $\chi_L - 1$  pairs of incompatible Seifert circles. Since  $\chi_L$  is finite, we can apply a total of  $\chi_L$  reducing moves to  $L$  to obtain a projection with 0 incompatible pairs of Seifert circles. Thus we have reduced our problem to the initial case and so this projection of the link is a closed braid representation.  $\square$

To solidify the process we used, let us go through an example.

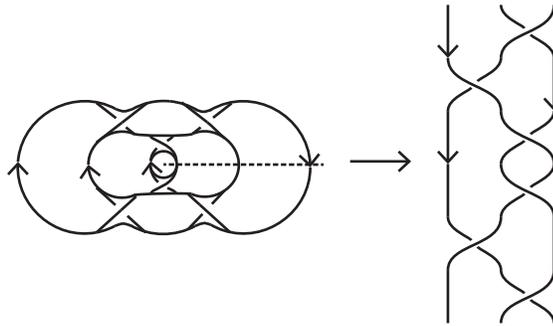
**Example 6.2** *Let  $L$  be the 3-component Hopf link from Figure 3.17. Since we already have the Seifert surface in the same figure, it is clear that not all of the Seifert circles are nested with parallel orientations. Hence, there is at least one pair of incompatible Seifert circles. In this case, the outer circle is compatible with both of the inner circles, but the two inner circles are incompatible. Let us apply a reducing move to the highlighted area in Figure 6.11.*

*Now all of the pairs of Seifert circles are compatible. According to our theorem,*



**Figure 6.11:** The 3-component Hopf link and its Seifert surface before and after a reducing move.

the link should now be a closed braid representation. We can realize this as shown in Figure 6.12.



**Figure 6.12:** A braid representation of the 3-component Hopf link.

Since we have concluded that every link has a braid representation, we will turn our attention toward representing braids in other ways.

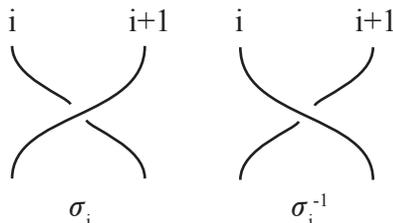
### 6.1.2 BRAID WORDS

Up-to-now, we have been working under a similar restriction as in our earlier discussion of knots: we have no clear means of systematically creating a braid. However, the structure of braids provides a relatively simple method of generating braids.

**Definition 6.4** [1] *Let  $D$  be a braid diagram on  $n$  strings. Represent a crossing of  $D$ , formed using strings  $i$  and  $i+1$ , by  $\sigma_i$  if string  $i$  is the undercrossing string. Otherwise,*

represent the crossing by  $\sigma_i^{-1}$ . Then the set  $\{\sigma_1\sigma_2, \dots, \sigma_{n-1}\}$  consists of the **braided generators** of  $D$ .

Furthermore, a **braided word** of  $D$  is obtained by describing every crossing in terms of generators and subsequently ordering them from the top of the braid to the bottom.



**Figure 6.13:** General form of braid generators.

We should note that it is convention to denote the braid word of the empty braid on  $n$  strings by 1 [3]. This is the braid which consists of  $n$  strings, none of which undercross or overcross any other string. Under the operation of composition, the  $n$ -strings braids form a group with identity 1, called the **braided group**  $B_n$  [3]. However, the braid group also has other properties which we will touch on later. For now, let us consider some examples.

**Example 6.3** The braid word for the braid in Figure 6.1 is  $\sigma_1\sigma_1^{-1}\sigma_2^{-1}\sigma_1$  and the braid in Figure 6.12 has braid word  $\sigma_2\sigma_1^{-1}\sigma_2^{-1}\sigma_2^{-1}\sigma_1\sigma_2$ .

Unfortunately, this definition does not eradicate all of the potential ambiguities that come from how we have defined braids. For instance, although arranging braid generators in a braid word based on their “height” in the braid diagram produces an ordering of the generators, we have not dealt with braid generators at the same height. Luckily, we will be able to show that, regardless of how we order these particular braid generators, the braid does not change. Hence, it is appropriate to refer to braid words as non-unique structures.

From a given braid word, we are able to recreate the braid by crossing the appropriate strings for each braid generator in the order that they are listed. Using

this precise construction methodology, we have created a map between braids and braid words on  $n$  strings. However, as noted above, we need to deal with the equivalency of braids.

## 6.2 BRAID EQUIVALENCE

Similar to our earlier notion of link equivalence, braid equivalence is a relation between isotopic braids.

**Definition 6.5** [1, 3] *Two braids on  $n$  strings  $B$  and  $B'$  are said to be **braid equivalent** if there is an ambient isotopy between  $B$  and  $B'$  which leaves the endpoints of each string fixed.*

From a more intuitive perspective, two braids are equivalent if the strings of the braids can be rearranged to look the same without moving the bars, all-the-while keeping the strings attached to the bars [1]. Furthermore, the strings cannot be pulled around the bars because the result would no longer be a braid [1].

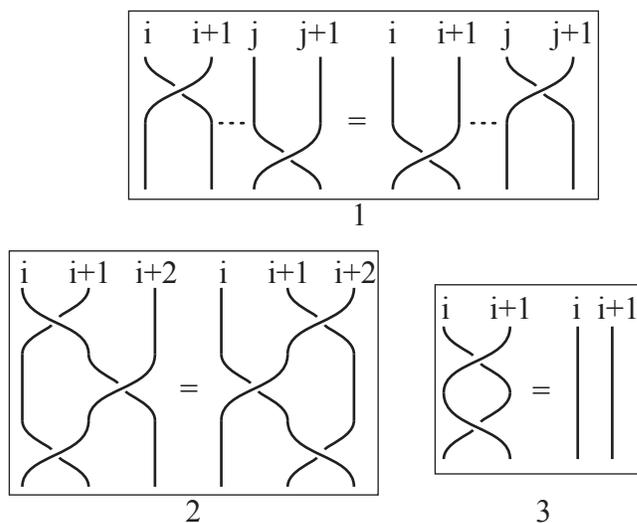
Somewhat fortuitously, there are similarities between acceptable isotopies of braids and Reidemeister moves. On the surface, it would seem as though this would be enough to handle braid representations of links in its entirety. However, the restrictions of the movement of the strings will require additional tinkering.

### 6.2.1 BRAID MOVES

First let us introduce three specific types of isotopies of braids. We will refer to these isotopies as **braid moves** since we will be focusing on how the isotopy affects the braid word. However, we will start with the geometric description of these moves and use this intuition to derive the formal braid moves.

As discussed earlier, in a geometric braid, it is possible to have multiple crossings occur on the same horizontal line. Within this set of crossings, it is irrelevant which

one is listed first in the braid word because the rest would follow immediately. When translated into a braid word, this information can appear lost because generators are represented so as to avoid this issue (with every crossing occurring on a different horizontal line). Thus the first braid move formalizes this geometric equivalence and produces a method of showing that two crossings can be presented on the same horizontal line, as shown in Figure 6.14.



**Figure 6.14:** Braid moves 1, 2 and 3.

The other two braid moves, also shown in Figure 6.14, are simply applying Reidemeister moves to the braid. The second braid move corresponds to a Reidemeister III move while the third braid move is precisely a Reidemeister II move [1].

**Definition 6.6** [1, 3] *Let  $B$  be an  $n$ -string braid. Then the following operations on the braid word of  $B$  are the result of applying an ambient isotopy to  $B$ :*

**Braid Move 1.** *If  $|i - j| > 1$  then  $\sigma_i^a \sigma_j^b = \sigma_j^b \sigma_i^a$  where  $a, b \in \{-1, 1\}$ .*

**Braid Move 2.** *For  $i \leq n - 2$  and  $a, b \in \{-1, 1\}$ ,  $\sigma_i^a \sigma_{i+1}^b \sigma_i^b = \sigma_{i+1}^b \sigma_i^b \sigma_{i+1}^a$ .*

**Braid Move 3.** *For any  $i$ ,  $\sigma_i \sigma_i^{-1} = 1 = \sigma_i^{-1} \sigma_i$ .*

The second braid move is seemingly the most complicated of the three moves. However, this is only because it must account for all the possible ways that a Reidemeister III move could be applied. The necessary distinctions between standard and inverse generators are not covered in either [1] or [3], which is a significant oversight. Both texts simply use the equivalence  $\sigma_i \sigma_{i+1} \sigma_i = \sigma_{i+1} \sigma_i \sigma_{i+1}$ . For example, by their restrictions, it is not apparent that  $\sigma_i^{-1} \sigma_{i+1} \sigma_i = \sigma_{i+1} \sigma_i \sigma_{i+1}^{-1}$  but it clearly holds geometrically as shown in Figure 6.14.

