Maypole Braids: An Analysis Using the Annular Braid Group

By

Moyi Tian

Submitted in partial fulfillment of the requirements
for departmental honors in Mathematics
Dickinson College, 2018-2019

David Richeson, Adviser
Jeffrey Forrester, Reader
Barry Tesman, Reader

May 10, 2019

The Department of Mathematics and Computer Science at Dickinson College hereby accepts this senior honors thesis by Moyi Tian, and awards departmental honors in Mathematics

David Richeson (Advisor) Date

Jeffrey Forrester (Reader) Date

Barry Tesman (Reader) Date

Grant Braught (Department chairperson) Date

Department of Mathematics and Computer Science
Dickinson College

May 10, 2019

Abstract

Maypole Braids: An Analysis Using the Annular Braid Group

by

Moyi Tian

The study of braids started in the early 20^{th} century with the motivation of revealing properties of knots and links. The Artin braid group gives an algebraic tool to analyze the braid actions and the equivalence of braids. Later, a variation of ordinary braids, the annular braids, was introduced with additional rules added. In this thesis, we give three presentations to describe the annular braid group. We also use the annular braid group as a medium to abstract the braids in maypole dances and therefore apply an algebraic analysis. Finally, we discuss some essential properties embedded in the maypole braids, which are related to the invariants of annular braids - the crossing number and the step number.

Acknowledgments

I would like to express my great gratitude to my thesis advisor Professor David Richeson for giving me patient guidance, encouraging support and valuable insights to my research. I would also like to thank the Department of Mathematics and Computer Science and my peers for the constructive conversations and the supportive environment provided during the research process.

Contents

1. Introduction

A *braid* is a set of n strings attached to two horizontal bars where one is at the top and the other at the bottom. As we move along any one of the strings from the top to the bottom, the string is always heading downwards. The concept of braids first came about because of the motivation to provide insights into *knots* and *links* ([Sti93]).

Braids have a natural group structure. Emil Artin introduced braid groups in 1926, which gives a way to present ordinary braids using generators, relations and a binary operation. The ordinary braid group, known as the *Artin braid group*, opens the door for people to understand braids algebraically. It enables people to describe a braid by its crossings, which refers to the word for a braid.

The idea of annular braids is similar to the one of ordinary braids except that the strands run from a top circle to a bottom circle instead of bars and there is a pole running down the center. Cromwell explained the related topological perspective using cylindrical tangles and the composition rule in [Cro04]. As we shall see, there are different algebraic presentations to describe the annular braid group.

Maypole dances, which happen in some European May Day festivals, are analogous to annular braids. A maypole is a wooden pole which usually has ribbons attached to its top. In a maypole dance, each person holds a ribbon standing underneath the pole and follows some designated rules to move past the other people while walking in a circle. While some dancers move clockwise, some move counterclockwise, some move over the ones they encounter, and some move under the people coming in the opposite direction. The ribbons gradually braid around the pole and finally form a special decoration on it with an interesting pattern. The ribbons in a maypole dance can be treated as strands in annular braids, and the crossing-overs of the ribbons during a dance can be described by generators that are similar to Artin's approach for braid groups. Overall, the set of all different maypole dances forms a group which can be called maypole braid groups.

This thesis will investigate three different algebraic presentations of the annular braid group with one of which proved explicitly, use the annular braid to describe and analyze some traditional maypole dances, and discuss the invariants of braids - the crossing number and the step number - that are inspired by the maypole braids.

2. Background of braids

The study of braids is closely related to the knot problem. We can think of a *knot* as a knotted loop of a string that has no thickness and never intersects itself. Knots live in the three dimensional space, but we can project a knot onto the two dimensional plane (see Figure 2.1). Several knots that tangle up together will result in a *link* (see Figure 2.2).

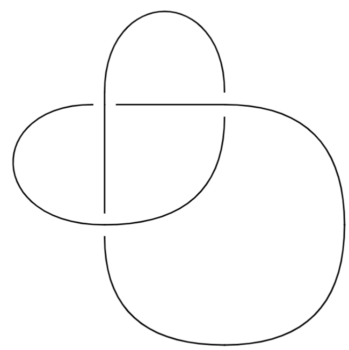

Figure 2.1. An example of the projection of a knot. This is known as the trefoil knot.

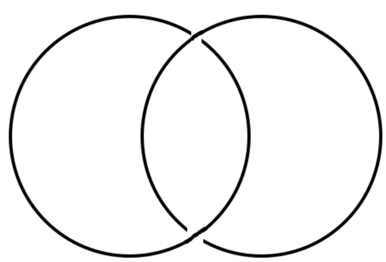

Figure 2.2. An example of link. This is known as the Hopf link.

A knot can be thought of as a rubber band, as it is highly deformable. Two knots are equivalent if there is an ambient isotopy from one to the other in $\mathbb{R}^3$. There are several major ways by which we can deform a knot projection without changing the knot itself. One is called *planar isotopy*. This allows us to shrink, expand or deform any part of the projection in the plane (see Figure 2.3). The others are the *Reidemeister moves*. There are three types of Reidemeister moves (see Figure 2.4). If two projections are the same knot, we can get from one to the other by applying planar isotopies and a finite number of Reidemeister moves.

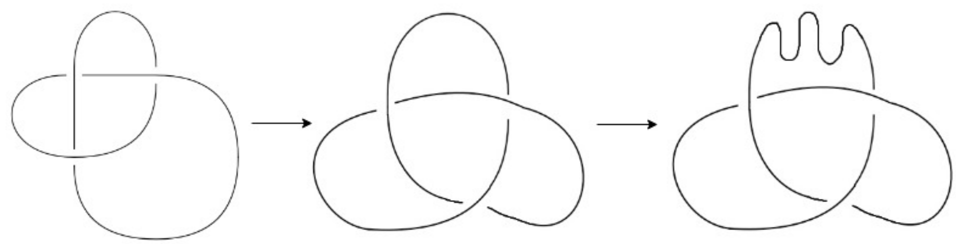

FIGURE 2.3. Planar isotopies of the trefoil knot

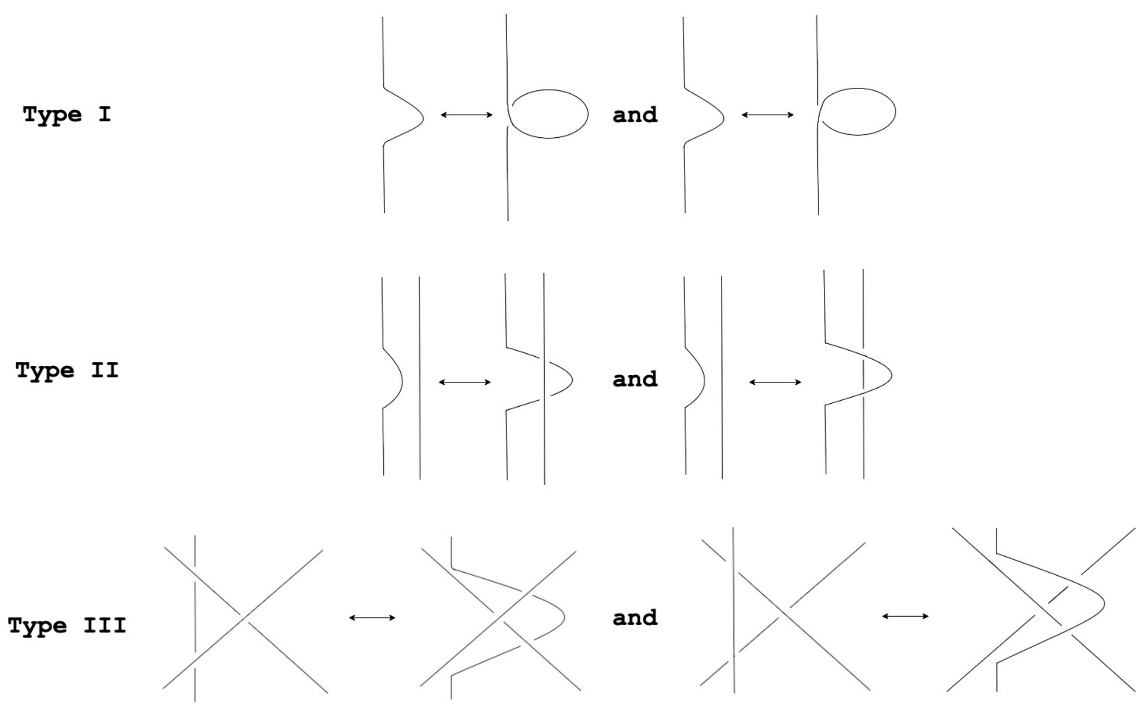

FIGURE 2.4. Three types of Reidemeister moves

Similarly, we can have an intuitive understanding of *braid*: a braid is a set of n strings attaching to two horizontal bars where one is at the top and the other is at the bottom, and as we move along any one of the strings from the top to the bottom, the string is always heading downwards (see Figure 2.5). Like knots, two braids are equivalent if and only if there is an ambient isotopy between them.

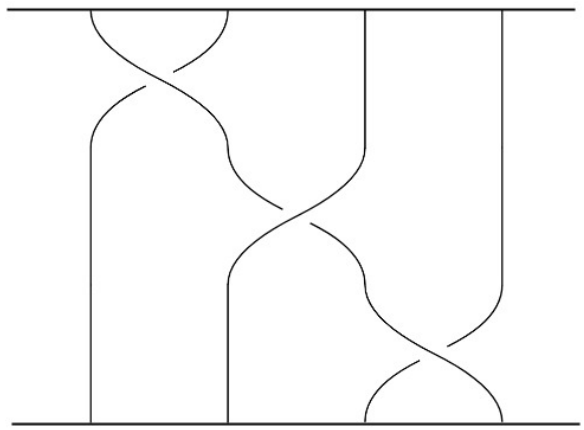

FIGURE 2.5. An example of braid

Since a braid has its physical presentation, it is straightforward to describe a braid geometrically as in Figure 2.5. The technical definition of a *geometric braid* is as follows.

Definition 2.1. A *geometric braid* on $n \geq 1$ strings is a set $b \subset \mathbb{R}^2 \times I$ formed by n disjoint topological intervals called the *strings* of b such that the projection $\mathbb{R}^2 \times I \to I$ maps each string homeomorphically onto I and

$$b \cap (\mathbb{R}^2 \times \{0\}) = \{(1,0,0), (2,0,0), \dots, (n,0,0)\},$$

$$b \cap (\mathbb{R}^2 \times \{1\}) = \{(1,0,1), (2,0,1), \dots, (n,0,1)\}.$$

Every string of b meets each plane $\mathbb{R}^2 \times \{t\}$ with $t \in I$ in exactly one point and connects a point $(i,0,0)$ to a point $(s(i),0,1)$, where $i, s(i) \in \{1,2,\dots,n\}$. The sequence $(s(1), s(2), \dots, s(n))$ is a permutation of the set $\{1,2,\dots,n\}$, which is an element of the symmetric group S_n, called the *underlying permutation* of b.

Definition 2.1 relates braids to elements in the symmetric group S_n. We will make this precise in Theorem 3.3.

If we draw out the three axes, then it shows as Figure 2.6. However, for simplification, we usually omit the axes.

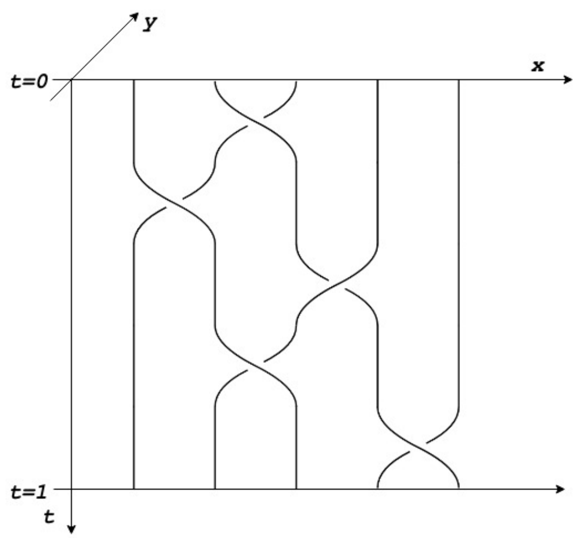

FIGURE 2.6. A geometric braid on five strings

Moreover, we can obtain the closure of a braid by gluing the top and the bottom of the braids together. As a result, every braid produces a knot or link. In 1923, James Alexander proved that every knot or link has a closed braid representation. The *braid index*, which is defined as the least number of strings of a braid that corresponds to the closed form, is an invariant for knots and links. For more details on the connection between knots and braids, see [BB05] and [Liv93].

3. Background of braid groups

The natural group structure in braids was first introduced by Emil Artin in 1926. He was aiming to tackle the knot problem ([Sti93]), which asks whether a given knot is trivial (the unknot).

Before talking about the algebraic definition of braid groups, we first consider the geometric presentations of braids. Define the *geometric braid group* with n strands as $\mathcal{B}_n$ and the elementary braids $s_i^+, s_i^- \in \mathcal{B}_n$ for $i = 1, 2, \ldots, n-1$ as shown in Figure 3.7.

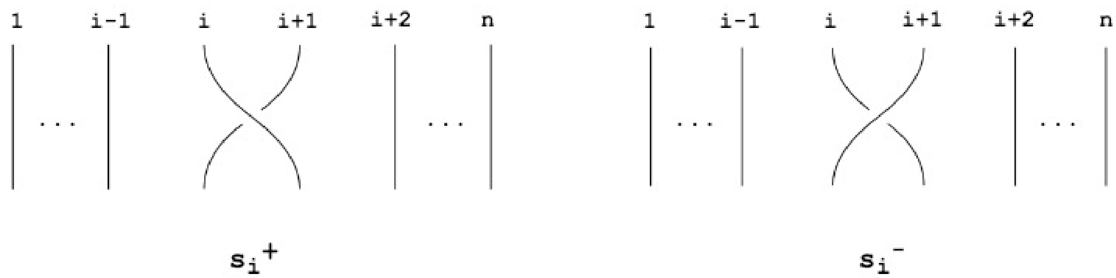

Figure 3.7. The elementary braids s_i^+ and s_i^-

The braids $s_1^+, s_2^+, \ldots, s_{n-1}^+, s_1^-, s_2^-, \ldots, s_{n-1}^- \in \mathcal{B}_n$ generate $\mathcal{B}_n$ as a group. Two braids are equivalent if there is an ambient isotopy between them in $\mathbb{R}^3$. A binary operation on braids can be defined as shrinking both of the original braids to half of their heights and composing these two together by connecting the strands at the bottom of the first braid to the corresponding strands at the top of the second (see Figure 3.8).

The identity element in $\mathcal{B}_n$ is the braid that consists of n parallel strands from top to bottom. The inverse of a certain braid undoes the twists of this braid. It can be obtained by flipping the top and bottom by $180°$ with respect to the middle horizontal line at $t = \frac{1}{2}$ on the xt-plane. Multiplying the inverse to the braid gives the identity element.

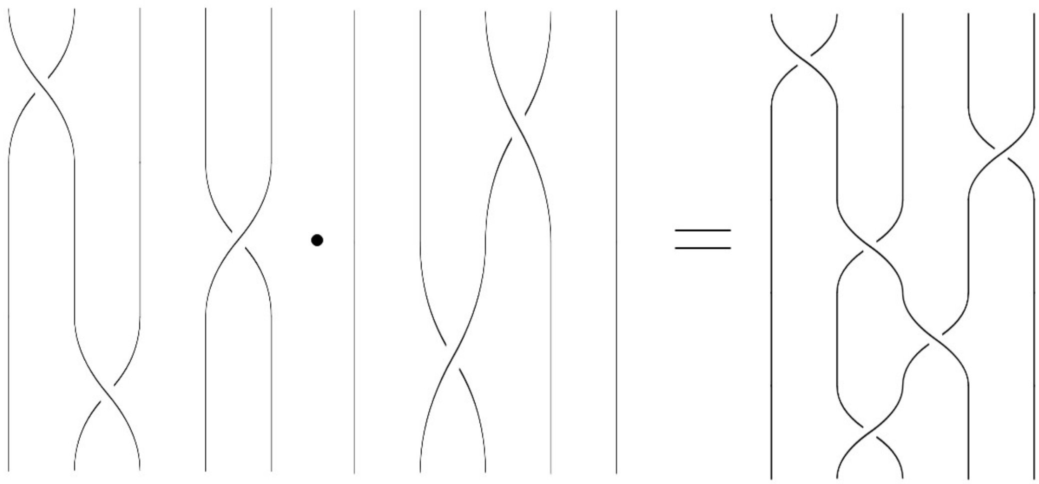

FIGURE 3.8. Example of the product of two braids. The binary operation is represented as multiplication "$\cdot$".

The braid group proposed by Artin, denoted B_n, gives an algebraic way to present braids using generators, relations and a binary operation.

Definition 3.1. The *Artin braid group* B_n is the group generated by $n-1$ generators s_1, $s_2 \dots, s_{n-1}$ and the *braid relations*

1) $s_i s_j = s_j s_i$ for all $i, j = 1, 2, \dots, n-1$ with $|i-j| \geq 2$, and

2) $s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1}$ for $i = 1, 2, \dots, n-2$.

As shown in Figure 3.9, s_i corresponds to the element of twisting the i^{th} strand over the $(i+1)^{st}$ strand and s_i^{-1} corresponds to the element of twisting the $(i+1)^{st}$ strand over the i^{th} strand. Viewed in this way, the braid relation (1) is shown geometrically as in Figure 3.10 and (2) as in Figure 3.11.

More algebraic properties of the braid group are discussed in [Kra05].

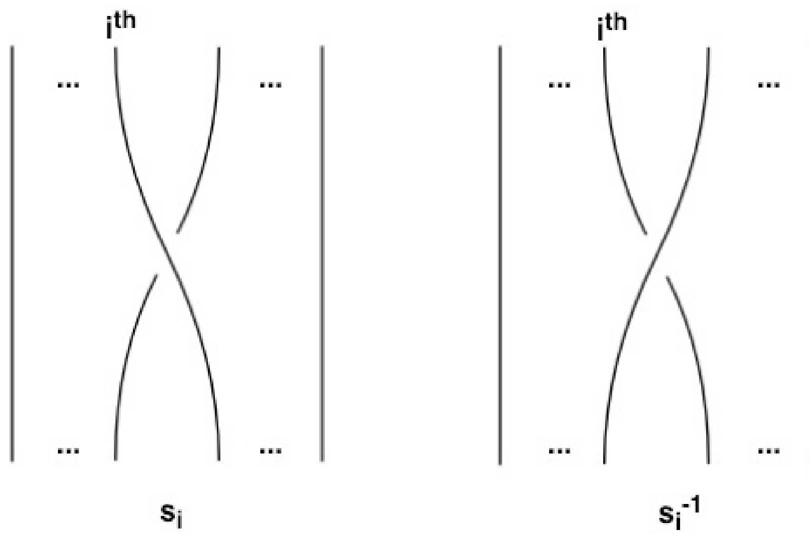

FIGURE 3.9. Crossings

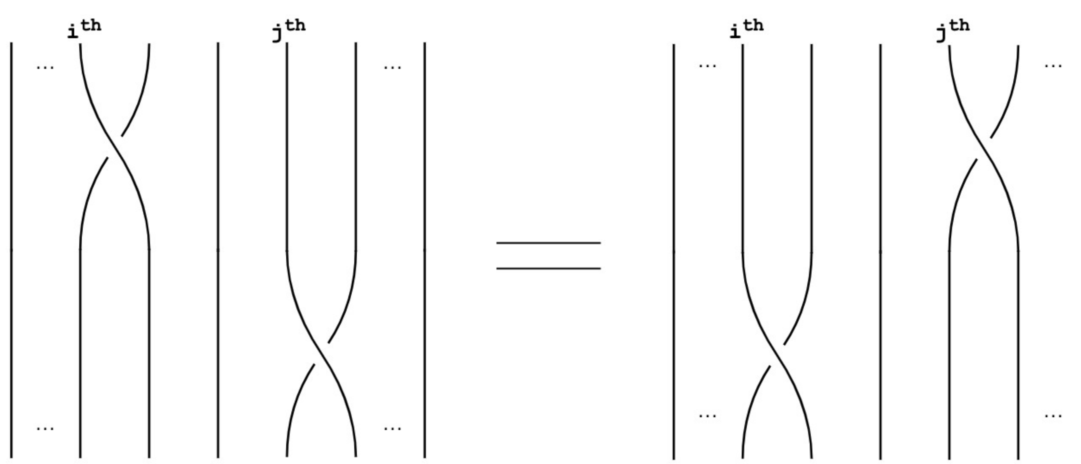

FIGURE 3.10. $s_i s_j = s_j s_i$

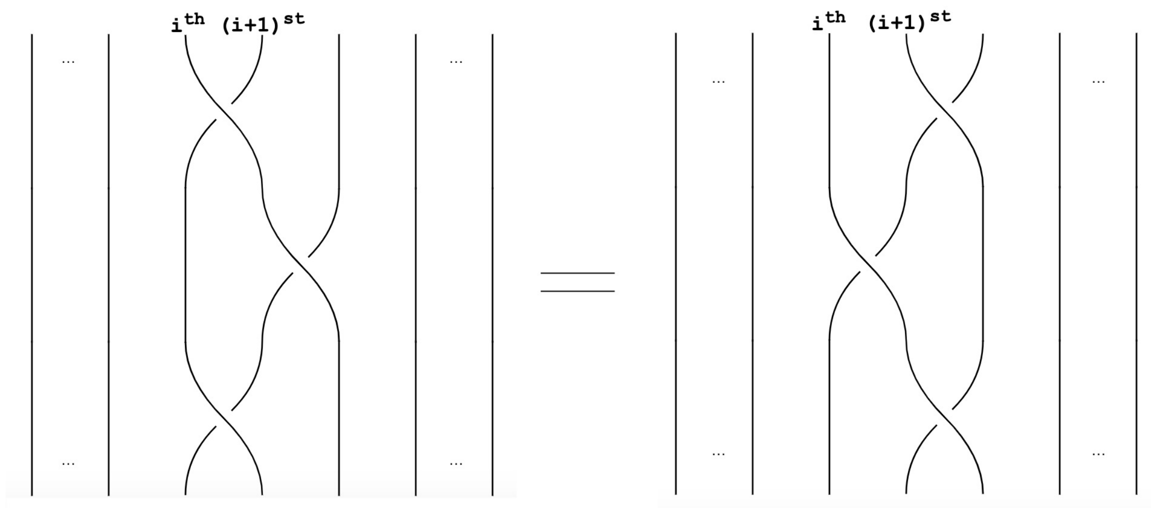

FIGURE 3.11. $s_is_{i+1}s_i = s_{i+1}s_is_{i+1}$

Theorem 3.2 states that the geometric braid group and the algebraic braid group are the same.

Theorem 3.2. *B_n and $\mathcal{B}_n$ are isomorphic.*

A rigorous and systematic formulation of the geometric and the algebraic braid group can be found in [KT08]. From now on, we can describe any braids interchangeably in the geometric way and algebraic way. We will call them both B_n.

The Artin braid group serves as the theoretical foundation of this thesis. The definitions of braid groups above allow us to describe a braid by listing the crossing-overs of the strings as we trace down the braid. We can arrange the braid in a nonunique way so that no two crossings occur at exactly the same height. This way of listing the crossings from top to bottom is called the *word* of a braid. For example, the word of the braid in Figure 2.5 is $s_1s_2^{-1}s_3$, and the word of the braid in Figure 2.6 is $s_2s_1s_3^{-1}s_2s_4$.

Theorem 3.3. *Let S_n be a symmetric group. Suppose that $\mathfrak{s}_1, \mathfrak{s}_2, ..., \mathfrak{s}_{n-1} \in S_n$ where $\mathfrak{s}_i$ stands for the transposition $(i, i{+}1)$. There is a unique group homomorphism $\pi : B_n \to S_n$ such that $\mathfrak{s}_i = \pi(s_i)$ for all $i = 1, 2, \ldots, n-1$, and this homomorphism is surjective.*

For example, the element in S_n that the braid in Figure 2.5 corresponds to is

$$\mathfrak{s}_1\mathfrak{s}_2\mathfrak{s}_3 = (1,2)(2,3)(3,4) = (3,2,1,4),$$

which can be understood as that the third strand at the top ends at the second position, the second at the top ends at the first, the first at the top ends at the fourth and the fourth at the top ends at the third.

The element that the braid in Figure 2.6 corresponds to is

$$\mathfrak{s}_2\mathfrak{s}_1\mathfrak{s}_3\mathfrak{s}_2\mathfrak{s}_4 = (2,3)(1,2)(3,4)(2,3)(4,5) = (1,3)(5,4,2),$$

which can be understood as that the first and the third strands exchange their positions when traveling to the bottom, the fifth strand at the top ends at the fourth position, the fourth at the top ends at the second and the second at the top ends at the fifth.

A classical question that usually arises in group theory is the *word problem*. In B_n, the word problem is about whether or not two given braid words are equivalent. Artin was the first person who solved the word problem in 1947 in his paper [Art47]. He proposed that every braid can be put into a normal form called the *combed braid*, and if the combed braids from two braids are the same, then we can conclude that the two original braids are equivalent. The "combing" technique is to put the braid into the form where the first strand first moves with all the other strands staying straight, and then the second starts moving with the rest staying straight, and this pattern goes on until the very last strand. Figure 3.12 shows an example of a braid and its combed braid.

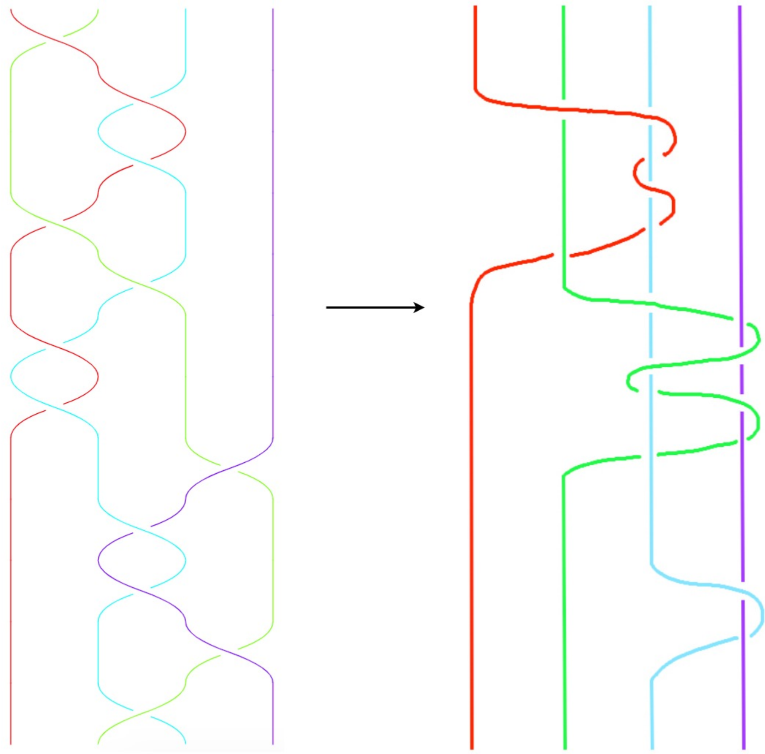

FIGURE 3.12. The braid $s_1 s_2 s_2 s_1 s_2 s_1 s_1 s_3^{-1} s_2 s_2 s_3 s_2^{-1}$ can be combed into the combed braid shown on the right.

More approaches of solving the word problem are illustrated in [Fed03].

4. Annular braid groups

Maypole dances can be viewed as annular braids. The set of maypole dances with n ribbons forms a group that we call the *maypole braid group* or the *annular braid group*. The following sections will investigate properties of this group.

The idea of annular braids is similar to ordinary braids except that we start from top and bottom circles with a pole down the center instead of top and bottom bars. By doing this, we rename generators s_i from B_n as σ_i for the new group, which corresponds to the i^{th} strand passing over the $(i+1)^{st}$ stand. We also need to allow two more generators for the resultant group. One is that the n^{th} string of a braid can go over or under the first string; these are σ_0 and σ_0^{-1}, respectively. The other is that we can apply a twist to all the strings. We call a twist in which the strand in the i^{th} position moves to the $(i-1)^{st}$ position for all $i = 2, 3, ..., n$ and the one in the first position moves to the n^{th} position generator τ (see Figure 4.13). As a convention in this thesis, we label the strings in an increasing order counterclockwise looking down from above.

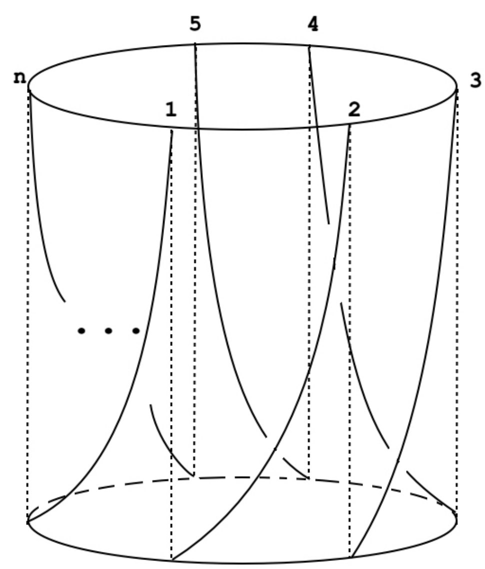

Figure 4.13. Diagram of τ

In the next section, we will present three ways of describing the annular braid group. For simplicity of illustration, we will present an annular braid with a diagram in such a way where we cut along the line of $(n, 0, 0)$ and $(n, 0, 1)$ and expand onto a plane. Figure 4.14 shows the expanded diagram of Figure 4.13.

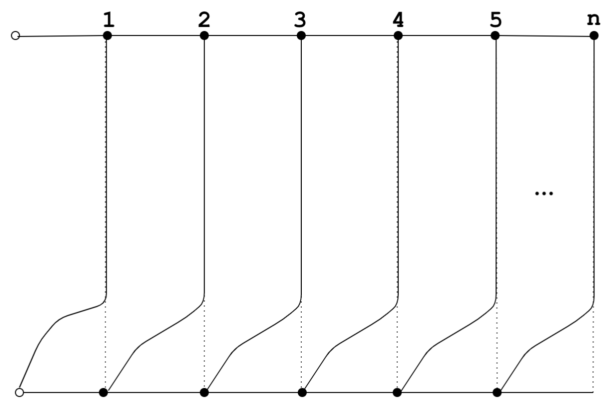

FIGURE 4.14. Diagram of τ

Sometimes it is more straightforward to work on the planar diagram and then wrap it back to the annular braid diagram. As a convention, we will always wrap in a way that the planar surface is facing outward in the annular braid, so the n^{th} string needs to go all the way around from the back to reach and be pasted onto the position next to the 1^{st} string on the left, as shown by hollow circles in Figure 4.14.