While we have listed the isotopy used in braid move 2 as a distinct move, it is a direct consequence of braids forming a group [3]. Hence, it is actually the first two moves that are essential for determining braid equivalence [3].

Currently, we have braid moves equivalent to Reidemeister II and III moves. While we may want something equivalent to a Reidemeister I move, it does not make sense in this context: the strings of braids do not cross themselves. However, using some additional terminology, we will be able to handle the equivalent scenario for a braid.

### 6.2.2 MARKOV EQUIVALENCE

In order to expand the list of acceptable braid moves, we require a more lenient type of equivalence than braid equivalence.

**Definition 6.7** [1] *Two braids are **Markov equivalent** if their closures produce equivalent links.*

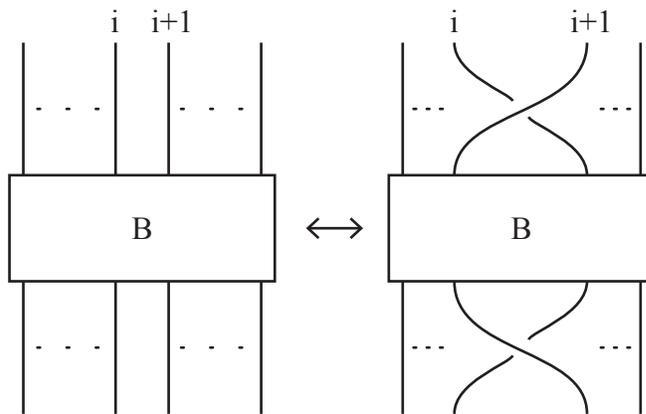
Rather than focusing on equivalent braids, Markov equivalence allows us to include moves that will not change the link type of the closed braid. All of the standard braid moves produce Markov equivalent braids because the isotopies do not change link type [1]. However, there are two more moves that produce Markov equivalent braids, known as Markov moves. While it is possible to derive these operations geometrically, it is a somewhat less natural way to describe the result. So we will first introduce Markov moves algebraically and then extract the geometric implications.

**Definition 6.8** [1, 3] *Let  $B$  be the braid word for an  $n$ -string braid. Then the following operations on  $B$ , called **Markov moves**, produce Markov equivalent braids via an ambient isotopy of the closed braid:*

**Conjugation.** *Let  $i \leq n - 1$ . Replace  $B$  with either  $\sigma_i B \sigma_i^{-1}$  or  $\sigma_i^{-1} B \sigma_i$ .*

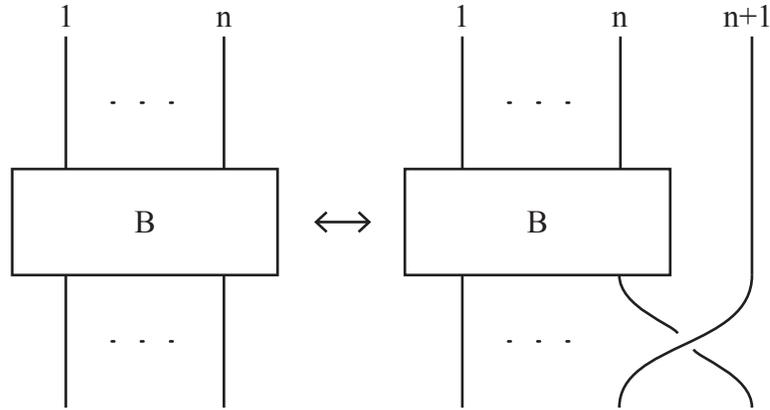
**Stabilization.** *Replace  $B$  with  $B \sigma_n$  or  $B \sigma_n^{-1}$ .*

As alluded to already, Markov moves produce Markov equivalent braids. Braid conjugation, shown in Figure 6.15, is a more subtle way of performing a Reidemeister II move on a braid compared to braid move 3. Let us use conjugation to add two generators to the braid word and then close the braid. Then we can isolate the crossing on top of the new braid and use ambient isotopies to shift it around to the bottom of the braid. Then it is clear that we can use a Reidemeister II move to remove these crossings. Hence the two braids produced equivalent links and so they are Markov equivalent.



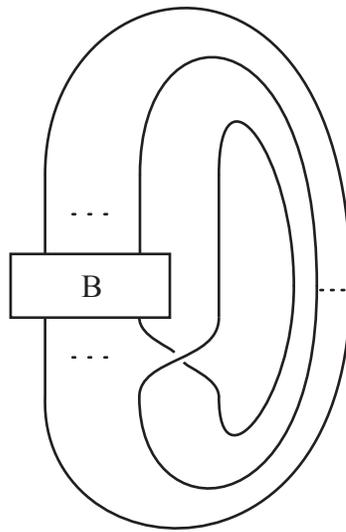
**Figure 6.15:** Example of braid conjugation.

Stabilizing a braid is the closest approximation of a Reidemeister I move available to braids. It takes an  $n$ -string braid and, after stabilization, it either becomes an  $(n + 1)$ -string braid, as in Figure 6.16, or an  $(n - 1)$ -string braid.



**Figure 6.16:** Example of braid stabilization.

Let  $B$  be a braid word for an  $n$ -string braid. Stabilizing  $B$  to produce  $B\sigma_n^{\pm 1}$ , we can see that the closure of the new braid has one additional loop in the center of the braid as in Figure 6.17. This loop can be undone using a Reidemeister I move and the result is simply the closure of  $B$ . Thus stabilization does not change the link type of the closed braid. Hence  $B$  and the stabilization of  $B$  are Markov equivalent braids.



**Figure 6.17:** Result of a stabilization operation.

The inverse stabilization move requires a bit more detail. In order for an inverse stabilization to be possible, there must be precisely one  $\sigma_{n-1}$  in an  $n$ -string braid.

The inverse operation will then remove this generator and the  $n$ th string to create a Markov-equivalent  $(n - 1)$ -string braid.

While we now have an expanded list of moves that will not affect the link type, we have not eliminated the possibility of there being additional isotopies that we should consider. Time for another major theorem!

**Theorem 6.2** (MARKOV'S THEOREM): *[1, 3] Two braids are Markov equivalent if and only if they are related by a finite sequence of Markov moves and braid moves.*

This result was proven by A. Markov in his 1935 paper, but we will not be delving into his proof [1]. It is sufficient for us to know that we have a complete list of moves to determine Markov equivalence. Instead, we will go through an example and develop more theory surrounding braids, including a discussion of a link invariant called the braid index.

**Example 6.4** *It is relatively easy to graphically realize that  $\sigma_2\sigma_1^{-1}\sigma_2^{-2}\sigma_1\sigma_2$  (Figure 6.12) and  $\sigma_1^{-1}\sigma_3\sigma_2\sigma_1^{-1}\sigma_3$  are Markov equivalent braids. We will use braid moves and Markov moves to demonstrate this equivalence.*

$$\begin{aligned}
& \sigma_1^{-1}\sigma_3\sigma_2(\sigma_1^{-1}\sigma_3) \xrightarrow{1} \sigma_1^{-1}(\sigma_3\sigma_2\sigma_3)\sigma_1^{-1} \xrightarrow{2} \sigma_1^{-1}\sigma_2\sigma_3\sigma_2\sigma_1^{-1} \\
& \xrightarrow{\text{conjugation}} \sigma_2\sigma_1^{-1}\sigma_1^{-1}\sigma_2\sigma_3\sigma_2(\sigma_1^{-1}\sigma_1)\sigma_2^{-1} \xrightarrow{3} \sigma_2\sigma_1^{-2}\sigma_2\sigma_3\sigma_2\sigma_2^{-1} \xrightarrow{3} \sigma_2\sigma_1^{-2}\sigma_2\sigma_3 \\
& \xrightarrow{\text{stabilization}} \sigma_2\sigma_1^{-2}\sigma_2 \xrightarrow{\text{conjugation,3}} \sigma_2^2\sigma_1^{-2} \xrightarrow{3} \sigma_2(\sigma_2\sigma_1^{-1}\sigma_2^{-1})\sigma_2\sigma_1^{-1} \\
& \xrightarrow{2} \sigma_2\sigma_1^{-1}\sigma_2^{-1}(\sigma_1\sigma_2\sigma_1^{-1}) \xrightarrow{2} \sigma_2\sigma_1^{-1}\sigma_2^{-1}\sigma_2^{-1}\sigma_1\sigma_2 \longrightarrow \sigma_2\sigma_1^{-1}\sigma_2^{-2}\sigma_1\sigma_2.
\end{aligned}$$

### 6.3 BRAID INDEX

In our previous discussion, we have mentioned that it is possible to represent links using different numbers of strings. In fact, repeatedly applying a stabilization move can create a braid representation with an arbitrarily large number of strings. However, what is less clear is the fewest number of strings required to represent a braid.

**Definition 6.9** [1] The **braid index** of a link  $L$ , denoted  $b(L)$ , is the fewest number of strings required in any braid presentation of  $L$ .

The fact that braid index is a link invariant is rather trivial. By Theorem 6.1, we know that every link has a braid presentation. Since braids necessarily have a finite number of strings, there must be a minimal such number. Hence, the braid index exists for every link. Moreover, it will not depend on the projection of the link because we have considered the number of strings across all possible braid representations.

While we can compute upper bounds on the braid index simply by finding a braid presentation for a link and counting the number of strings, it is difficult to know when the braid index has been realized. In fact, a great deal of research has gone into finding a lower bound for the braid index. We know that the unknot will have braid index 1 because the closure of an untwisted string is the unknot. Conversely, a non-trivial knot must have braid index greater than 1 because a single string cannot cross itself. Moreover, a link  $L$  must have  $b(L) \geq \mu(L)$  because each of the  $\mu(L)$  components must use at least one string.

**Example 6.5** We earlier demonstrated the Markov equivalence of the two braid words in example 6.4. Since the first braid word uses 3 strings, we know that the 3-component Hopf link has a braid index of at most 3. However, since it has 3 components, it must be that the braid index is exactly 3.

**Example 6.6** As we have already seen a braid representation of the trefoil knot using two strings (Figure 2.17), and the trefoil knot is non-trivial, we know that its braid index must be two. Indeed, there is a class of links called the 2-braid knots which are created by twisting two strings together [1].

More complicated bounds for braid index include a result by Yoshiyuki Ohyama which uses the crossing number and states that, for a non-split link  $L$ ,  $b(L) \leq \frac{c(L)}{2} + 1$

[1]. However, the most used bound has come to be referred to as the Morton-Franks-Williams inequality after the researchers who discovered it [1].

**Theorem 6.3** (MORTON-FRANKS-WILLIAMS INEQUALITY): [1, 2, 4] *Let  $L$  be a link projection and  $P_L(l, m)$  its HOMFLY-PT polynomial where  $e$  and  $E$  are respectively the smallest and largest degrees of  $l$  in  $P_L(l, m)$ . Then*

$$b(L) \geq \frac{E - e}{2} + 1.$$

The Morton-Franks-Williams inequality is sharp for most of the knots with 10 or fewer crossings. While we will not be applying the result ourselves, it is an important inequality to keep in mind.

## BRAID HEIGHT AND THE MACHETE NUMBER

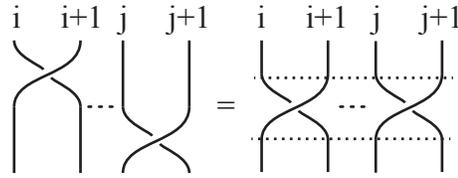
At this point, we should have sufficient experience and understanding of classical knot theory to enjoy the machete number. We begin by defining new properties of braids and then use them to define and explore the machete number. To conclude, we discuss a myriad of conjectures that arose during the investigation.

## 7.1 BRAID HEIGHT

The first new braid concept is that of braid level. Intuitively, a braid level consists of all the crossings along a given horizontal line in a braid.

**Definition 7.1** *Let  $B$  be a braid word and  $\omega = \sigma_{i_1}\sigma_{i_2}\cdots\sigma_{i_m}$  a non-empty subsequence of generators from  $B$ . Then  $[\omega]$  is a **braid level** of  $B$  if  $|i_k - i_l| > 1$  for all  $k \neq l$  with  $1 \leq k, l \leq m$ .*

We can match our formal definition of a braid level with the intuitive understanding rather simply. In a braid level, every pair of braid generators must satisfy the conditions for braid move 1 from Definition 6.6. Hence, within the braid level, we can commute any two generators. Geometrically, this precisely corresponds to having a braid projection where all of the crossings in the braid level can occur along a single horizontal line as in Figure 7.1.



**Figure 7.1:** Braid generators which can be presented on the same horizontal line.

In order to demarcate braid levels in a braid word, we will adopt the notation of enclosing the braid level in square brackets. It naturally follows that we can partition a braid word  $B$  into braid levels by requiring that every generator be a part of some braid level. Let us consider a few examples to solidify the idea.

**Example 7.1** Let  $B_1 = \sigma_1\sigma_3\sigma_5^{-1}\sigma_3\sigma_4^2$ . The following are three partitions of  $B_1$  into braid levels:

$$[\sigma_1\sigma_3\sigma_5^{-1}][\sigma_3][\sigma_4][\sigma_4], \quad [\sigma_1\sigma_3][\sigma_5^{-1}\sigma_3][\sigma_4][\sigma_4], \quad [\sigma_1][\sigma_3][\sigma_5^{-1}\sigma_3][\sigma_4][\sigma_4].$$

In the first two cases,  $B_1$  is partitioned into 4 braid levels, but the third partition has 5 braid levels. For the first partition, the first braid level is “maximal” in a sense – we cannot include any neighboring generators in the braid word into that level. However, in the second partition, the second braid level would be considered “maximal.” Finally, in the third partition, none of the first three braid levels can be considered “maximal.” Thus we cannot truly define an optimal partition of a braid word.

Now take  $B_2 = [\sigma_1][\sigma_2\sigma_4]$ . This relatively simple braid effectively illustrates an important fact about braid levels. First note that it is currently impossible to include  $\sigma_1$  in any other braid level. However, since  $[\sigma_2\sigma_4]$  is a braid level, we have  $\sigma_1[\sigma_2\sigma_4] = \sigma_1[\sigma_4\sigma_2]$ . Now we can partition the same braid word as  $[\sigma_1\sigma_4][\sigma_2]$ . Hence, not only is partitioning a non-unique process, but it may not be possible to realize a given partition without applying braid moves.

As we can see throughout Example 7.1, the decomposition of a braid into braid

levels is not unique and we cannot optimize braid word partitions of a given braid word. Even so, we can use braid levels to define a new link invariant.

**Definition 7.2** *Let  $L$  be a link. The **braid height** of  $L$ , denoted  $bh(L)$ , is the smallest  $n \in \mathbb{N}$  such that there is a braid representation of  $L$  with  $n$  braid levels.*

*A braid representation of  $L$  that realizes  $bh(L)$  braid levels is called a **braid height representation** of  $L$ .*

The fact that braid height is a link invariant follows from the existence of a braid representation (Theorem 6.1), the existence of a braid level decomposition (we can always partition a braid into levels consisting of one generator), and the choice of minimizing braid levels across all possible braid representations. Note that the unlink of  $n$  components has a braid representation consisting of the empty braid word on  $n$  strings. Thus the unlink has braid height 0. While finding the braid height turns out to be a relatively difficult task, we can employ a simple bound for non-trivial links.

**Proposition 7.1** *Every non-trivial link  $L$  has  $bh(L) > 1$ .*

*Proof.* To the contrary, suppose that  $L$  is a link with  $bh(L) = 1$ . Consider a braid height minimal representation of  $L$ . Since every string is used by at most one generator, the strings can be partitioned into collections of strings consisting of either individual strings or strings used by a generator. Upon closing the braid, these collections form a split link projection of  $L$ , where each component has one crossing. By Corollary 2.1, each such component is trivial and so  $L$  is the unlink. So every link  $L$  with  $bh(L) < 2$  is trivial and thus a non-trivial link has braid height of at least 2.  $\square$

The apparent usefulness of the braid height invariant is that it can be computed algorithmically. Starting with a list of links with braid height  $i$ , it may be possible to compute all possible braids with braid height  $(i + 1)$ . We simply enumerate all possibilities, take the closures, and eliminate those links of braid height  $i$ ! In general, this is not a reasonable task because we could always add additional strings to the

braid and we would have another possibility. We can simplify matters significantly by restricting our search to the braid heights of non-split links.

**Lemma 7.1** *Let  $L$  be a non-split link with a braid representation on  $n$  strings. Then any braid word for  $L$  must contain  $\sigma_i$  for all  $1 \leq i \leq n - 1$ .*

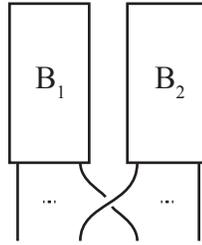
*Proof.* Suppose that  $B$  is a braid word for  $L$  which does not contain  $\sigma_i$  for some  $i$ . Then strings  $i$  and  $(i + 1)$  never cross and so the closure of strings 1 through  $i$  form a link that is split from the closure of strings  $(i + 1)$  through  $n$ . However,  $L$  was not a split link and so  $\sigma_i$  must be present in  $B$ .  $\square$

Lemma 7.1 tells us that we must have every generator in a braid word for a non-split link. Unfortunately, even with this additional requirement, there is an infinite family of non-split links with braid height 3. Thus we still have a computational nightmare on our hands. However, we can still classify all the non-split links up to braid height 3. To accomplish this classification, we will apply the following results that bound the braid height of composite links.