5. Presentations of annular braid group

There are multiple ways to present the annular braid group. This section will give three presentations that are equivalent.

5.1. **Presentation of AB_n.**

As explained in the previous section, the annular braid group of n strings can be described by renaming the s_i from the Artin braid group B_n as σ_i and introducing two more generators, σ_0 which corresponds to the new property that the first string and the last string are neighbors, and τ which corresponds to the action of twisting each string without crossings. Besides these changes, the only difference between B_n and AB_n is that the distance between generators is calculated modulo n. The presentation is

$$\mathcal{P} = \langle \tau, \sigma_0, \sigma_1, \sigma_2, \ldots, \sigma_{n-1} | \sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1} \text{ for } i = 0, 1, \ldots, n-1,$$

$$\sigma_i\sigma_j = \sigma_j\sigma_i \text{ for } |i-j| \geq 2, \tau\sigma_i\tau^{-1} = \sigma_{i+1} \text{ for } i = 0, 1, \ldots, n-1\rangle.$$

While the first two relations are not new to us, other than that the first and last strings are considered neighbors, the third relation shows that the rotation τ gives a new property to the group. Graphically, it looks as Figure 5.15.

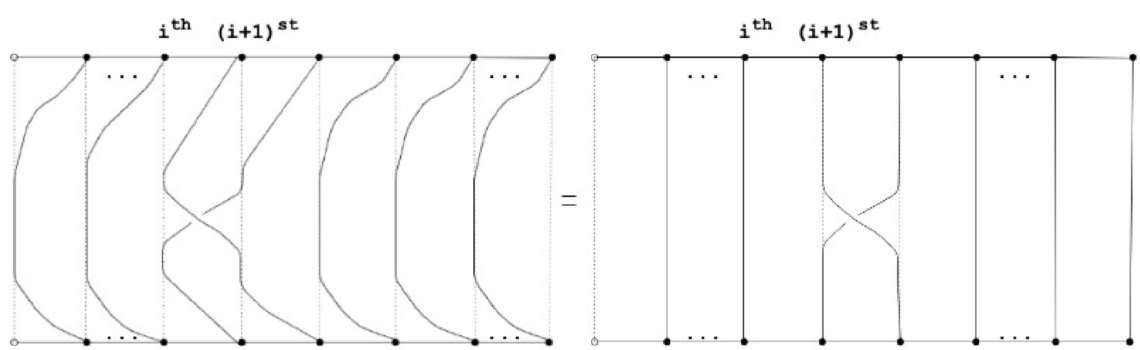

Figure 5.15. $\tau\sigma_i\tau^{-1} = \sigma_{i+1}$

The other two presentations are not as straightforward, but they have the advantage that they view the annular braids as special cases of ordinary braids. AB_n is isomorphic to a subgroup of B_{n+1}. We can transform an n-string annular braid to an $(n+1)$-string ordinary braid by treating the pole as the extra string.

5.2. **Presentation of D_{n+1}.**

Geometrically speaking, we define D_{n+1} as a subgroup of B_{n+1}, in which the first string always ends in the first spot on the bottom bar. In D_{n+1}, the generators $\gamma_2, \gamma_3, \ldots, \gamma_n$ are the $s_2, s_3, \ldots, s_n$ in B_{n+1}, so the distance between these generators is calculated modulo $n+1$.

Since D_{n+1} has the fixed end position of the first string, a new corresponding set of generators, $a_2, a_3, \ldots, a_{n+1}$, needs to be introduced. The generator a_i describes that the first sting goes under the second, the third until the $(i-1)^{st}$ string, and then goes over the i^{th} string, turns around, crosses under the i^{th} string, and then goes back to the first position on the bottom while crossing under all the strings on the way (see Figure 5.16). The generator a_i in D_{n+1} corresponds to $\sigma_1^{-1}\sigma_2^{-1}\ldots\sigma_{i-2}^{-1}\sigma_{i-1}\sigma_{i-1}\sigma_{i-2}\ldots\sigma_2\sigma_1$ in AB_{n+1}.

The presentation of D_{n+1} is

$$\begin{aligned}\mathcal{D} =&\langle \gamma_2, \gamma_3, \ldots, \gamma_n, a_2, a_3, \ldots, a_{n+1} | \gamma_i\gamma_{i+1}\gamma_i = \gamma_{i+1}\gamma_i\gamma_{i+1} \text{ for } i = 2, 3, \ldots, n-1,\\ &\gamma_i\gamma_j = \gamma_j\gamma_i \text{ for } |i-j| \geq 2, \gamma_i a_k \gamma_i^{-1} = a_k \text{ for } k \neq i, i+1,\\ &\gamma_i a_i \gamma_i^{-1} = a_{i+1}, \gamma_i a_{i+1}\gamma_i^{-1} = a_{i+1}^{-1} a_i a_{i+1} \text{ for } i = 2, 3, \ldots, n\rangle.\end{aligned}$$

The last three relations are new to us. To help understand the relations, some examples are shown in Figure 5.17, Figure 5.18 and Figure 5.19.

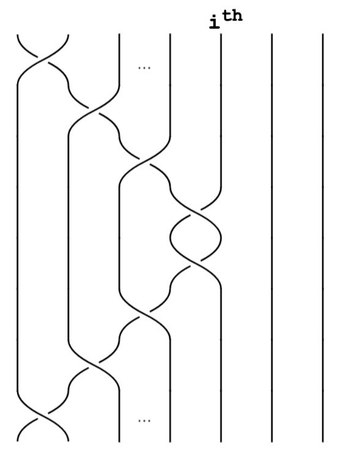

FIGURE 5.16. Diagram of generator a_i

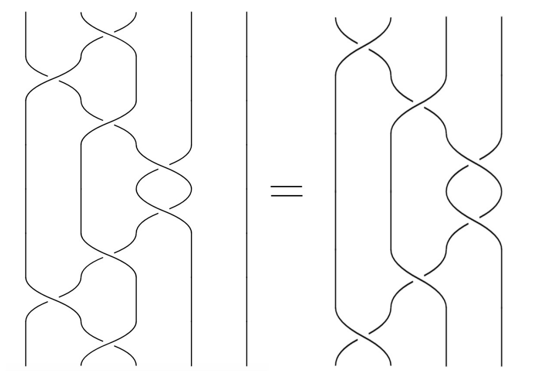

FIGURE 5.17. $\gamma_2 a_4 \gamma_2^{-1} = a_4$

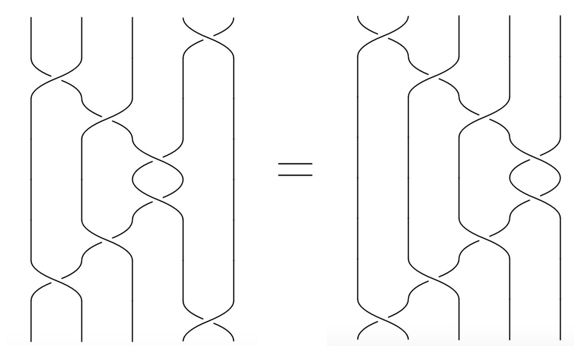

FIGURE 5.18. $\gamma_4 a_4 \gamma_4^{-1} = a_5$

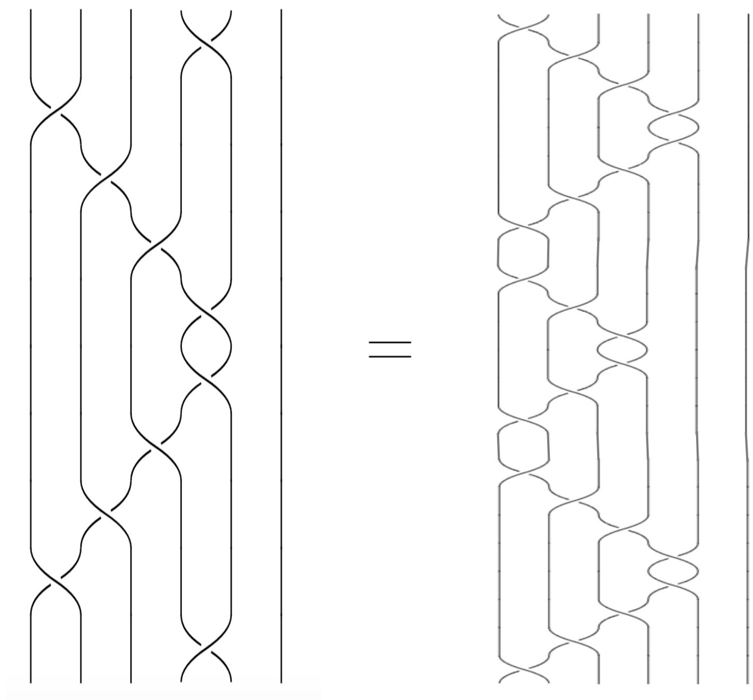

FIGURE 5.19. $\gamma_4 a_5 \gamma_4^{-1} = a_5^{-1} a_4 a_5$

Kent and Peifer ([KP02]) shows that the presentations $\mathcal{P}$ and $\mathcal{D}$ are equivalent. Therefore, $AB_n \cong D_{n+1}$.

In AB_n, we have σ_0, which corresponds to the n^{th} string crossing over the first string. In D_{n+1}, the effect of σ_0 can be achieved by applying $a_2^{-1}\gamma_2^{-1}\gamma_3^{-1}\dots\gamma_{n-1}^{-1}\gamma_n^{-1}\gamma_n\gamma_n\gamma_{n-1}\dots\gamma_3\gamma_2 a_2$ (see Figure 5.20). Furthermore, the effect of τ in AB_n is shifting each string to the neighboring position with a smaller index. In D_{n+1}, this can be achieved by applying $a_2^{-1}\gamma_2^{-1}\gamma_3^{-1}\dots\gamma_{n-1}^{-1}\gamma_n^{-1}$ (see Figure 5.21).

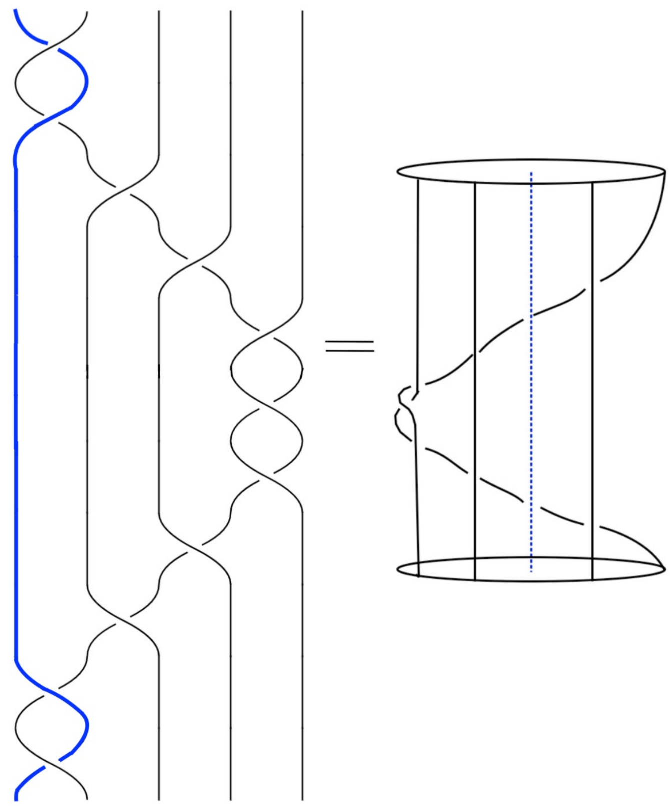

FIGURE 5.20. Equivalent effect of σ_0 in D_{n+1}. The strand representing the pole is in blue.

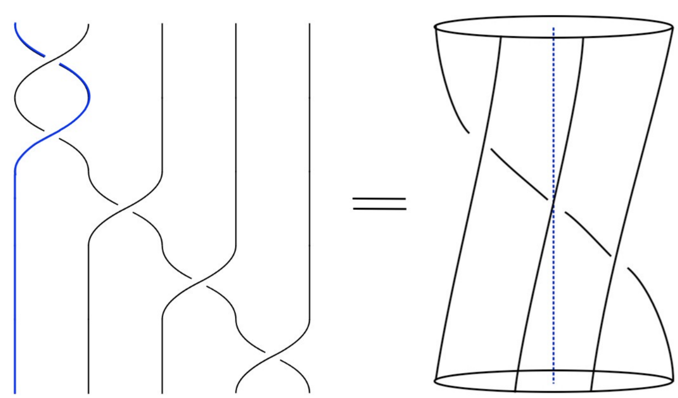

FIGURE 5.21. Equivalent effect of τ in D_{n+1}. The strand representing the pole is in blue.

5.3. **Presentation of CB_{n+1}.**

The third way also uses $n+1$ strings to represent the annular braid with n strings. Jean-Luc Thiffeault and Marko Budišić ([TB15]) showed a way to define the generators so that we can view the $(n+1)^{st}$ string as the fixed pole in the annular braid. However, they omitted the generator of rotation in the annular braid group. They also did not prove that the new group would be indeed isomorphic to the group AB_n. We will provide a definition of the rotation generator and the presentation of the group. Also, we will give an algebraic proof of the isomorphism between this new group and AB_n. To avoid confusion and be consistent with D_{n+1}, we use the same idea but rearrange the generators so that the first string is the fixed pole. We call this group CB_{n+1}. There are n generators. We let $\sigma_2, \sigma_3, \ldots, \sigma_n$ in AB_{n+1} correspond to the new generators $\Sigma_2, \Sigma_3, \ldots, \Sigma_n$. In other words, $\Sigma_i = \sigma_i$ for $i = 2, 3, \ldots, n$. We also create two new generator Σ_0 and $\mathcal{T}$ using the generators from AB_{n+1}, which are

$$\Sigma_0 = \sigma_n \sigma_{n-1} \ldots \sigma_2 \sigma_1^2 \sigma_2 \sigma_1^{-2} \sigma_2^{-1} \ldots \sigma_{n-1}^{-1} \sigma_n^{-1}$$

$$\mathcal{T} = \sigma_1^{-2} \sigma_2^{-1} \sigma_3^{-1} \ldots \sigma_{n-1}^{-1} \sigma_n^{-1}$$

We will see that Σ_0 and $\mathcal{T}$ play the same roles as σ_0 and τ respectively in AB_n (see Figure 5.22 and Figure 5.23).

In the following algebraic proof, we will use Tietze transformations to show that $CB_{n+1} \cong AB_n$. The theorem shows the procedures to prove that two groups given by generators and relations are isomorphic. In particular, each transformation yields a presentation for a group isomorphic to the previous group.

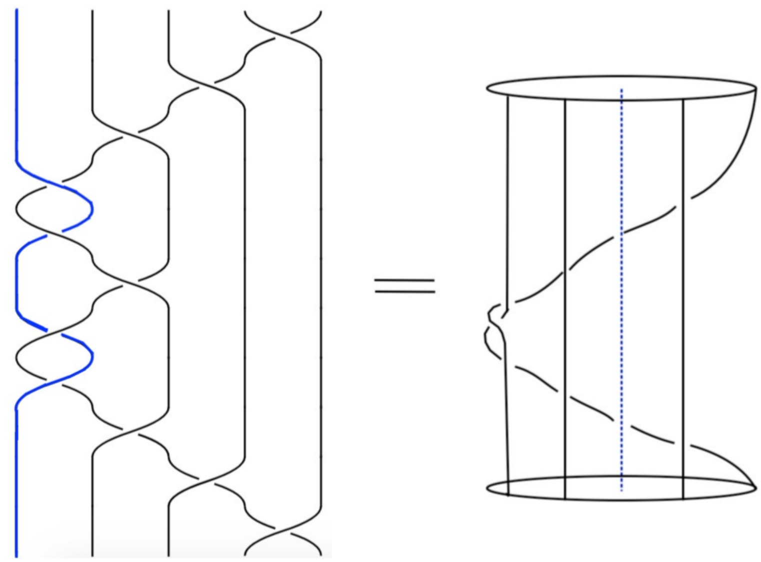

FIGURE 5.22. Σ_0 in CB_{n+1} is σ_0 in AB_n. The strand representing the pole is in blue.