**Theorem 7.1** *For any links  $L_1$  and  $L_2$ ,  $bh(L_1 \# L_2) \leq \max \{bh(L_1), bh(L_2)\} + 1$ .*

*Proof.* Consider braid height representations of  $L_1$  and  $L_2$ ,  $B_1$  and  $B_2$  respectively. Without loss of generality, let  $B_1$  be a braid on  $n$  strings and let  $bh(L_1) \geq bh(L_2)$ . So  $B_1$  is a sequence of generators selected from  $\{\sigma_1, \sigma_2, \sigma_{n-1}\}$  and inverses of these generators. Lay  $B_2$  next to  $B_1$  and compose the resulting braid with  $\sigma_n$  to form  $B_3$ , as in Figure 7.2.

The closure of  $B_3$  is  $L_1 \# L_2$ . Moreover, in  $B_3$ , we can extend the  $i$ th braid level of  $B_1$  to include the generators in the  $i$ th braid level of  $B_2$  because they use disjoint strings except for the single  $\sigma_n$ . Assuming that  $[\sigma_n]$  is its own level of  $B_3$ , this process used  $bh(L_1) + 1$  braid levels to represent  $B_3$  and so  $bh(L_1 \# L_2) \leq \max \{bh(L_1), bh(L_2)\} + 1$ .  $\square$



**Figure 7.2:**  $B_3$ , the composition of  $B_1$  and  $B_2$ .

Using a similar idea, we can establish a strong lower bound on the braid height of a composite link as well.

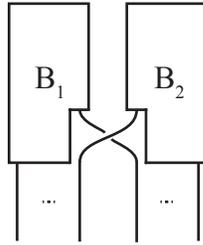
**Theorem 7.2** *For any links  $L_1$  and  $L_2$ ,  $bh(L_1\#L_2) \geq \max\{bh(L_1), bh(L_2)\}$ .*

*Proof.* Let  $B_1$  and  $B_2$  be braid height representations for links  $L_1$  and  $L_2$  respectively. Assume, without loss of generality, that  $B_1$  uses  $n$  strings and  $bh(L_1) \geq bh(L_2)$ . Finally, to the contrary of our statement, suppose that  $bh(L_1\#L_2) < bh(L_1)$ .

Let  $B_3$  be a braid height representation of  $L_1\#L_2$ . Since  $L_1\#L_2$  is a composite link, there is a disk,  $D$ , in the closure of  $B_3$  which allows us to realize a decomposition of the link into  $L_1$  and  $L_2$ . Since  $B_3$  was a braid, the decomposition using  $D$  will create closed braid representations of  $L_1$  and  $L_2$ . Moreover, each of these braid representations are partitioned into at most  $bh(L_1\#L_2)$  braid levels. Specifically, there is a braid representation of  $L_1$  that uses  $bh(L_1\#L_2)$  braid levels, but this contradicts the minimality of  $bh(L_1)$ . Thus  $bh(L_1\#L_2) \geq \max\{bh(L_1), bh(L_2)\}$ .  $\square$

Now we know that a composite link has either  $bh(L_1\#L_2) = \max\{bh(L_1), bh(L_2)\}$  or  $bh(L_1\#L_2) = \max\{bh(L_1), bh(L_2)\} + 1$ . In our earlier proofs, we ignored the possibility of including  $\sigma_n$  in an existing braid level of the composite link (Figure 7.3). As it turns out, this possibility is actually a fairly common occurrence. However, we must rely on a lemma to prove the result.

**Lemma 7.2** *Let  $L$  be a non-trivial non-split link with an  $n$ -string braid representation*



**Figure 7.3:** Composition of  $B_1$  and  $B_2$  with  $\sigma_n$  in an existing braid level.

*B.* Then  $n = 2$  if and only if  $B$  can be decomposed into braid levels such that every braid level contains  $\sigma_i$  for some  $1 \leq i \leq n - 1$ .

*Proof.* First suppose that  $n = 2$ . Then  $B$  is the composition of some number of  $\sigma_1$ 's and  $\sigma_1^{-1}$ 's. Since both of these generators use string 1, every braid level of  $B$  can have at most one generator. Moreover, every level must contain at least one generator and so every level of a partition of  $B$  would contain  $\sigma_1$  or its inverse.

Now, to the contrary, suppose  $n \neq 2$ . Since  $L$  is non-trivial, we may ignore the possibility that  $n = 1$  and so  $n \geq 3$ . By Lemma 7.1, at least one of  $\sigma_{i-1}$  and  $\sigma_{i+1}$  must be present in some braid level of  $B$ . However, the braid level containing either  $\sigma_{i-1}$  or  $\sigma_{i+1}$  cannot contain  $\sigma_i$  by definition. Yet we assumed that  $\sigma_i$  was present in every braid level. Thus  $n = 2$ .  $\square$

**Theorem 7.3** *If neither  $L_1$  nor  $L_2$  is a 2-braid link, then*

$$bh(L_1 \# L_2) = \max \{bh(L_1), bh(L_2)\}.$$

*Proof.* Suppose that a braid height representation of  $L_1$  uses  $n$  strings. Without loss of generality, let us say that  $bh(L_1) \geq bh(L_2)$ . There are two cases to consider:

**Case 1:** Suppose that  $bh(L_1) > bh(L_2)$ . Then it is sufficient to find a braid level of  $L_1$  in which  $\sigma_n$  can be added. If such a level exists, we can manipulate the braid such that it becomes the final level of the braid. Visually, this simply requires us to take the closed braid form and push each of the braid levels below it around the loop

so that they start stacking onto the top of the braid. Formally, this corresponds to a sequence of conjugation operations. Since  $L_1$  is not a 2-braid, there is some level for which  $\sigma_n$  can be added by Lemma 7.2.

**Case 2:** Suppose that  $bh(L_1) = bh(L_2)$ . By Case 1, we are able to find a braid level of  $L_1$  in which  $\sigma_n$  can be added. However, we still require a level of  $L_2$  in which the first string of  $L_2$  is not used so that  $\sigma_n$  can be incorporated into a braid level of  $L_1 \# L_2$ . Such a level exists by Lemma 7.2.

In both cases, we have the desired result.  $\square$

Note that our argument above never made use of  $L_2$  not being a 2-braid when  $bh(L_1) > bh(L_2)$ , so we can strengthen this result trivially.

**Corollary 7.1** *Let  $L_1$  and  $L_2$  be links with  $bh(L_1) > bh(L_2)$ . If  $L_1$  is not a 2-braid then  $bh(L_1 \# L_2) = \max\{bh(L_1), bh(L_2)\}$ .*

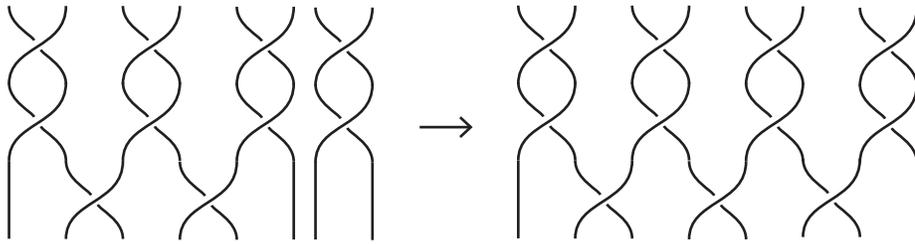
Let us now classify all links with braid height of at most 3. To minimize confusion and to clearly see the progression of links, we will break this result into smaller results. By Proposition 7.1, we know that we only need to consider links of braid height at least 2.

**Theorem 7.4** *Let  $L$  be a non-trivial non-split link. If  $bh(L) = 2$  then  $L$  is the Hopf link.*

*Proof.* Let  $B$  be a braid word for a braid height representation of  $L$ . If  $B$  is not a 2-braid, then  $B$  uses  $n \geq 3$  strings. By Lemma 7.1,  $\sigma_1, \sigma_2, \dots, \sigma_{n-1}$  must all be applied in  $B$ . Since  $\sigma_i$  and  $\sigma_{i+1}$  both use the  $(i+1)$ st string, they must be on different braid levels of  $B$ . Moreover, since  $B$  has only two braid levels, each generator must be applied exactly once. Hence we can stabilize  $B$   $n-1$  times to obtain the trivial braid on 1 string, which has braid height 0. Therefore, any braid of braid height 2 must use at most two strings.

Suppose that  $B$  is a 2-braid instead. By Lemma 7.2, we know that every decomposition of  $B$  into braid levels will contain  $\sigma_1$  or  $\sigma_1^{-1}$ . Since  $\sigma_1\sigma_1^{-1} = 1$ , this braid would have braid height 0. So  $B = \sigma_1^2$  or  $B = \sigma_1^{-2}$ . In either case, the closure of  $B$  is the Hopf link.  $\square$

Theorem 7.4 tells us that the Hopf link is the first link with non-zero braid height. This also means that, in cataloging braids of greater braid height, we can use the Hopf link as a building block. For example, if we take the link composition of two Hopf links, we obtain the 3-component Hopf link. Generalizing this process, we can compose an  $n$ -component Hopf link with another Hopf link to get an  $(n + 1)$ -component Hopf link by choosing to compose the “end” components of the respective Hopf links (see Figure 7.4). It turns out that the  $n$ -component Hopf link has a predictable braid height.



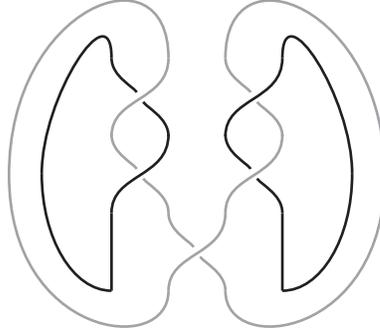
**Figure 7.4:** Composition of a 4-component Hopf link with another Hopf link.

**Proposition 7.2** *The  $n$ -component Hopf link has braid height 3 for all  $n \geq 3$ . Moreover, a braid height representation can be realized by the braid on  $(2n - 2)$  strings,*

$$H_n = [\sigma_1\sigma_3 \cdots \sigma_{2n-3}][\sigma_1\sigma_3 \cdots \sigma_{2n-3}][\sigma_2\sigma_4 \cdots \sigma_{2n-4}].$$

*Proof.* We will proceed by mathematical induction on  $n$ . A composition of two Hopf links is given by  $[\sigma_1\sigma_3][\sigma_1\sigma_3][\sigma_2]$  which, if we examine the closure, is clearly the 3-component Hopf link as shown in Figure 7.5.

Suppose we know that the braid height of the  $k$ -component Hopf link is 3 and that a braid height representation is  $H_k = [\sigma_1\sigma_3 \cdots \sigma_{2k-3}][\sigma_1\sigma_3 \cdots \sigma_{2k-3}][\sigma_2\sigma_4 \cdots \sigma_{2k-4}]$ . Let



**Figure 7.5:** A braid representation of the 3-component Hopf link.

us lay a Hopf link braid next to  $H_k$ , adding two strings and then composing  $H_k$  with  $\sigma_{2k-1}^2$ . Thus we have a  $2k$ -string braid. The composition of these two links is obtained by further composing this braid with  $\sigma_{2k-2}$ . We claim that  $H_{k+1} = H_k \sigma_{2k-1}^2 \sigma_{2k-2}$ .

Through repeated application of braid move 1, we can commute  $\sigma_{2k-1}$  with any generator of  $H_k$ . In particular, we get

$$H_k \sigma_{2k-1}^2 \sigma_{2k-2} = [\sigma_1 \sigma_3 \cdots \sigma_{2k-3} \sigma_{2k-1}] [\sigma_1 \sigma_3 \cdots \sigma_{2k-3} \sigma_{2k-1}] [\sigma_2 \sigma_4 \cdots \sigma_{2k-4}] [\sigma_{2k-2}].$$

Additionally, we can include  $\sigma_{2k-2}$  in the final braid level because  $|(2k-4) - (2k-2)| = 2 > 1$  and thus obtain  $H_{k+1}$  as desired. By Theorem 7.2, we know that  $bh(H_{k+1}) \geq \max\{bh(H_k), bh(H_2)\} = \max\{3, 2\} = 3$ . Since we have a braid representation of the  $(k+1)$ -component Hopf link with 3 braid levels,  $bh(H_{k+1}) = 3$ .  $\square$

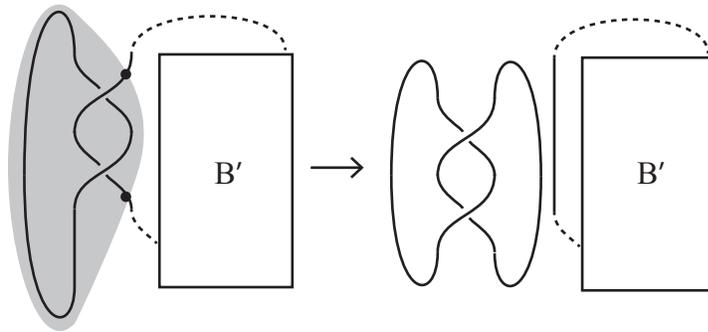
Using this fact, we can classify all the links of braid height 3.

**Theorem 7.5** *Let  $L$  be a non-trivial non-split link. If  $bh(L) = 3$  then  $L$  is either the trefoil knot or the  $n$ -component Hopf link.*

*Proof.* Let  $B$  be a braid word for a braid height representation of  $L$  that uses  $n$  strings. As we have seen already,  $\sigma_1^3$  and  $\sigma_1^{-3}$  produce the trefoil knots (Figure 2.17). Thus, if  $n = 2$ , it follows from Lemma 7.2 that  $L$  is a trefoil knot. Using Proposition 7.2,

another possibility is the composition of Hopf links. So suppose we have a link of braid height 3 on  $n > 2$  strings.

Since  $L$  is non-split, every  $\sigma_i$  must be applied in  $B$  by Lemma 7.1. Moreover, every generator cannot be applied more than twice without producing the trefoil knot again. If every  $\sigma_i$  is applied exactly once, then we can trivially untwist  $L$  to get the unknot, a knot of braid height 0. So there must be some  $\sigma_i$  that is applied twice in  $B$ . Pick  $k$  to be the minimal  $i$  such that  $\sigma_i$  appears twice. Then, since  $L$  is non-split, the first  $k$  strings of  $B$  are trivially twisted in the closure of  $B$ . So  $\sigma_1, \sigma_2, \dots, \sigma_{k-1}$  all appear exactly once and we perform  $k-1$  Reidemeister I moves to remove these  $k-1$  generators. Since  $\sigma_k$  appears twice, we are able to decompose  $B$  into the braids  $\sigma_1^2$  and  $B'$  as shown in Figure 7.6.



**Figure 7.6:** Decomposition of a braid height 3 link.

We will ignore the Hopf link and focus on the remaining braid,  $B'$ . Since  $L$  was non-split, the closure of  $B'$  is also non-split. So we can apply the above process again to split off another Hopf link. Since  $B$  had  $n$  strings originally,  $L$  is the composition of at most  $\lfloor n/2 \rfloor$  Hopf links. Moreover, based on the decomposition method, every composition needed to reform  $L$  involves the end components of the respective Hopf links and so  $L$  was a multi-component Hopf link.  $\square$

Once we start considering links of braid height 4, it becomes less clear that we have considered every possible link. However, included in this list are the figure-eight knot

(given by  $\sigma_1\sigma_2^{-1}\sigma_1\sigma_2^{-1}$ ), the composition of trefoil knots, the composition of trefoil knots and Hopf links, and possibly more.

## 7.2 THE MACHETE NUMBER

Recall how, in the process of creating a Seifert surface, we smoothed every crossing of a link and the result was a set of unlinked circles. The machete number generalizes this process for braids and turns it into a link invariant. While the goal of smoothing was to trivialize a fixed projection, the machete number looks at how quickly a link can be reduced to the unlink. The essential idea is that we select a braid level and smooth every generator in that level. Visually, we can imagine a machete slicing through all the generators along the given horizontal line. To produce an invariant, we first require a formal definition of this generalization.

**Definition 7.3** *Let  $B$  be a braid and  $[\omega]$  a braid level of  $B$ . A ***machete cut*** of  $B$  is the braid  $B'$  that is obtained by smoothing every generator in  $\omega$ .  $B'$  is called the ***result of cutting***  $B$ .*

*If  $k \in \mathbb{N}$  and  $[\omega]$  contains at most  $k$  generators, a machete cut of  $B$  is called a ***k-cut*** of  $B$  and  $B'$  is called the ***result of k-cutting***  $B$ .*

If a given braid level has more than  $k$  generators, we can always trivially partition it into smaller braid levels. Let us consider a few examples of this cutting process.