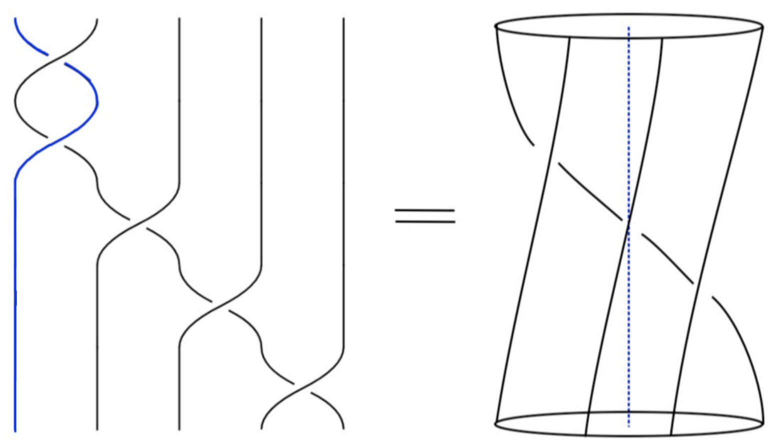

FIGURE 5.23. $\mathcal{T}$ in CB_{n+1} is τ in AB_n. The strand representing the pole is in blue.

Tietze transformations are stated as follows.

Consider the group $G = \langle \chi | \mathcal{R} \rangle$.

(T1) If ω is a relation on χ and ω can be derived from the set $\mathcal{R}$, then add ω to $\mathcal{R}$.

(T2) If ω is a relation in $\mathcal{R}$ that can be derived from the other relations, then remove ω from $\mathcal{R}$.

(T3) If ν is a word on χ and t is a symbol not in χ, then add t to the generating set and $t = \nu$ to $\mathcal{R}$.

(T4) If one relation takes the form $t = \nu$, where $t \in \chi$ and ν is a word on $\chi - \{t\}$, then remove t from the set χ, remove $t = \nu$ from $\mathcal{R}$, and substitute ν for all occurrences of t in the remaining relations.

Generally speaking, (T1) allows us to insert a relation, (T2) allows us to delete a relation, (T3) allows us to insert a generator, and (T4) allows us to delete a generator in a presentation.

Algebraic proof.

Kent and Peifer showed that $AB_n \cong D_{n+1}$, and by construction, D_{n+1} is a subgroup of B_{n+1}. Thus, AB_n is isomorphic to a subgroup of B_{n+1}. Likewise, CB_{n+1} is a subgroup of B_{n+1}. We will prove that $CB_{n+1} \cong D_{n+1} \cong AB_{n+1}$.

CB_{n+1} is the subgroup of B_{n+1} with generators:

$$s_2, s_3, s_4, \ldots, s_n,$$

$$s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1},$$

$$s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1},$$

and the only relations are the relations from B_{n+1}.

(I) With the relations from B_{n+1}, we have the original presentation of this subgroup:

$$\begin{aligned}CB_{n+1} = \mathcal{C}_1 =&\langle s_2, s_3, s_4, \dots, s_n, s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1},\\ &s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1} | s_i s_j = s_j s_i \text{ for all } i, j = 1, 2, \dots, n-1, n\\ &\text{with } |i-j| \geq 2, \text{ and } s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1} \text{ for } i = 1, 2, \dots, n-1\rangle.\end{aligned}$$

Note that from the relation $s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1}$, we can further derive

$$s_i^{-1} s_{i+1} s_i = s_{i+1} s_i s_{i+1}^{-1};$$

$$s_i^{-1} s_{i+1}^{-1} s_i = s_{i+1} s_i^{-1} s_{i+1}^{-1};$$

$$\text{and } s_i^{-1} s_{i+1}^{-1} s_i^{-1} = s_{i+1}^{-1} s_i^{-1} s_{i+1}^{-1}$$

We use these relations in the following derivations.

(II) We can derive some new relations from the original ones presented in $\mathcal{C}_1$.

(i) Consider $j = 3, 4, \dots, n-1$,

$$\begin{aligned}&(s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_j\\ =&s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{j-1}^{-1} s_j^{-1} s_{j+1}^{-1} s_j s_{j+2}^{-1} \dots s_{n-1}^{-1} s_n^{-1}\\ =&s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{j-1}^{-1} s_{j+1} s_j^{-1} s_{j+1}^{-1} s_{j+2}^{-1} \dots s_{n-1}^{-1} s_n^{-1}\\ =&s_n s_{n-1} \dots s_{j+2} s_{j+1} s_j s_{j+1} s_{j-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{j-1}^{-1} s_j^{-1} s_{j+1}^{-1} s_{j+2}^{-1} \dots s_{n-1}^{-1} s_n^{-1}\\ =&s_n s_{n-1} \dots s_{j+2} s_j s_{j+1} s_j s_{j-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}\\ =&s_j (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1})\end{aligned}$$

Thus, by (T1), we can update the presentation as the following. Note that from now on, the updated relations that are different from the ones in the last presentation will be marked in blue.

$$\mathcal{C}_2 = \langle s_2, s_3, s_4, \dots, s_n, s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1},$$

$$s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1} | s_i s_j = s_j s_i \text{ for all } i, j = 1, 2, \dots, n-1, n \text{ with } |i-j| \geq 2,$$

$$s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1} \text{ for } i = 1, 2, \dots, n-1 \text{ and}$$

$$(s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_j = s_j (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1})$$

$$\text{for } j = 3, 4, \dots, n-1 \rangle.$$

(ii) (a) For $i = 2, 3, \dots, n-1$,

$$\begin{aligned}
&(s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_i (s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})^{-1} \\
=& s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{i-1}^{-1} s_i^{-1} s_{i+1}^{-1} s_i s_{i+2}^{-1} \dots s_{n-1}^{-1} s_n^{-1} (s_n s_{n-1} \dots s_3 s_2 s_1^2) \\
=& s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{i-1}^{-1} s_{i+1} s_i^{-1} s_{i+1}^{-1} s_{i+2}^{-1} \dots s_{n-1}^{-1} s_n^{-1} s_n s_{n-1} \dots s_3 s_2 s_1^2 \\
=& s_{i+1} s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1} s_n s_{n-1} \dots s_3 s_2 s_1^2 \\
=& s_{i+1}
\end{aligned}$$

(b)

$$\begin{aligned}
&(s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) \\
=& (s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1} s_n s_{n-1} \dots s_2 s_1^2)(s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) \\
=& s_2 (s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1})
\end{aligned}$$

(c)

$$(s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})$$

$$=(s_n s_{n-1} \dots s_3 s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} s_1^{-2} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})$$

$$=(s_n s_{n-1} \dots s_3 s_2 s_1 (s_1 s_2 s_1^{-1})(s_1^{-1} s_2^{-1} s_1^{-1}) s_1^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})$$

$$=(s_n s_{n-1} \dots s_3 s_2 s_1 (s_2^{-1} s_1 s_2)(s_2^{-1} s_1^{-1} s_2^{-1}) s_1^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})$$

$$=(s_n s_{n-1} \dots s_3 (s_2 s_1 s_2^{-1}) s_2^{-1} s_1^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})$$

$$=(s_n s_{n-1} \dots s_3 s_1^{-1} s_2 (s_1 s_2^{-1} s_1^{-1}) s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})$$

$$=(s_n s_{n-1} \dots s_3 s_1^{-1} s_2 s_2^{-1} s_1^{-1} s_2 s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})$$

$$=(s_n s_{n-1} \dots s_3 s_1^{-2} s_2 s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})$$

$$=s_1^{-2}(s_n s_{n-1} \dots (s_3 s_2 s_3^{-1}) s_2^{-1} s_4^{-1} s_3^{-1} s_5^{-1} \dots s_{n-2}^{-1} s_{n-3}^{-1} s_{n-1}^{-1} s_{n-2}^{-1} s_n^{-1} s_{n-1}^{-1} s_n^{-1})$$

$$=s_1^{-2}(s_n s_{n-1} \dots s_4 (s_2^{-1} s_3 s_2) s_2^{-1} s_4^{-1} s_3^{-1} s_5^{-1} \dots s_{n-2}^{-1} s_{n-3}^{-1} s_{n-1}^{-1} s_{n-2}^{-1} s_n^{-1} s_{n-1}^{-1} s_n^{-1})$$

$$=s_1^{-2} s_2^{-1}(s_n s_{n-1} \dots s_5 (s_4 s_3 s_4^{-1}) s_3^{-1} s_5^{-1} \dots s_{n-2}^{-1} s_{n-3}^{-1} s_{n-1}^{-1} s_{n-2}^{-1} s_n^{-1} s_{n-1}^{-1} s_n^{-1})$$

$$=s_1^{-2} s_2^{-1}(s_n s_{n-1} \dots s_5 (s_3^{-1} s_4 s_3) s_3^{-1} s_5^{-1} \dots s_{n-2}^{-1} s_{n-3}^{-1} s_{n-1}^{-1} s_{n-2}^{-1} s_n^{-1} s_{n-1}^{-1} s_n^{-1})$$

$$=s_1^{-2} s_2^{-1} s_3^{-1}(s_n s_{n-1} \dots s_6 (s_5 s_4 s_5^{-1}) s_4^{-1} s_6^{-1} \dots s_{n-2}^{-1} s_{n-3}^{-1} s_{n-1}^{-1} s_{n-2}^{-1} s_n^{-1} s_{n-1}^{-1} s_n^{-1})$$

$$= \dots$$

$$=s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-2}^{-1}(s_n s_{n-1} s_n^{-1} s_{n-1}^{-1} s_n^{-1})$$

$$=s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-2}^{-1}(s_n s_n^{-1} s_{n-1}^{-1} s_n s_n^{-1})$$

$$=s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-2}^{-1} s_{n-1}^{-1}$$

$$=(s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-2}^{-1} s_{n-1}^{-1} s_n^{-1}) s_n$$

Thus, by (T1), we can update the presentation as

$$
\begin{aligned}
\mathcal{C}_3 =&\langle s_2, s_3, s_4, \ldots, s_n, s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}, s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1} \mid \\
&s_i s_j = s_j s_i \text{ for all } i, j = 1, 2, \ldots, n-1, n \text{ with } |i-j| \geq 2, \\
&s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1} \text{ for } i = 1, 2, \ldots, n-1, \\
&(s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_j = s_j (s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) \\
&\text{ for } j = 3, 4, \ldots, n-1, \\
&(s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_i (s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1})^{-1} = s_{i+1} \text{ for } i = 2, 3, \ldots, n-1, \\
&(s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1})(s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) = s_2 (s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}), \text{ and} \\
&(s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1})(s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) = (s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_n \rangle.
\end{aligned}
$$

(iii) (a)

$$
\begin{aligned}
&(s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_2 (s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) \\
=& s_n s_{n-1} \ldots s_4 s_3 s_2 s_1^2 s_2 s_1^{-2} (s_2^{-1} s_3^{-1} s_2) s_3 s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} s_3^{-1} s_4^{-1} \ldots s_{n-1}^{-1} s_n^{-1} \\
=& s_n s_{n-1} \ldots s_4 s_3 s_2 s_1^2 s_2 s_1^{-2} s_3 s_2^{-1} s_3^{-1} s_3 s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} s_3^{-1} s_4^{-1} \ldots s_{n-1}^{-1} s_n^{-1} \\
=& s_n s_{n-1} \ldots s_4 s_3 s_2 s_1^2 (s_2 s_3 s_2) s_1^{-2} s_2^{-1} s_3^{-1} s_4^{-1} \ldots s_{n-1}^{-1} s_n^{-1} \\
=& s_n s_{n-1} \ldots s_4 s_3 s_2 s_1^2 (s_3 s_2 s_3) s_1^{-2} s_2^{-1} s_3^{-1} s_4^{-1} \ldots s_{n-1}^{-1} s_n^{-1} \\
=& s_n s_{n-1} \ldots s_4 (s_3 s_2 s_3) s_1^2 s_2 s_1^{-2} (s_3 s_2^{-1} s_3^{-1}) s_4^{-1} \ldots s_{n-1}^{-1} s_n^{-1} \\
=& s_n s_{n-1} \ldots s_4 (s_2 s_3 s_2) s_1^2 s_2 s_1^{-2} (s_2^{-1} s_3^{-1} s_2) s_4^{-1} \ldots s_{n-1}^{-1} s_n^{-1} \\
=& s_2 (s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_2
\end{aligned}
$$

(b)

$$(s_ns_{n-1}\dots s_2s_1^2s_2s_1^{-2}s_2^{-1}\dots s_{n-1}^{-1}s_n^{-1})s_n(s_ns_{n-1}\dots s_2s_1^2s_2s_1^{-2}s_2^{-1}\dots s_{n-1}^{-1}s_n^{-1})$$

$$=(s_n\dots s_2s_1^2s_2s_1^{-2}s_2^{-1}s_3^{-1}\dots s_n^{-1})s_n(s_ns_{n-1}\dots s_4s_3(s_2s_1(s_1s_2s_1^{-1})s_1^{-1}s_2^{-1})s_3^{-1}s_4^{-1}\dots s_{n-1}^{-1}s_n^{-1})$$

$$=(s_n\dots s_2s_1^2s_2s_1^{-2}s_2^{-1}s_3^{-1}\dots s_n^{-1})s_n(s_ns_{n-1}\dots s_4s_3(s_2s_1s_2^{-1})(s_1s_2s_1^{-1})s_2^{-1}s_3^{-1}s_4^{-1}\dots s_{n-1}^{-1}s_n^{-1})$$

$$=(s_n\dots s_2s_1^2s_2s_1^{-2}s_2^{-1}s_3^{-1}\dots s_n^{-1})s_n(s_ns_{n-1}\dots s_4s_3s_1^{-1}(s_2s_1s_2^{-1})s_1s_2s_2^{-1}s_3^{-1}s_4^{-1}\dots s_{n-1}^{-1}s_n^{-1})$$

$$=(s_n\dots s_3s_2s_1^2s_2s_1^{-2}s_2^{-1}s_3^{-1}s_4^{-1}\dots s_n^{-1})s_n(s_ns_{n-1}\dots s_4s_3s_1^{-2}s_2s_1^2s_2s_2^{-1}s_3^{-1}s_4^{-1}\dots s_{n-1}^{-1}s_n^{-1})$$

$$=(s_n\dots s_3(s_2s_1^2s_2s_1^{-2}s_2^{-1}s_1^{-2})s_3^{-1}s_4^{-1}\dots s_n^{-1})s_n(s_ns_{n-1}\dots s_4(s_3s_2s_3^{-1})s_4^{-1}\dots s_{n-1}^{-1}s_n^{-1})s_1^2$$

$$=(s_n\dots s_3s_1^{-2}s_2s_3^{-1}s_4^{-1}\dots s_n^{-1})s_n(s_ns_{n-1}\dots s_4s_2^{-1}s_3s_2s_4^{-1}\dots s_{n-1}^{-1}s_n^{-1})s_1^2$$

$$=s_1^{-2}(s_n\dots s_3(s_2s_3^{-1}s_2^{-1})s_4^{-1}\dots s_n^{-1})s_n(s_ns_{n-1}\dots(s_4s_3s_4^{-1})\dots s_{n-1}^{-1}s_n^{-1})s_2s_1^2$$

$$=s_1^{-2}(s_n\dots s_3s_3^{-1}s_2^{-1}s_3s_4^{-1}\dots s_n^{-1})s_n(s_ns_{n-1}\dots s_3^{-1}s_4s_3\dots s_{n-1}^{-1}s_n^{-1})s_2s_1^2$$

$$=s_1^{-2}s_2^{-1}(s_n\dots s_4s_3s_4^{-1}\dots s_n^{-1})s_n(s_ns_{n-1}\dots s_3^{-1}s_4s_3\dots s_{n-1}^{-1}s_n^{-1})s_2s_1^2$$

$$=s_1^{-2}s_2^{-1}(s_n\dots s_5s_4(s_3s_4^{-1}s_3^{-1})s_5^{-1}\dots s_n^{-1})s_n(s_ns_{n-1}\dots(s_5s_4s_5^{-1})\dots s_{n-1}^{-1}s_n^{-1})s_3s_2s_1^2$$

$$=\dots$$

$$=s_1^{-2}s_2^{-1}s_3^{-1}\dots s_{n-3}^{-1}(s_ns_{n-1}s_{n-2}s_{n-1}^{-1}s_n^{-1})s_n(s_ns_{n-1}s_{n-2}s_{n-1}^{-1}s_n^{-1})s_{n-3}s_{n-4}\dots s_2s_1^2$$

$$=s_ns_{n-1}(s_1^{-2}s_2^{-1}s_3^{-1}\dots s_{n-3}^{-1})(s_{n-2}s_{n-1}^{-1}s_ns_{n-1}s_{n-2}s_{n-1}^{-1}s_n^{-1})s_{n-3}s_{n-4}\dots s_2s_1^2$$

Note that

$$\begin{aligned}
s_{n-2}(s_{n-1}^{-1}s_ns_{n-1})s_{n-2}s_{n-1}^{-1}s_n^{-1} &= s_{n-2}(s_ns_{n-1}s_n^{-1})s_{n-2}s_{n-1}^{-1}s_n^{-1}\\
&= s_n(s_{n-2}s_{n-1}s_{n-2})(s_n^{-1}s_{n-1}^{-1}s_n^{-1})\\
&= s_n(s_{n-1}s_{n-2}s_{n-1})(s_{n-1}^{-1}s_n^{-1}s_{n-1}^{-1})\\
&= s_ns_{n-1}s_{n-2}s_n^{-1}s_{n-1}^{-1}
\end{aligned}$$

Therefore,

$$s_n s_{n-1}(s_1^{-2}s_2^{-1}s_3^{-1}\dots s_{n-3}^{-1})(s_{n-2}s_{n-1}^{-1}s_n s_{n-1}s_{n-2}s_{n-1}^{-1}s_n^{-1})s_{n-3}s_{n-4}\dots s_2 s_1^2$$

$$=s_n s_{n-1}(s_1^{-2}s_2^{-1}s_3^{-1}\dots s_{n-3}^{-1})(s_n s_{n-1}s_{n-2}s_n^{-1}s_{n-1}^{-1})s_{n-3}s_{n-4}\dots s_2 s_1^2$$

$$=s_n(s_{n-1}s_n s_{n-1})s_n^{-1}(s_1^{-2}s_2^{-1}s_3^{-1}\dots s_{n-4}^{-1}(s_{n-3}^{-1}s_{n-2}s_{n-3})s_{n-4}\dots s_2 s_1^2)s_{n-1}^{-1}$$

$$=s_n(s_n s_{n-1}s_n)s_n^{-1}(s_1^{-2}s_2^{-1}s_3^{-1}\dots s_{n-4}^{-1}(s_{n-2}s_{n-3}s_{n-2}^{-1})s_{n-4}\dots s_2 s_1^2)s_{n-1}^{-1}$$

$$=s_n s_n s_{n-1}s_{n-2}(s_1^{-2}s_2^{-1}s_3^{-1}\dots(s_{n-4}^{-1}s_{n-3}s_{n-4})\dots s_2 s_1^2)s_{n-2}^{-1}s_{n-1}^{-1}$$

$$=\dots$$

$$=s_n(s_n s_{n-1}s_{n-2}\dots s_4(s_1^{-2}s_2^{-1}s_3 s_2 s_1^2)s_4^{-1}s_5^{-1}\dots s_{n-2}^{-1}s_{n-1}^{-1})(s_n^{-1}s_n)$$

Note that

$$\begin{aligned}
s_1^{-2}s_2^{-1}s_3 s_2 s_1^2 &= s_1^{-2}s_3 s_2 s_3^{-1}s_1^2\\
&= s_3 s_1^{-1}(s_1^{-1}s_2 s_1)s_1 s_3^{-1}\\
&= s_3(s_1^{-1}s_2 s_1)s_2^{-1}s_1 s_3^{-1}\\
&= s_3(s_2 s_1 s_2^{-1})s_2^{-1}s_1 s_3^{-1}\\
&= s_3 s_2 s_1 s_2^{-1}(s_1 s_1^{-1})s_2^{-1}s_1 s_3^{-1}\\
&= s_3 s_2 s_1 s_2^{-1}s_1(s_1^{-1}s_2^{-1}s_1)s_3^{-1}\\
&= s_3 s_2 s_1(s_2^{-1}s_1 s_2)s_1^{-1}s_2^{-1}s_3^{-1}\\
&= s_3 s_2 s_1(s_1 s_2 s_1^{-1})s_1^{-1}s_2^{-1}s_3^{-1}\\
&= s_3 s_2 s_1^2 s_2 s_1^{-2}s_2^{-1}s_3^{-1}
\end{aligned}$$

It follows that

$$
\begin{aligned}
& s_n(s_n s_{n-1} s_{n-2} \dots s_4 (s_1^{-2} s_2^{-1} s_3 s_2 s_1^2) s_4^{-1} s_5^{-1} \dots s_{n-2}^{-1} s_{n-1}^{-1})(s_n^{-1} s_n) \\
=& s_n(s_n s_{n-1} s_{n-2} \dots s_4 (s_3 s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} s_3^{-1}) s_4^{-1} s_5^{-1} \dots s_{n-2}^{-1} s_{n-1}^{-1})(s_n^{-1} s_n) \\
=& s_n(s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_n
\end{aligned}
$$

Thus, by (T1), we can update the presentation as

$$
\begin{aligned}
\mathcal{C}_4 =& \langle s_2, s_3, s_4, \dots, s_n, s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}, s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1} \mid \\
& s_i s_j = s_j s_i \text{ for all } i, j = 1, 2, \dots, n-1, n \text{ with } |i-j| \geq 2, \\
& s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1} \text{ for } i = 1, 2, \dots, n-1, \\
& (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_j = s_j (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) \\
& \text{ for } j = 3, 4, \dots, n-1, \\
& (s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_i (s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})^{-1} = s_{i+1} \text{ for } i = 2, 3, \dots, n-1, \\
& (s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) = s_2 (s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}), \\
& (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1})(s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1}) = (s_1^{-2} s_2^{-1} s_3^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_n, \\
& (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_2 (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) \\
& = s_2 (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_2, \\
& (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_n (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) \\
& = s_n (s_n s_{n-1} \dots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \dots s_{n-1}^{-1} s_n^{-1}) s_n \rangle.
\end{aligned}
$$

(III) Notice that the generators contain only even powers of s_1, so we can delete the relations with odd power of s_1, as they will never arise. Thus, the presentation becomes

$$\mathcal{C}_5 = \langle s_2, s_3, s_4, \ldots, s_n, s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}, s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1} |$$

$s_i s_j = s_j s_i$ for all $i, j = 2, 3, \ldots, n-1, n$ with $|i-j| \geq 2$,

$s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1}$ for $i = 2, 3, \ldots, n-1,$

$(s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_j = s_j (s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1})$

for $j = 3, 4, \ldots, n-1$,

$(s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_i (s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1})^{-1} = s_{i+1}$ for $i = 2, 3, \ldots, n-1$,

$(s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1})(s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) = s_2 (s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1})$,

$(s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1})(s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) = (s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_n$,

$(s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_2 (s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1})$

$= s_2 (s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_2$,

$(s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_n (s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1})$

$= s_n (s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}) s_n \rangle$.

(IV) Now, we rename s_i as Σ_i, $s_n s_{n-1} \ldots s_2 s_1^2 s_2 s_1^{-2} s_2^{-1} \ldots s_{n-1}^{-1} s_n^{-1}$ as Σ_0 and $s_1^{-2} s_2^{-1} s_3^{-1} \ldots s_{n-1}^{-1} s_n^{-1}$ as $\mathcal{T}$. Then, the presentation becomes

$$\mathcal{C}_6 = \langle \Sigma_2, \Sigma_3, \ldots, \Sigma_{n-1}, \Sigma_n, \Sigma_0, \mathcal{T} | \Sigma_i \Sigma_j = \Sigma_j \Sigma_i \text{ for all } i, j = 2, 3, \ldots, n \text{ with } |i-j| \geq 2,$$

$\Sigma_i \Sigma_{i+1} \Sigma_i = \Sigma_{i+1} \Sigma_i \Sigma_{i+1}$ for $i = 2, 3, \ldots, n-1, \Sigma_0 \Sigma_j = \Sigma_j \Sigma_0$ for $j = 3, 4, \ldots, n-1$,

$\mathcal{T} \Sigma_i \mathcal{T}^{-1} = \Sigma_{i+1}$ for $i = 2, 3, \ldots, n-1, \mathcal{T} \Sigma_0 = \Sigma_2 \mathcal{T}, \Sigma_0 \mathcal{T} = \mathcal{T} \Sigma_n$,

$\Sigma_0 \Sigma_2 \Sigma_0 = \Sigma_2 \Sigma_0 \Sigma_2, \Sigma_0 \Sigma_n \Sigma_0 = \Sigma_n \Sigma_0 \Sigma_n \rangle$.

This is the same as the following presentation of group with subscripts viewed modulo $n+1$.

$$\begin{aligned}\mathcal{C}_6 =& \langle \Sigma_0\Sigma_2, \Sigma_3, \ldots, \Sigma_n, \mathcal{T} | \Sigma_i\Sigma_j = \Sigma_j\Sigma_i \text{ for all } i,j = 0,2,3,\ldots,n \text{ with } |i-j| \geq 2, \\ & \text{except that } \Sigma_2\Sigma_0 \neq \Sigma_0\Sigma_2, \Sigma_i\Sigma_{i+1}\Sigma_i = \Sigma_{i+1}\Sigma_i\Sigma_{i+1} \text{ for } i = 2,3,\ldots,n \\ & \text{and } \Sigma_0\Sigma_2\Sigma_0 = \Sigma_2\Sigma_0\Sigma_2, \mathcal{T}\Sigma_i\mathcal{T}^{-1} = \Sigma_{i+1} \text{ for } i = 2,3,\ldots,n \text{ and } \mathcal{T}\Sigma_0\mathcal{T}^{-1} = \Sigma_2\rangle.\end{aligned}$$

Thus, we can get the presentation $\mathcal{C}_6$ from the presentation of AB_n by renaming σ_0 as Σ_0, σ_i as Σ_{i+1} for $i = 1,2,\ldots,n-1$, and τ as $\mathcal{T}$. So, $\mathcal{C}_6 \cong AB_n$. Hence, we have

$$CB_{n+1} = \mathcal{C}_1 \cong \mathcal{C}_2 \cong \mathcal{C}_3 \cong \mathcal{C}_4 \cong \mathcal{C}_5 \cong \mathcal{C}_6 \cong AB_n$$

We now have the presentation of CB_{n+1} as

$$\begin{aligned}\mathcal{C} =& \langle \Sigma_0\Sigma_2, \Sigma_3, \ldots, \Sigma_n, \mathcal{T} | \Sigma_i\Sigma_j = \Sigma_j\Sigma_i \text{ for all } i,j = 0,2,3,\ldots,n \text{ with } |i-j| \geq 2, \\ & \text{except that } \Sigma_2\Sigma_0 \neq \Sigma_0\Sigma_2, \Sigma_i\Sigma_{i+1}\Sigma_i = \Sigma_{i+1}\Sigma_i\Sigma_{i+1} \text{ for } i = 2,3,\ldots,n \\ & \text{and } \Sigma_0\Sigma_2\Sigma_0 = \Sigma_2\Sigma_0\Sigma_2, \mathcal{T}\Sigma_i\mathcal{T}^{-1} = \Sigma_{i+1} \text{ for } i = 2,3,\ldots,n \text{ and } \mathcal{T}\Sigma_0\mathcal{T}^{-1} = \Sigma_2\rangle.\end{aligned}$$

The distance between generators in the group CB_{n+1} is computed modulo $n+1$.

6. Analyzing specific maypole dances - Grand Chain and Gypsies' Tent

Traditional maypole dances tend to either have the entire group of dancers moving at once in unison, or make the dancers divided into two teams and let each move independently. In the former case, the dancers just move in a circle without overtaking each other. Since all the dancers are facing and moving in the same direction, the only effect we can get from this group is either a rotation clockwise or counterclockwise. This is the same as the generator τ. While τ corresponds to moving clockwise, τ^{-1} corresponds to the opposite. The latter case can be more complicated. Usually the two teams are lined up in a circle in an alternating fashion so that no two members from the same team are neighbors. It implies that such a group has a constraint that the number of people is always a positive even number.

We can understand a dance at three different levels - the sequence of instructions to each dancer, the sequence of overhead moves and the braid diagram. Figure 6.24 shows an example using the three levels to describe the simple maypole dance where everyone rotates clockwise.

6.1. **Maypole dances with two teams.**

Interesting phenomena occur when we have more than one big group. We start with the case of two teams. The basic setting is that the dancers are divided into 2 teams, team A and team B, alternating around the pole.

What are the instructions we can tell to the dancers? The crossings of the strings result from having the neighboring dancers interchanging their positions. If we tell the A dancers to interchange with their neighboring B dancers while going clockwise on the outside, it is the same as telling B dancers to go counterclockwise and under their A

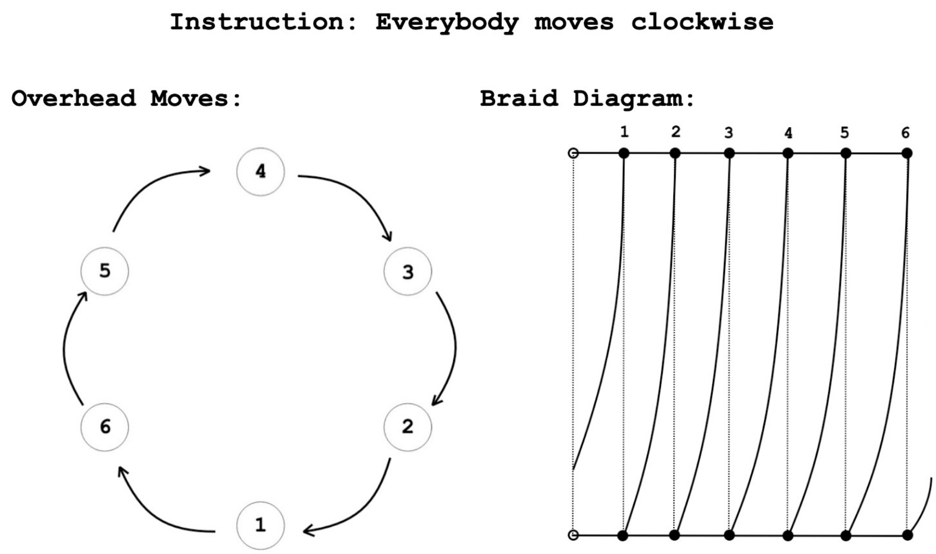

FIGURE 6.24. The three levels of one group of maypole dancers rotating clockwise.

neighbors. It turns out that we can tell A members to go clockwise on the outside or inside, or tell B members to go clockwise on the outside or inside, or tell everyone to rotate clockwise or counterclockwise. The possible instructions are shown in Figure 6.25.