**Example 7.2** *Let us focus on the braid on 5 strings with braid word*

$$B = [\sigma_4\sigma_1][\sigma_2][\sigma_3\sigma_1][\sigma_4].$$

*The closure of this braid is the 3-component Hopf link. With the given braid level partition, note that we cannot apply a 1-cut to either the first or third braid levels. However, we can apply a  $k$ -cut for any  $k \geq 2$  (or simply use a machete cut). Applying*

a 2-cut to  $[\sigma_4\sigma_1]$ , we obtain  $B' = [\sigma_2][\sigma_3\sigma_1][\sigma_4]$  which, when closed, is simply the unknot.

Suppose we had instead 2-cut  $[\sigma_3\sigma_1]$ . The result of this 2-cut is  $B'' = [\sigma_4\sigma_1][\sigma_2][\sigma_4]$ . It does not seem as though the choice of cut should have made a significant difference and yet the closure of  $B''$  is the disjoint union of the unknot and the Hopf link.

Finally, suppose we were interested in a 1-cut to  $B$ . We can take the braid level  $[\sigma_4\sigma_1]$  and break it into two braid levels,  $[\sigma_4][\sigma_1]$ . Let us apply a 1-cut to the first  $[\sigma_4]$  in the braid word, giving us  $B''' = [\sigma_1][\sigma_2][\sigma_3\sigma_1][\sigma_4]$ . A braid whose closure is the Hopf link.

Clearly, if we only consider 1-cuts of a braid, we have reduced cutting to simply smoothing crossings of the braid. Our second-to-last essential definition is the notion of cutting down a link.

**Definition 7.4** *Let  $L$  be a link,  $B$  a braid representation of  $L$ , and  $B'$  the result of applying a finite number of  $k$ -cuts to  $B$  for a fixed  $k$ . Then  $L$  is **cut down** if the closure of  $B'$  is the unlink with some number of components.*

An immediate consequence of this definition is the following:

**Proposition 7.3** *Every link can be cut down.*

*Proof.* By Theorem 6.1, we know that  $L$  has a braid representation  $B$  with  $m$  generators and  $n$  strings. Decompose  $B$  into braid levels which each contain a single generator. By applying  $m$   $k$ -cuts to  $B$ , one for each braid level, we have the empty braid on  $n$  number of strings. Since the closure of the empty braid is the unlink of  $n$  components,  $L$  has been cut down.  $\square$

For an example, let us turn back to the braid in Example 7.2. We cut down  $B$  with our first choice of a 2-cut but not our second choice. However, this recent proposition tells us that it was possible to cut  $B''$  in some way such that the closure eventually

produced the trivial link. At long last, using the notion of cutting down links, we can formally define the machete number.

**Definition 7.5** *The **machete number** of  $L$ , denoted  $\mathcal{M}(L)$ , is the smallest  $n \in \mathbb{N}$  such that there is some braid representation of  $L$  which can be cut down in  $n$  machete cuts.*

*For a fixed  $k \in \mathbb{N}$ , the  **$k$ -machete number** of a link  $L$ , denoted  $\mathcal{M}_k(L)$ , is the smallest  $n \in \mathbb{N}$  such that there is some braid representation of  $L$  which can be cut down in  $n$   $k$ -cuts.*

The fact that the  $k$ -machete number is a link invariant is based entirely on the minimality statement in the definition and Proposition 7.3. Additionally, we can relate the different  $k$ -machete numbers.

**Proposition 7.4** *For any link  $L$  and any  $k \in \mathbb{N}$ ,  $\mathcal{M}_{k+1}(L) \leq \mathcal{M}_k(L)$ .*

*Proof.* Pick a sequence of  $\mathcal{M}_k(L)$   $k$ -cuts which cuts down  $L$ . By Definition 7.3, each  $k$ -cut is applied to a braid level with at most  $k$  generators. Trivially, each level has at most  $(k+1)$  generators. Hence each  $k$ -cut can be replicated with a  $(k+1)$ -cut and so  $L$  can be reduced in  $\mathcal{M}_k(L)$   $(k+1)$ -cuts. Thus  $\mathcal{M}_{k+1}(L) \leq \mathcal{M}_k(L)$ .  $\square$

Since machete cuts can cut through a braid level of any size, the following corollary is trivial.

**Corollary 7.2** *For any link  $L$ ,  $\mathcal{M}(L) = \min_k \{\mathcal{M}_k(L)\}$ .*

In Example 7.2, we found a single 2-cut which cut down  $B$ . Hence  $\mathcal{M}_2(B) \leq 1$ . By Corollary 7.2, we know that  $\mathcal{M}(B) \leq 1$  as well. As we will soon show, this is actually sharp and so  $\mathcal{M}(B) = 1$ . To demonstrate this fact, we will also prove a very important property of the machete number: it distinguishes between trivial and non-trivial links.

**Proposition 7.5** *For any non-trivial link  $L$ ,  $\mathcal{M}_k(L) \geq 1$ .*

*Proof.* Since  $L$  is non-trivial, there is no braid representation of  $L$  that is trivial. Hence we must apply at least one  $k$ -cut and so  $\mathcal{M}_k(L) \geq 1$ .  $\square$

Without Proposition 7.5, the machete number would be relatively useless as an invariant. It tells us that the machete number distinguishes between trivial links and non-trivial links. Although some link invariants, such as the linking number, cannot make this distinction, it is comforting when an invariant successfully determines the triviality of a link. An additional simplification, analogous to braid height, is that we can reduce the machete number problem to non-split links.

**Proposition 7.6** *Suppose that  $L$  is a split link where  $L$  can be split into  $L_1$  and  $L_2$ . Then  $\mathcal{M}(L) = \max \{\mathcal{M}(L_1), \mathcal{M}(L_2)\}$ .*

*Proof.* Suppose  $B_i$  is a braid representation of  $L_i$  for  $i = 1, 2$ . Laying  $B_1$  and  $B_2$  next to one another and considering them as a single braid, we have a braid representation of  $L$ ,  $B_3$ . We can use braid move 3 and conjugation to line up any given braid level of  $B_1$  with a level of  $B_2$ . In  $B_3$ , since  $B_1$  and  $B_2$  do not share strings, we can extend the braid level to include all of these braid generators. We then cut down  $B_3$  by following a sequence of machete cuts that minimally cuts down both  $B_1$  and  $B_2$ .  $\square$

It is unfortunate that this same result does not apply to the  $k$ -machete number in general, but it is also a significantly more controlled process.

Before we turn our discussion toward bounds on the machete number, we will demonstrate that the machete number cannot be realized from specific types of link projections. Our next example demonstrates that braid representations of a link  $L$  with  $c(L)$  crossings cannot necessarily be cut down minimally.

**Example 7.3** *A crossing-minimal braid projection of the 3-component Hopf link is given by  $B = \sigma_1^2 \sigma_2^2$ . In this projection, each braid level necessarily consists of a single generator and so applying any cut to the braid will remove a single generator. Also independent of the chosen generator, the closure of the braid resulting from a cut of*

any size will be the Hopf link. Since the Hopf link is not the unlink, a second cut is necessary to cut down  $B$ . However, using the braid representation from Proposition 7.2, we have  $B' = [\sigma_1\sigma_3][\sigma_1\sigma_3][\sigma_2]$ . We can cut down  $B'$  by cutting the first braid level and so the machete number of the 3-component Hopf link is  $1 \neq 2$ .

Example 7.3 also demonstrates that, for an arbitrary link  $L$ , the machete number cannot necessarily be realized on a  $b(L)$ -string braid representation of  $L$ . Despite almost certainty that the machete number cannot necessarily be realized in a braid height representation, we have not shown it conclusively.

### 7.3 BOUNDING THE MACHETE NUMBER

In order to demonstrate the usefulness of the machete number, it is necessary to be able to compute the value. Unfortunately, in all but the most trivial cases, it remains unknown. However, we will be able to prove a variety of upper bounds on the machete number. Using braid height, we can establish a simple upper bound.

**Proposition 7.7** *For any link  $L$ ,  $\mathcal{M}(L) \leq bh(L) - 1$ .*

*Proof.* Let  $B$  be a braid height minimal presentation of  $L$ . Suppose we machete cut  $bh(L) - 1$  levels of  $B$  and the result is  $B'$ . By the contrapositive of Proposition 7.1, the closure of  $B'$  must be trivial. Hence  $B$  was cut down in  $bh(L) - 1$  machete cuts and so  $\mathcal{M}(L) \leq bh(L) - 1$ . □

While the bound given by Proposition 7.7 is sharp for the Hopf link, it is not difficult to construct an infinite family of links for which it is not tight. However, at the same time, we will find an infinite family of links with machete number 1.