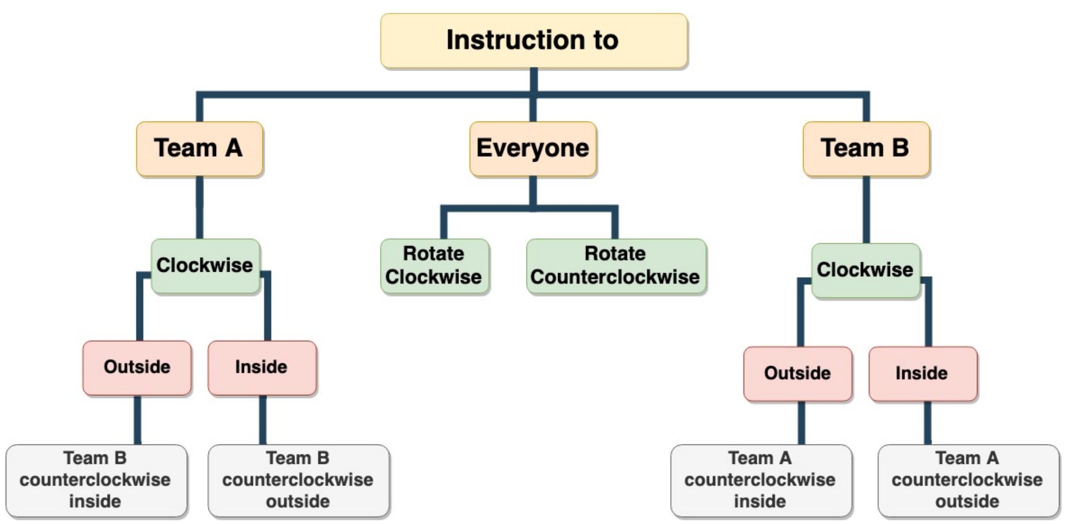

FIGURE 6.25. Instructions

Each of these instructions has a corresponding overhead diagram. For example, Figure 6.26 shows the overhead moves of the instruction that tells team A to move clockwise on the outside.

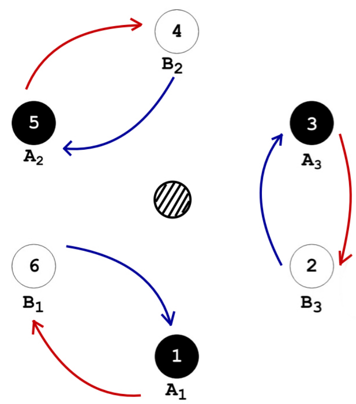

FIGURE 6.26. Team A goes clockwise on the outside while team B goes counterclockwise on the inside

Each instruction not only has a corresponding overhead diagram as the example shows in Figure 6.26, but also has a corresponding braid diagram if we now look horizontally and focus on how the ribbons cross over each other. Figure 6.27 shows the braid diagram corresponding to Figure 6.26.

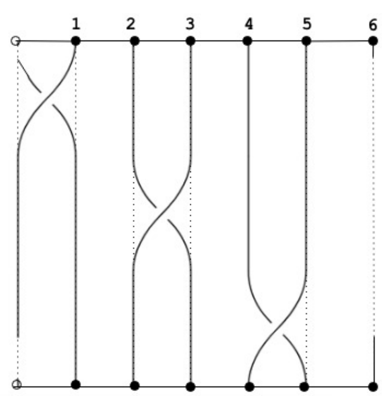

FIGURE 6.27. Braid diagram of A going clockwise on the outside and B going counterclockwise on the inside

We can further describe the braid diagram using words. The word of the braid shown in Figure 6.27 is $\sigma_0^{-1}\sigma_2^{-1}\sigma_4^{-1}$. In general, for a maypole dance group of $2n$ people where $n \in \mathbb{N}^+$, we have the following words representing the basic dance moves:

$$\begin{aligned}
\text{People at odd-number positions go clockwise from inside:} &\quad \sigma_0\sigma_2\sigma_4\ldots\sigma_{2n-2} \\
\text{People at odd-number positions go clockwise from outside:} &\quad \sigma_0^{-1}\sigma_2^{-1}\sigma_4^{-1}\ldots\sigma_{2n-2}^{-1} \\
\text{People at even-number positions go clockwise from inside:} &\quad \sigma_1\sigma_3\sigma_5\ldots\sigma_{2n-1} \\
\text{People at even-number positions go clockwise from outside:} &\quad \sigma_1^{-1}\sigma_3^{-1}\sigma_5^{-1}\ldots\sigma_{2n-1}^{-1} \\
\text{Everyone rotates clockwise:} &\quad \tau \\
\text{Everyone rotates counterclockwise:} &\quad \tau^{-1}
\end{aligned}$$

Note that

$$\begin{aligned}
(\sigma_0\sigma_2\sigma_4\ldots\sigma_{2n-2})^{-1} &= \sigma_{2n-2}^{-1}\ldots\sigma_4^{-1}\sigma_2^{-1}\sigma_0^{-1} \\
&= \sigma_0^{-1}\sigma_2^{-1}\sigma_4^{-1}\ldots\sigma_{2n-2}^{-1} \text{ (by the braid relations)}
\end{aligned}$$

It follows that there are just three generators of the fundamental dance moves with two teams. We define them as: $a = \sigma_0\sigma_2\sigma_4\ldots\sigma_{2n-2}, b = \sigma_1\sigma_3\sigma_5\ldots\sigma_{2n-1}$, and τ.

In an actual maypole dance, the dancers usually need to reverse their dance moves to unwind the braids. The unwinding is the same as applying the inverses in the reverse order. We will use the three levels to investigate two kinds of maypoles dances. Since the point is to understand the essential property, we will not focus on the unwinding process.

The Grand Chain.

In a Grand Chain dance, we have team A and team B facing each other. One team always goes clockwise and the other always goes counterclockwise. We can give an instruction to A dancers that they move clockwise, first pass their B neighbors from the outside, then pass the next person from the inside, and repeat. The instruction for B to follow will to be the opposite. Note that while moving clockwise, passing the person coming in the opposite from the outside implies that your braid crosses over the other's. Figure 6.28 shows the first two overhead dance moves of six people with A moving clockwise, first going outside, then inside and repeat.

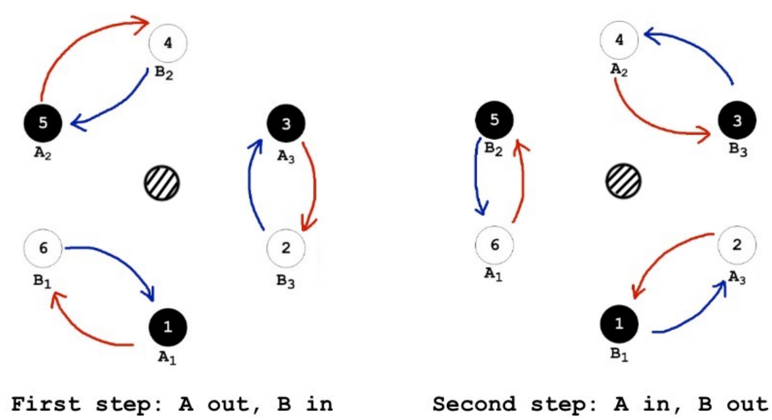

FIGURE 6.28. The Grand Chain overhead moves with six dancers. A moves clockwise while B moves counterclockwise.

The first step corresponds to the generator a^{-1}, and the second step corresponds to b. We observe that A members land in the spots of odd-number strings again after the first two steps. This means the generator for the third step is a^{-1} as well. Figure 6.29 shows the braid diagram of the first four moves.

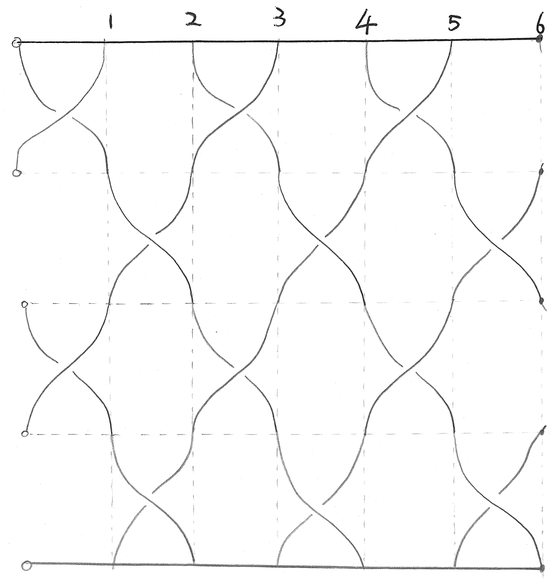

FIGURE 6.29. Braid diagram of Grand Chain

The instruction for this dance is $a^{-1}b$. To unwind, our instruction is the inverse of $a^{-1}b$, which is $b^{-1}a$. Notice that the Grand Chain braids followed by this specific instruction is a subgroup of all maypole dances. Its generator is $a^{-1}b$.

The followings are the possible instructions of the Grand Chain and the corresponding generators of the subgroups of dances.

$$\left.\begin{cases} \text{Team } A\text{: clockwise, out-in-out-in-}\ldots \\ \\ \text{Team } B\text{: counterclockwise, in-out-in-out-}\ldots \end{cases}\right\} \text{generator of the group: } a^{-1}b$$

$$\left.\begin{cases} \text{Team } A\text{: counterclockwise, out-in-out-in-}\ldots \\ \\ \text{Team } B\text{: clockwise, in-out-in-out-}\ldots \end{cases}\right\} \text{generator of the group: } ba^{-1}$$

$$\left.\begin{cases}\text{Team } A\text{: clockwise, in-out-in-out-}\dots\\[2ex] \text{Team } B\text{: counterclockwise, out-in-out-in-}\dots\end{cases}\right\}\ \text{generator of the group: } ab^{-1}$$

$$\left.\begin{cases}\text{Team } A\text{: counterclockwise, in-out-in-out-}\dots\\[2ex] \text{Team } B\text{: clockwise, out-in-out-in-}\dots\end{cases}\right\}\ \text{generator of the group: } b^{-1}a$$

The general description of the Grand Chain maypole dance gives us the idea to formulate some other dances in similar ways. For example, we can tell A and B to not repeat "in" and "out" in period of 2, but instead, to follow "out-out-in" or "in-in-out". Figure 6.30 shows the first three overhead steps of such a dance with six people.

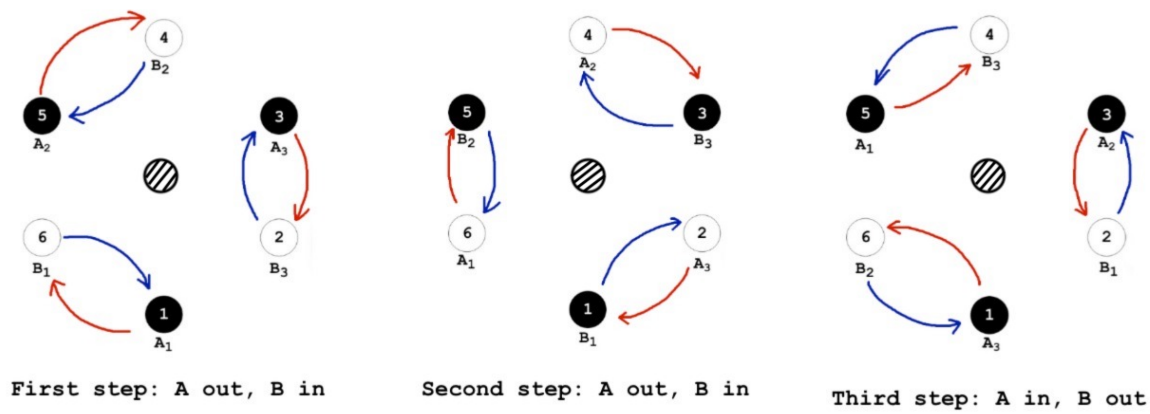

FIGURE 6.30. The Grand Chain overhead moves with 6 dancers. A moves clockwise while B moves counterclockwise.

Just like how we analyze the regular Grand Chain dance, this new dance follows $a^{-1}b^{-1}ab^{-1}a^{-1}b$ on the braid, which is the generator of this subgroup of dances. Figure 6.31 shows the braid diagram of the first six moves with six dancers.

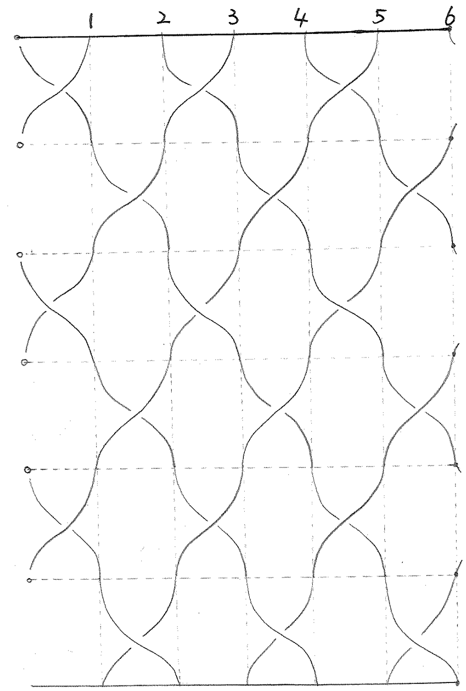

FIGURE 6.31. Braid diagram of the new Grand Chain

We define the number of fundamental dance moves, namely $a, a^{-1}, b, b^{-1}, \tau$ or τ^{-1}, in the generator of the subgroup dance as *period of the braid.* One interesting result is that we now have period of 6 for the braid diagram instead of 2 for the former case. The instruction repeats after every three steps, so we might have thought that the diagram has a period of 3. Why do we have the discrepancy? It is because the new dance brings team A dancers from odd-number spots to even-number spots after three movements. To start from the odd positions at downbeat again, we need another three movements to take A back. This shows that the period of the braid does not necessarily equal to the period of the instruction to the dancers.

The Gypsies' Tent.

In a Gypsies' Tent dance, we also have two alternating teams, A and B. However, the path of the moves is more complicated. Each dancer i from team A has a partner,

$i-1$. The members of team A move while the team B members stand still. A dancer i from team A dances around their partner $i-1$ counterclockwise, then around $i+1$ counterclockwise, then around $i-1$ counterclockwise again, $i-3$ counterclockwise and finally $i-1$ counterclockwise, returning to his or her original position. The trajectory of the member A_1 in a eight-dancer group is shown as an example in Figure 6.32.

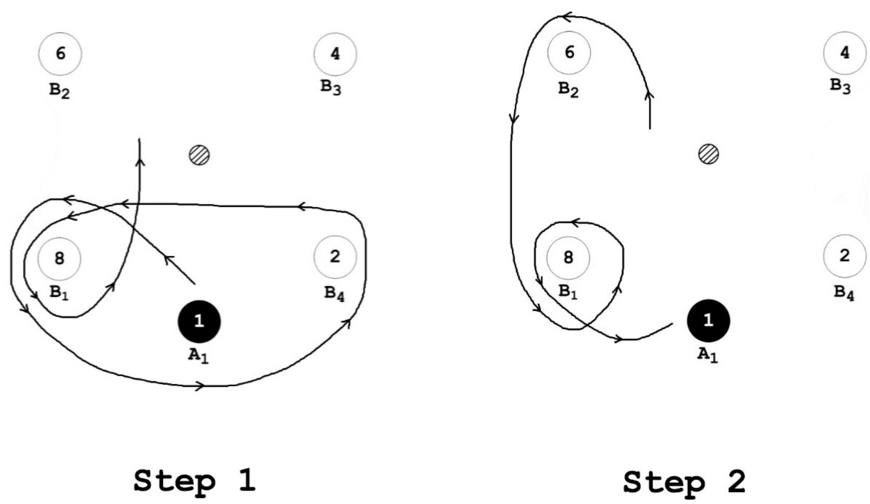

FIGURE 6.32. Gypsies' Tent dance move for dancer A_1 with eight dancers. To avoid confusion, only A_1 and team B dancers are shown.

When we talk about weaving around a person, there are actually several steps happening. For example, at the beginning, A_1 winds around B_1 in Figure 6.32. To be realized on the braid diagram, this is broken down into four steps: A_1 and B_1 interchange with B_1 over A_1, both of them rotate clockwise, then A_1 and B_1 interchange again but with A_1 over B_1 this time, and finally, both of them move counterclockwise. One thing to notice is that the description above was explained in terms of the dancer. However, all dancers move to different string positions at each step. A braid diagram focuses, instead, on the crossings of the strings at fixed positions, so the generators correspond to the positions of the strands. We need to remind ourselves about the correlation between each dancer and the string number at each step.

If we give all A dancers the instruction to complete their dance moves as shown in Figure 6.32, with a total of six dancers, the resultant braid diagram will look like Figure 6.33.

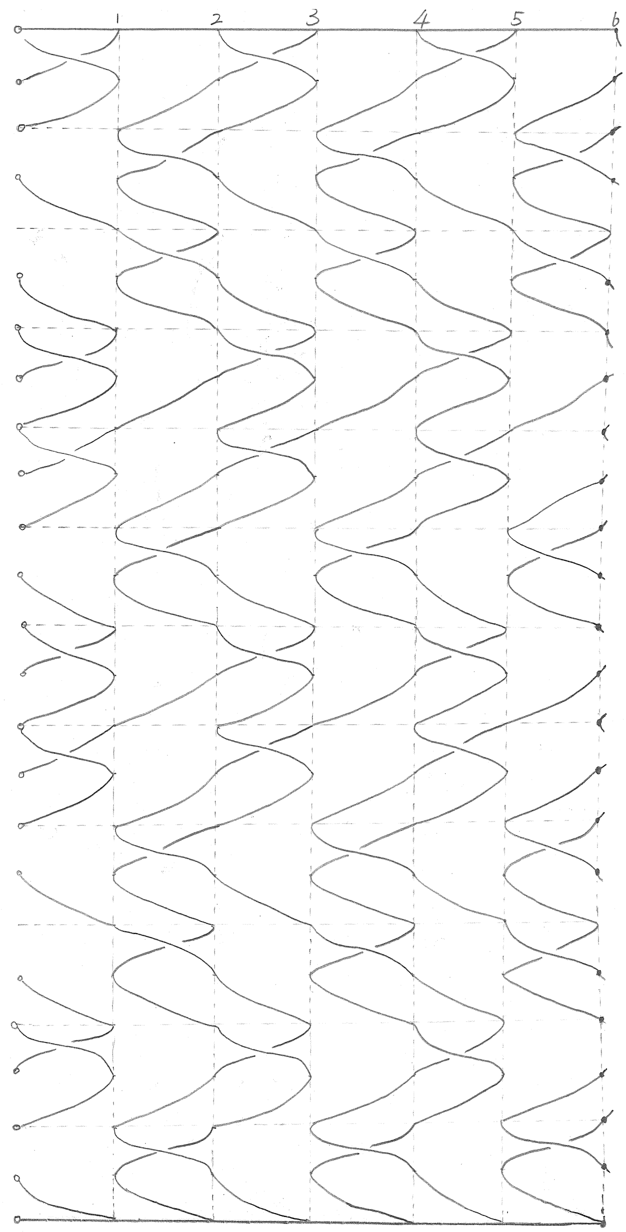

FIGURE 6.33. Braid diagram of Gypsies' Tent

If we write out the word of this braid, it will be

$$(\sigma_0\sigma_2\sigma_4\tau)(\sigma_1\sigma_3\sigma_5\tau^{-1})^2(\sigma_0\sigma_2\sigma_4\tau)^2(\sigma_1\sigma_3\sigma_5\tau^{-1})(\sigma_0\sigma_2\sigma_4\tau)^2(\sigma_1\sigma_3\sigma_5\tau^{-1})^2(\sigma_0\sigma_2\sigma_4\tau)(\sigma_1\sigma_3\sigma_5\tau^{-1})$$

$$=a\tau(b\tau^{-1})^2(a\tau)^2b\tau^{-1}(a\tau)^2(b\tau^{-1})^2a\tau b\tau^{-1}$$

Even though it looks complicated, the only effective generators here are $a\tau$ and $b\tau^{-1}$. Now, we can give a similar instruction to B dancers and tell each dancer i from team B to dance around their partner $i+1$ clockwise, then around $i-1$ clockwise, then around $i+1$ clockwise again, $i+3$ clockwise and finally $i+1$ clockwise, returning to the original position. We can apply the same analysis to dance moves of B as what we have done for A. The generators of their steps are $a^{-1}\tau^{-1}$ and $b^{-1}\tau$.

6.2. **Maypole dances with more than two teams.**

Even though maypole dances usually do not have more than two groups of dancers, we can investigate what will happen if we have three or four or more teams. Suppose we have $3n$ people where $n \in \mathbb{N}^+$. We divide them into team A, B and C and let members from A, B, C line up one by one in a circle, so that facing to the outside, team A dancers have Bs on their left and Cs on their right. The possible instructions are having A and B neighbors interchanged, A and C neighbors interchanged, B and C neighbors interchanged or the overall rotation. The generators of the dance moves are: $a = \sigma_0\sigma_3\sigma_6\ldots\sigma_{3n-3}, b = \sigma_1\sigma_4\sigma_7\ldots\sigma_{3n-2}, c = \sigma_2\sigma_5\sigma_8\ldots\sigma_{3n-1}$, and τ. Analogously, we can analyze cases for more teams.

7. Invariants

There is a large number of different maypoles dances. Which ones will get us the same braid diagram? This question can be answered by the word problem, which has been discussed in the previous section. We might also want to ask what can be a quick way for us to recognize that two dances definitely cannot yield the same braid, even after applying all possible sequences of the braid relations. One way to do this is to find invariants for annular braids. If the invariant is different for two braids, the braids are different. The number of strings is an obvious invariant for an annular braid.

Each τ generator gives the braid a rotation by one position. Note that from the relation $\tau\sigma_i\tau^{-1} = \sigma_{i+1}$ for annular braids, we get

$$\tau\sigma_i = \sigma_{i+1}\tau \tag{7.1}$$

This indicates that whenever there is a τ in a given word, we can move the τ to the right by adding 1 (mod n) to the index of the σ_i generator that is behind the τ. For τ^{-1}, it is $\tau^{-1}\sigma_{i+1} = \sigma_i\tau^{-1}$. Thus, we need to deduct 1 (mod n) from the index of the following σ-generator when moving the τ^{-1} backwards. The relation allows us to rewrite the word of every annular braid and get down to a word where all the τs and τ^{-1}s are found at the very end. After cancelling out the inverses, we can get the number of "net" τs. We call this integer *net rotation*, which is an invariant of annular braids.

For clarity, we let *the braid* be defined as the equivalence class of a braid up to ambient isotopy. We denote the net rotation of the braid $\mathcal{A}$ by $\mathbf{N}(\mathcal{A})$. For example, if both $\mathcal{A}$ and $\mathcal{A}'$ are on four strands, a given word representing $\mathcal{A}$ is $\sigma_3\tau\sigma_3\sigma_1^{-1}\sigma_0$ and a word for $\mathcal{A}'$ is $\tau^{-1}\sigma_0\sigma_1\tau\sigma_2^{-1}\sigma_1$, then we can compute their net rotations to see whether they are different. By the braid relations, $\sigma_3\tau\sigma_3\sigma_1^{-1}\sigma_0 = \sigma_3\sigma_0\sigma_2^{-1}\sigma_1\tau$, so $\mathbf{N}(\mathcal{A}) = 1$. For $\mathcal{A}'$, $\tau^{-1}\sigma_0\sigma_1\tau\sigma_2^{-1}\sigma_1 = \sigma_3\sigma_0\sigma_2^{-1}\sigma_1$, so $\mathbf{N}(\mathcal{A}') = 0$. Therefore, $\mathcal{A}$ and $\mathcal{A}'$ are different braids.

As we saw in the previous section, we can tell a team of dancers to move simultaneously and then tell another team to move next. The net rotation invariant also tells us that the number of τs in a word cannot be changed by applying the braid relations. If we leave the rotation out, is there anything special about the remaining σs? This question leads to the discovery of the following two invariants of both ordinary and annular braids - the crossing number and the the step number.

7.1. **The crossing number.**

The σ-generators represent the crossings in a braid. We cannot reduce the number of crossings by applying any of the annular braid relations. The only way we can decrease the number of crossings is by cancelling out inverses, such as $\sigma_i\sigma_i^{-1} = e$. Applying any of the braid relations will give us a transformation of the original word, but *the braid* is essentially not changed. Because all the τ-generators can be moved to the back in any given word, we can simply apply the braid relations and get the word down to such a form, then ignore the τs and only pay attention to the σs. Thus, we can represent the braid $\mathcal{A}$ with word A that contains only σ-generators. If we have a sequence of cancellations or applications of the braid identities to the word A, then we denote the sequence of the words as $A_1, A_2, \ldots$.

For the given word A, we let $C(A)$ denote the number of crossings for this word; it is simply the number of σ-generators in the word. We can apply the braid relations to transform the word of $A = A_0$ to any form A_i where $i \in \mathbb{N}^+$ and obtain the number of crossings for A_i, $C(A_i)$. Let $\mathbf{C}(\mathcal{A}) = \min\{C(A_i)|i = 0, 1, 2, 3, \ldots\}$ where the minimum is over all possible sequences of transformations. Then, $\mathbf{C}(\mathcal{A})$ is *crossing number* for $\mathcal{A}$.

Algorithm to get $\mathbf{C}(\mathcal{A})$.

Our concept of the crossing number is related to the shortest word problem in the braid group. The problem of finding a representative in a word class of B_n with the shortest length is proved to be at least as hard as an NP-complete problem ([BKL98]). According to Bangert ([Ban09]), there are cases where a braid must increase in length before getting to the shortest word. However, in general, we can at least achieve a local minimum, if not the global minimum number, by the greedy algorithm, in which we apply the braid relations to cancel as many pairs of generators as possible while never increasing the number of crossings during the search. Even though the solution of the shortest word problem is desirable for explaining maypole dances as well as some other physical phenomena related to knots and links, the difficulty of this problem actually opens up the application of braid words in cryptography ([BS10]).

7.2. **The step number.**

The exploration in maypole dances reveals the fact that the crossings can sometimes happen at the same time but that not all steps can be taken together. For example, we can do σ_1 with σ_3^{-1} simultaneously, but not with σ_2. Figure 7.34 shows the example.

Each maypole dance can be achieved by a minimum number of simultaneous dance moves. For any general annular braids, we can apply the same the idea and find the minimum number of steps for each of them. We call the minimum number of steps for the braid $\mathcal{A}$ as $\mathbf{D}(\mathcal{A})$, the *step number*.

For an ordinary braid on n strands where $n \geq 2$, suppose that the number of positive and negative generators is denoted by P and N, respectively. The worst case is that no crossing can happen simultaneous with any other ones, so the upper bound of the

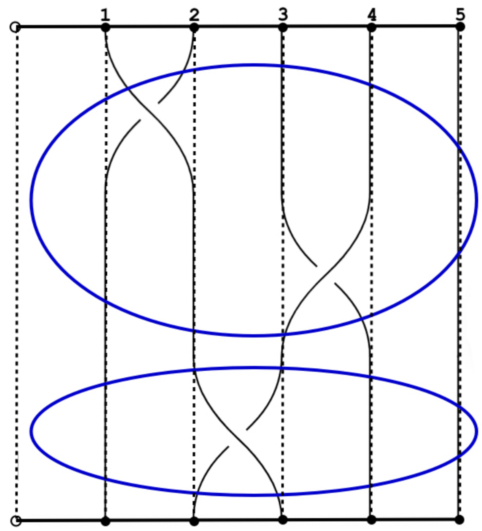

FIGURE 7.34. σ_1 and σ_3^{-1} can happen simultaneously, but σ_2 can only be completed by itself.

minimum number of steps is $P + N$. One example is a braid on 3 strands with the word $s_1 s_2^{-1} s_2^{-1} s_1$. The number of positive generators is 2 and the number of negative generators is also 2. Since s_1 and s_2^{-1} are too close to be carried out simultaneously, the step number of this braid is $2 + 2 = 4$.

The best we can do is to have all positive or all negative generators cancel with the corresponding pairs and the remaining $|P - N|$ generators are spread out as much as they can. The largest number of generators we can put into one level is $\lfloor \frac{n}{2} \rfloor$ in B_n. For example, consider a braid on 7 strands with the word $s_1 s_3 s_5 s_1 s_3 s_5 s_2 s_4 s_6$. The generators s_1, s_3 and s_5 can be at the same level, and also, the generators s_2, s_4 and s_6 can be at the same level. In fact, these are the best that we can do and it is impossible to place more than 3 generators at the same level for $n = 7$. The formula tells that the largest number of generators we can put into one level is $\lfloor \frac{7}{2} \rfloor = 3$, which is consistent with our observation. We then can assert that the lower bound is $\left\lceil \frac{|P-N|}{\lfloor \frac{n}{2} \rfloor} \right\rceil$.

Analogously, for an annular braid on n strands where $n \geq 3$, suppose that the number of positive generators is P, the number of negative generators is N and the absolute value of net rotation is T. The worst case is that no generator can be grouped with any other ones, so the upper bound of the step number is $P + N + T$.

Similarly, the best we can do is that all positives or negatives are cancelled with their pairs and the remaining ones are far enough away from each other that they can be grouped into a small number of levels. The largest number of generators we can put into one group is $\lfloor \frac{n}{2} \rfloor$ in AB_n. Note that neither τ nor τ^{-1} can be applied with any other generators at the same time, so T is the number of groups for the rotation generators. Hence, the lower bound is $\left\lceil \frac{|P-N|}{\lfloor \frac{n}{2} \rfloor} \right\rceil + T$.

Bounds on the step number given an initial word.

Now, if we know a specific word of the given braid, the first thought will perhaps be to find a lower bound and an upper bound and squeeze towards the value of the step number. Suppose the braid we want to investigate is $\mathcal{A}$ given by the initial word A. As discussed above, we can find P, N and T for A and therefore calculate the lower bound.