**Proposition 7.8** *The  $n$ -component Hopf link has machete number 1.*

*Proof.* Let  $n \geq 2$  and consider the braid representation of the  $n$ -component Hopf link used in Proposition 7.2. Applying a machete cut to the first braid level, the resulting braid is the  $(2n - 2)$ -string braid given by

$$B' = [\sigma_1\sigma_3 \cdots \sigma_{2n-3}][\sigma_2\sigma_4 \cdots \sigma_{2n-4}].$$

Note that every generator in  $B'$  is used exactly once. Specifically,  $\sigma_{2n-2-1}$  is used exactly once and so string  $2n - 2$  can be removed through stabilization. We can now repeat this stabilization process, on progressively smaller strings,  $2n - 3$  more times to obtain the empty braid on one string. Thus we cut down the  $n$ -component Hopf link in a single machete cut and, since it is a non-trivial link, we have that it has machete number 1.  $\square$

Since the  $n$ -component Hopf link has braid height 3 and machete number 1, we have an infinite family of links for which Proposition 7.1 is not sharp.

Let us look at the family of links with braid index 2. Such links necessarily have a braid representation  $\sigma_1^n$  for some  $n \in \mathbb{N}$ . These links are also referred to as the  $(n, 2)$ -torus links [5]. Without going into too much detail, the idea of a torus link is that they are links which can be wrapped around the surface of the torus without self-intersection [1]. A so-called  $(m, n)$ -torus link can be represented on the  $n$ -string braid by the braid word  $T(m, n) = (\sigma_1\sigma_2 \cdots \sigma_{n-1})^m$  [1, 5]. Moreover, it turns out  $T(m, n)$  and  $T(n, m)$  are Markov-equivalent braids [1]. Using these facts, we are able to prove a simple upper bound for the machete number of an  $(m, 2)$ -torus link.

**Proposition 7.9** *For  $m > 1$ , let  $L$  be an  $(m, 2)$ -torus link. Then  $\mathcal{M}_2(L) \leq \lceil m/2 \rceil$ .*

*Proof.* We will proceed by mathematical induction on  $m$ . If  $m = 2$  then  $L$  is the Hopf link and so  $\mathcal{M}(L) = 1 = \lceil 2/2 \rceil$ . Suppose that the bound holds for all  $k < m$ . Let  $L$

be the closure of  $T(m, 2) = \sigma_1^m$ . Then, since  $T(m, n) = T(n, m)$ , an equivalent braid representation of  $L$  is

$$T(2, m) = (\sigma_1 \cdots \sigma_{m-1})^2 = \sigma_1 \cdots \sigma_{m-2} [\sigma_{m-1} \sigma_1] \sigma_2 \cdots \sigma_{m-1}.$$

Applying a 2-cut to  $[\sigma_{m-1} \sigma_1]$ , we stabilize the resulting braid to obtain

$$\sigma_1 \cdots \sigma_{m-2} \sigma_2 \cdots \sigma_{m-2}.$$

In the closed braid, the  $\sigma_1$  is also a trivial crossing and can be removed by a Reidemeister I move. Relabeling the generators, this new braid has braid word

$$\sigma_1 \cdots \sigma_{m-3} \sigma_1 \cdots \sigma_{m-3} = T(2, m-2).$$

By the inductive hypothesis,  $\mathcal{M}_2(T(2, m-2)) \leq \lceil \frac{m-2}{2} \rceil$ . Thus  $\mathcal{M}_2(L) \leq \lceil \frac{m-2}{2} \rceil + 1 = \lceil \frac{m}{2} \rceil$ .  $\square$

Extending some of the ideas of this proof, we obtain a general upper bound for the machete number of a torus link.

**Proposition 7.10** *For  $m, n \in \mathbb{N}$ ,*

$$\mathcal{M}(T(m, n)) \leq \min \{ \mathcal{M}(T(m-1, n)) + \lceil n/2 \rceil, \mathcal{M}(T(m, n-1)) + (n-1) \}.$$

Before turning to the proof of this proposition, let us consider an example of the process we will generalize to obtain  $\mathcal{M}(T(m, n)) \leq \mathcal{M}(T(m-1, n)) + \lceil n/2 \rceil$ .

**Example 7.4** *Suppose that  $m = 2$  and  $n = 5$ . Then*

$$T(2, 5) = (\sigma_1 \sigma_2 \sigma_3 \sigma_4)^2 = \sigma_1 \sigma_2 \sigma_3 [\sigma_4 \sigma_1] \sigma_2 \sigma_3 \sigma_4.$$

Cutting  $[\sigma_4\sigma_1]$ , we obtain the braid  $\sigma_1\sigma_2\sigma_3\sigma_2\sigma_3\sigma_4$ . Applying a machete cut to  $\sigma_2$  and another machete cut to  $\sigma_3$ , our result is  $\sigma_1\sigma_2\sigma_3\sigma_4 = T(1, 5)$ . In the process, we used  $3 = \lceil 5/2 \rceil$  machete cuts and so we require at most  $\mathcal{M}(T(1, 5))$  more cuts to cut down  $T(2, 5)$ .

Now suppose that  $m = 2$  and  $n = 6$ . Then

$$T(2, 6) = (\sigma_1\sigma_2\sigma_3\sigma_4\sigma_5)^2 = \sigma_1\sigma_2\sigma_3\sigma_4[\sigma_5\sigma_1]\sigma_2\sigma_3\sigma_4\sigma_5.$$

Cutting  $[\sigma_5\sigma_1]$ , the resulting braid is  $\sigma_1\sigma_2\sigma_3[\sigma_4\sigma_2]\sigma_3\sigma_4\sigma_5$ . Cutting  $[\sigma_4\sigma_2]$ , the simpler braid is  $\sigma_1\sigma_2\sigma_3\sigma_3\sigma_4\sigma_5$ . Lastly, cutting a  $\sigma_3$ , we get  $\sigma_1\sigma_2\sigma_3\sigma_4\sigma_5 = T(1, 6)$ . In the process, we used  $3 = \lceil 6/2 \rceil$  machete cuts and so we require at most  $\mathcal{M}(T(1, 6))$  more cuts to cut down  $T(2, 6)$ .

*Proof (Proposition 7.10).* If  $m = 1$  or  $n = 1$  then  $T(m, n)$  is a trivial knot and so it has machete number 0. So we can assume that  $m, n > 1$ .

First we will show that we can obtain  $T(m - 1, n)$  from  $T(m, n)$  in  $\lceil n/2 \rceil$  machete cuts. We know that  $T(m, n) = (\sigma_1\sigma_2 \cdots \sigma_{n-1})^m$ . If we have  $|(n - 1) - 1| > 1$  then, since  $m \geq 2$ , consider the subsequence

$$B = \sigma_1\sigma_2 \cdots [\sigma_{n-1}\sigma_1]\sigma_2 \cdots \sigma_{n-1}.$$

Cutting the braid level  $[\sigma_{n-1}\sigma_1]$ , we obtain  $B_1 = \sigma_1\sigma_2 \cdots \sigma_{n-2}\sigma_2 \cdots \sigma_{n-1}$ . If  $|(n - 2) - 2| > 1$ , we can partition  $B_1$  so that  $[\sigma_{n-2}\sigma_2]$  is a braid level. Then we cut that braid level to obtain  $B_2$ .

In general, in the  $k$ th step, we have  $B_k = \sigma_1\sigma_2 \cdots \sigma_{n-k}\sigma_k \cdots \sigma_{n-1}$  and have performed  $k - 1$  machete cuts. So long as  $|(n - k) - k| > 1$ , we can repeat the above procedure to produce  $B_{k+1}$ . Otherwise, we cannot repeat this process. Since  $n$  is finite, it will stop when  $|(n - k) - k| \leq 1$ . By our process,  $n - k \geq k$ .

If  $|n - 2k| = 1$  then  $n - 2k = 1$ . So  $n$  is odd and  $k = \frac{n-1}{2}$ . Thus  $k = \lfloor n/2 \rfloor$  and  $n - k = \lceil n/2 \rceil$ . Hence

$$B_k = \sigma_1 \cdots \sigma_{\lfloor n/2 \rfloor} [\sigma_{\lceil n/2 \rceil}] [\sigma_{\lfloor n/2 \rfloor}] \sigma_{\lceil n/2 \rceil} \cdots \sigma_{n-1}.$$

Using two machete cuts to remove  $[\sigma_{\lceil n/2 \rceil}] [\sigma_{\lfloor n/2 \rfloor}]$ , we obtain  $B_* = \sigma_1 \cdots \sigma_{n-1}$ . In total, we applied  $k - 1 + 2 = \lfloor n/2 \rfloor + 1 = \lceil n/2 \rceil$  machete cuts to  $B$ .

If  $|n - 2k| = 0$  then  $n = 2k$ . So  $n$  is even,  $k = \frac{n}{2}$ , and  $n - k = \frac{n}{2}$ . Hence

$$B_k = \sigma_1 \cdots \sigma_{n/2} [\sigma_{n/2}] \cdots \sigma_{n-1}.$$

Cutting  $[\sigma_{n/2}]$ , we obtain  $B_* = \sigma_1 \cdots \sigma_{n-1}$ . In total, we applied  $k - 1 + 1 = n/2 = \lceil n/2 \rceil$  machete cuts to  $B$ .

In both cases, we applied  $\lceil n/2 \rceil$  machete cuts to  $T(m, n)$  to obtain

$$B_*(\sigma_1 \sigma_2 \cdots \sigma_{n-1})^{m-2} = (\sigma_1 \sigma_2 \cdots \sigma_{n-1})^{m-1} = T(m-1, n).$$

Hence  $\mathcal{M}(T(m, n)) \leq \mathcal{M}(T(m-1, n)) + \lceil n/2 \rceil$ .

Now we will show that we can obtain  $T(m, n-1)$  from  $T(m, n)$  in  $m-1$  machete cuts. We know that the generator  $\sigma_{n-1}$  appears a total of  $m$  times in  $T(m, n)$ . Let us consider the first  $m-1$  such generators to be on their own braid level and cut each of these levels. This requires exactly  $m-1$  cuts. The result is a braid on  $n$  strings with braid word  $(\sigma_1 \sigma_2 \cdots \sigma_{n-2})^m \sigma_{n-1}$ . Since  $\sigma_{n-1}$  appears only once, we stabilize this braid to obtain the braid on  $n-1$  strings given by

$$(\sigma_1 \sigma_2 \cdots \sigma_{n-2})^m = T(m, n-1).$$

Thus  $\mathcal{M}(T(m, n)) \leq \mathcal{M}(T(m, n-1)) + (m-1)$ . Since the machete number is the minimal number of cuts required, we take the minimum of our two upper bounds.  $\square$

Note that, using only Proposition 7.10, we may not trivially obtain Proposition 7.9. We know that the machete number of the Hopf link, equivalently the  $(2, 2)$ -torus link, is 1. For the trefoil knot, also known as the  $(3, 2)$ -torus link, The smallest upper bound given by Proposition 7.10 matches the bound of Proposition 7.9. So  $\mathcal{M}(T(3, 2)) \leq 2$ . However, let us consider the  $(4, 2)$ -torus link. In this case,  $\mathcal{M}(T(4, 2)) \leq \min \{ \mathcal{M}(T(3, 2)) + 1, \mathcal{M}(T(4, 1)) + 3 \}$ . Since we do not currently know  $\mathcal{M}(T(3, 2))$ , this upper bound is either 2 or 3. In the latter case, the bound given by Proposition 7.9 is tighter as  $\lceil 4/2 \rceil = 2$ .

Since  $T(m, n) = T(n, m)$ , we can immediately improve Proposition 7.10 by considering the bounds for both  $T(m, n)$  and  $T(n, m)$ . Moreover, since the proof of Proposition 7.10 used 2-cuts only, we can restate it in terms of the 2-machete number instead.

**Corollary 7.3** For  $m, n \in \mathbb{N}$ ,

$$\mathcal{M}_2(T(m, n)) \leq \min \{ \mathcal{M}_2(T(m-1, n)) + \lceil n/2 \rceil, \mathcal{M}_2(T(m, n-1)) + (m-1) \}.$$

Finally, since the process of removing one string from the braid  $T(m, n)$  required 1-cuts in the proof of Proposition 7.10, we can bound the 1-machete number of a general torus link using the same process and the fact that  $T(m, n) = T(n, m)$ .

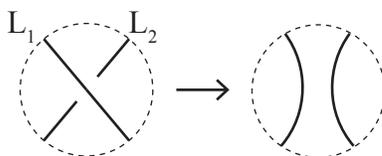
**Corollary 7.4** For  $m, n \in \mathbb{N}$ ,

$$\mathcal{M}_1(T(m, n)) \leq \min \{ \mathcal{M}_1(T(m-1, n)) + (n-1), \mathcal{M}_1(T(m, n-1)) + (m-1) \}.$$

## 7.4 CONCLUSIONS AND FUTURE WORK

Despite the fact that our upper bounds for the machete number were obtained using elementary methods, we also explored other methods of bounding the machete number. Some of these approaches included attempting to use the linking number, Alexander polynomial, and the HOMFLY-PT polynomial. It requires a painfully small amount of work to show that the linking number and Alexander polynomial are useless when bounding the machete number.

We know that a split link has linking number 0 and, since the machete number eventually cuts down a braid, the final result must be a split link. Hence, for a non-zero linking number, it seems as though we should be able to obtain a lower bound. However, the nature of smoothing poses a problem. Suppose you start with a nontrivial link of at least two components and smooth one of the crossings between two components. Then the two components become a knot after smoothing, as shown in Figure 7.7. Although it may seem as though this process could leave the components unaffected, a strand going into a link component must also leave that same component at some point and so the interactions of any two links will occur at an even number of crossings. Thus, in the case of 2-component links, smoothing makes the linking number a useless measure – a knot necessarily has linking number 0.



**Figure 7.7:** Effect of smoothing on strands of different components.

The Alexander polynomial also suffers from the nature of smoothing. As we know from Theorem 5.3, a split link has Alexander polynomial 0. For a braid on at least 3 strings, it is possible to apply machete cuts to the braid such that we have a split

link projection prior to cutting down the braid. At the point of obtaining a split link projection, we have lost any possible information from the Alexander polynomial.

While the HOMFLY-PT polynomial has been used to bound the braid index of a link, as in Theorem 6.3, it does not appear that we can use the degree difference of either the variable  $l$  or  $m$  to tell us anything about the machete number. However, all of this is based on anecdotal evidence, and it is possible that some information can be obtained. Afterall, the HOMFLY-PT polynomial distinguishes between at least some split links.

One calculation involving links that still shows promise is the writhe. Unfortunately, in general, the writhe is not a link invariant. Recall that the linking number is a sum over the signs of crossings between components. Writhe generalizes this notion by summing over the signs of all crossings in a link [1, 2]. This means that the writhe varies whenever a Reidemeister I move is applied to the link, and so it is not a link invariant. However, writhe is known to be an invariant for specific classes of braids. It is currently unclear what role the writhe plays in the machete number computation, but it seems to relate to expected machete numbers for certain links.

Given our relative lack of success in bounding and accurately determining the machete number of a large number of links, we conclude with a list of conjectures that arose during our investigation of the machete number. The first such conjecture seems only reasonable for a link invariant that should measure link complexity in some form.

**Conjecture 7.1** *Let  $L$  be a link and  $L'$  the result of cutting a braid form of  $L$ . Then  $\mathcal{M}(L') \leq \mathcal{M}(L)$ .*

A proof of this conjecture would confirm that the machete number truly measures link complexity. Unfortunately, especially since we have had trouble determining the machete number of even a special class of link, it has been difficult to either prove this statement or find a good counterexample.

Our second conjecture is the bound that seems to be given by the writhe.

**Conjecture 7.2** *Let  $B$  be a braid such that the writhe of the braid,  $\omega(B)$ , is an invariant. Then  $\mathcal{M}(B) \geq \frac{\omega(B)}{2}$ .*

Given that there may be a relationship between braid height and the machete number, a proof of the following conjecture may also be useful.

**Conjecture 7.3** *The braid height of the  $(m, n)$ -torus link can be realized in either  $T(m, n)$  or  $T(n, m)$ .*

Unfortunately, Conjecture 7.3 is all but speculation. However, given that many link invariants can be realized in the braid representation of the  $(m, n)$ -torus link, it seems plausible that braid height might be as well [1, 2].

The fourth and last conjecture on our list also relates back to the torus links, although it is specific to the  $(m, 2)$ -torus links.

**Conjecture 7.4** *The bound in Proposition 7.9 is sharp.*

While Proposition 7.10 is almost certainly not sharp for all torus links, Proposition 7.9 seems to reduce  $(m, 2)$ -torus links very efficiently. Even if it is not sharp for all  $(m, 2)$ -torus links, we suspect that it will be sharp for some of the early ones.



## REFERENCES

1. Colin Adams. *The Knot Book*. American Mathematical Society, Providence, RI, 2004.
2. Peter Cromwell. *Knots and Links*. Cambridge University Press, Cambridge, United Kingdom, 2004.
3. Christian Kassel and Vladimir Turaev. *Braid Groups*. Springer, New York, NY, 2008.
4. Charles Livingston. *Knot Theory*. The Mathematical Association of America, Washington, DC, 1993.
5. Kunio Murasugi. *Knot Theory & Its Applications*. Birkhäuser, New York, NY, 1996.
6. Dale Rolfsen. *Knots and Links*. AMS Chelsea Publishing, Providence, RI, 2003.