To find a lowered upper bound, we must examine the σ-generators. If we consider the extreme case in which we cannot apply any braid relations, then the given word is fixed after canceling inverses. First, we use the braid relation (7.1) to move all the τs in A to the right, set aside these τ-generators and denote the remaining word with purely σ-generators by A_1. Then, we cancel all pairs of inverses to obtain A_1'. Recall that in a dance, if two neighbors exchange positions, then at this moment, it is impossible for them to exchange positions with the other neighbors. This implies that no two σs with the indices having a difference of 0 or 1 can be applied simultaneously. Note that in an annular braid, σ_{n-1} and σ_0 have a difference of 1. Following this rule, we can start grouping the σs in A_1'. Beginning from the left, we put the generators in the same group if no two of them have indices with a difference less than 2 (modulo n). Otherwise, if the next generator on the line is too close to any one of the generators in the current group, then we mark this generator as the start of the new group. Finally, we will count the number of groups. To distinguish from the step number of the braid, we denote the number of groups obtained by this procedure for a braid word A_1' as $D(A_1')$. Adding this number to the absolute value of the net rotation gives us an upper bound of $\mathbf{D}(\mathcal{A})$.

For example, consider the word $A = \sigma_0\sigma_4^{-1}\tau^{-1}\sigma_3\sigma_4\sigma_4^{-1}\tau\sigma_2\sigma_5\sigma_0\tau^{-1}\sigma_2^{-1}$ of six strings. The number of positive generators is 6, the number of negative generators is 3, and the net rotation is $\mathbf{N}(\mathcal{A}) = -1$. So, $n = 6, P = 6, N = 3$ and $T = |\mathbf{N}(\mathcal{A})| = 1$. Then, $\left\lceil \frac{|P-N|}{\lfloor \frac{n}{2} \rfloor} \right\rceil + T = \left\lceil \frac{|6-3|}{\lfloor \frac{6}{2} \rfloor} \right\rceil + 1 = 2$, which is the lower bound.

As for the upper bound, by repeatedly applying the braid relation of rotation, we can rewrite A as $\sigma_0\sigma_4^{-1}\sigma_2\sigma_3\sigma_3^{-1}\sigma_2\sigma_5\sigma_0\sigma_1^{-1}\tau^{-1}$. Removing τ-generators gives us the σ-part of the word, which we denote $A_1 = \sigma_0\sigma_4^{-1}\sigma_2\sigma_3\sigma_3^{-1}\sigma_2\sigma_5\sigma_0\sigma_1^{-1}$. After canceling inverses, we yield $A_1' = \sigma_0\sigma_4^{-1}\sigma_2\sigma_2\sigma_5\sigma_0\sigma_1^{-1}$. The grouping is $(\sigma_0\sigma_4^{-1}\sigma_2)(\sigma_2\sigma_5)(\sigma_0)(\sigma_1^{-1})$. Since σ_2 is 0 from the σ_2 in the first group, we start counting a new group from it. Also, because there are six strings, not only σ_0 and σ_1 are neighbors, but σ_0 and σ_5 are also neighbors. Thus, σ_0 and σ_1 are in their own groups. We have $D(A_1') = 4$. Figure 7.35 shows each step geometrically. We then yield a bound on the step number of $\mathcal{A}$: $2 \leq \mathbf{D}(\mathcal{A}) \leq 4 + 1 = 5$.

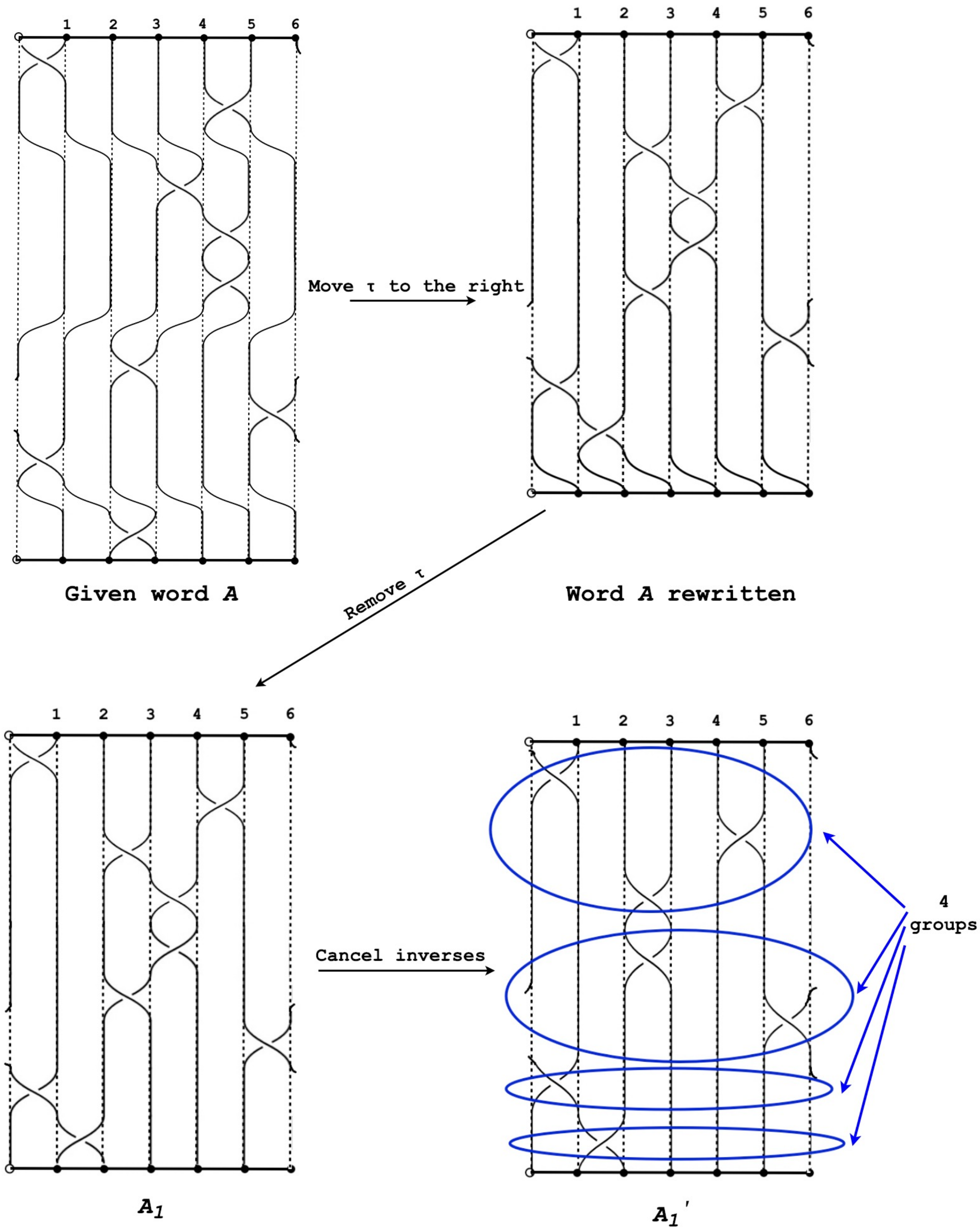

FIGURE 7.35. Steps to find out number of grouping for word A on six strings

Idea of drawing directed graphs for ordinary braids.

Since a braid has a geometric presentation, it is possible to analyze it by graph. For simplicity, we illustrate with ordinary braids. The annular braid case is similar except that we view σ_0 and σ_{n-1} as neighbors and we have the rotations, τ and τ^{-1}. The fact that we cannot have σ_{i-1} and σ_{i+1} at the same time with σ_i says that the presence of σ_i prevents the σ_{i-1} and σ_{i+1} from being at the same level as σ_i. This leads us to think of toy blocks sliding on rigid poles. Figure 7.36 shows an example.

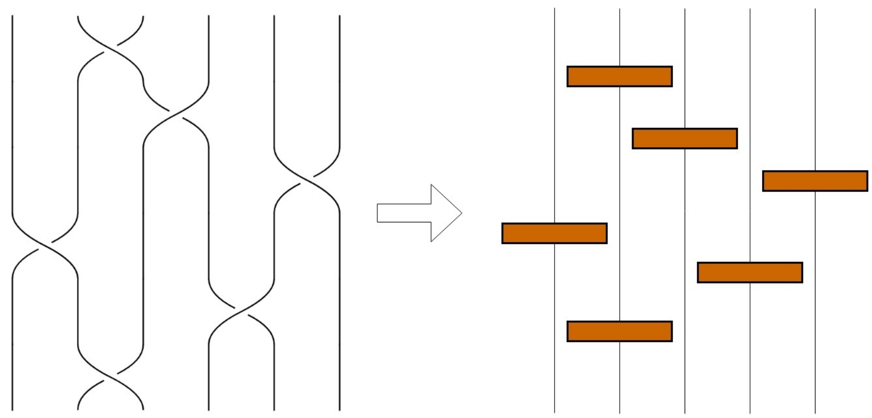

FIGURE 7.36. Braid $\sigma_2\sigma_3^{-1}\sigma_5\sigma_1\sigma_4^{-1}\sigma_2$. The generators are shown as the blocks.

The sticks correspond to the generators. In particular, a block on the i^{th} stick means that there is a twist between the i^{th} and the $(i+1)^{st}$ strands, so the corresponding generators are σ_i and σ_i^{-1}. We can further imagine that the blocks are free to slide up and down the sticks. Due to gravity, all the blocks will fall to the bottom (see Figure 7.37).

Interestingly, the blocks fall down into three levels in Figure 7.37. This relates back to the fact that some of the moves cannot happen at the same time. If we check the number of groups of the word, we will find that $(\sigma_2)(\sigma_3^{-1}\sigma_5\sigma_1)(\sigma_4^{-1}\sigma_2)$ yields three exactly.

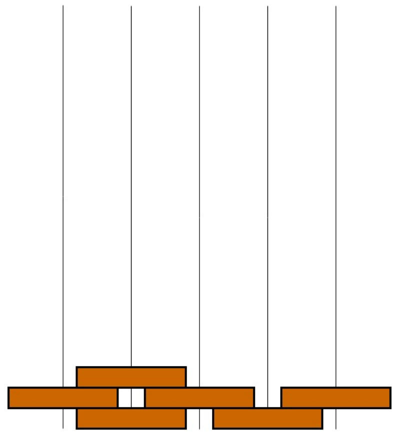

FIGURE 7.37. Blocks of $\sigma_2\sigma_3^{-1}\sigma_5\sigma_1\sigma_4^{-1}\sigma_2$ fall down

From this idea, we can formulate a way of creating the directed graph of a given word. If there are n strands in total, then we first draw $n-1$ sticks to represent the generators' positions. Next, we draw circles from the top to bottom while we read through the word. Whenever we see σ_i, we draw a solid dot on the i^{th} stick. Whenever we see σ_i^{-1}, we draw a hollow circle on the i^{th} stick. Figure 7.38 shows an example.

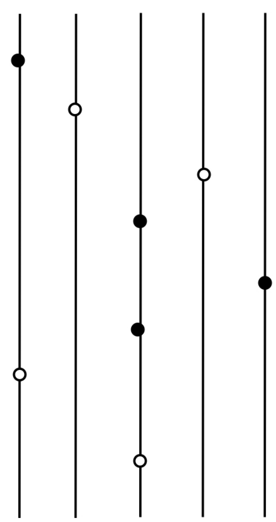

FIGURE 7.38. Word $\sigma_1\sigma_2^{-1}\sigma_4^{-1}\sigma_3\sigma_5\sigma_3\sigma_1^{-1}\sigma_3^{-1}$

We first cancel the inverses. The cancellation happens when we have σ_i and σ_i^{-1} next to each other on the i^{th} stick. Even though the difference in their heights does not need to be 1, we must guarantee that there is no $\sigma_i, \sigma_i^{-1}, \sigma_{i-1}, \sigma_{i-1}^{-1}, \sigma_{i+1}$ or σ_{i+1}^{-1} on the levels between the two. We can either choose to do the cancellation before we draw out the diagram of the circles, or we can do that directly on the diagram as shown in Figure 7.39.

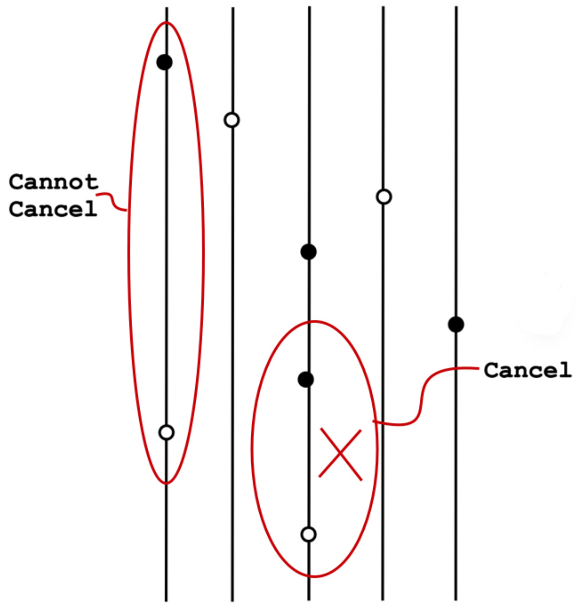

FIGURE 7.39. Canceling σ_3 with σ_3^{-1}

Now we have simplified the word. The next step is to draw direct edges between the circles. The rule of drawing is that for each of the σ_is in the word, we look at the $(i-1)^{st}, i^{th}$ and $(i+1)^{st}$ sticks and scan downwards from the height of the σ_i we choose to begin with. If the first circle that shows up is at i, then draw one arrow from the beginning σ_i to this following σ_i; if the first one is at $i-1$, and the second neighbor is at i or $i-1$, then draw only one arrow from σ_i to the first σ_{i-1}, but if the second is at $i+1$, then draw two arrows, from σ_i to both σ_{i-1} and σ_{i+1}; if the first is at $i+1$ and the second neighbor is at i or $i+1$, then draw only one arrow from σ_i to the first σ_{i+1}, otherwise, draw two arrows, from σ_i to both σ_{i-1} and σ_{i+1}. Graphically, the fundamental cases are shown in Figure 7.40. Note that symmetries with respect to the middle stick are also possible. The procedure yields a *directed acyclic graph*. Figure 7.41 shows the directed graph of $\sigma_1\sigma_2^{-1}\sigma_4^{-1}\sigma_3\sigma_5\sigma_3\sigma_1^{-1}\sigma_3^{-1}$ after cancellation.

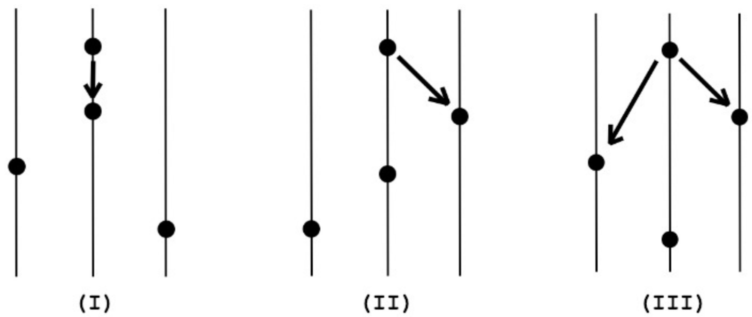

FIGURE 7.40. Basic cases of drawing the direct edges

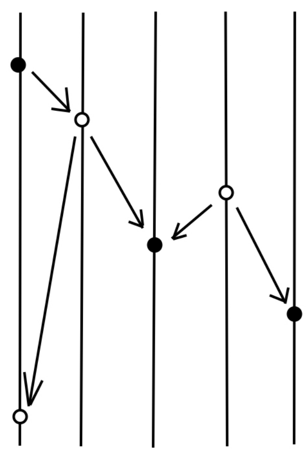

FIGURE 7.41. Directed graph after cancellation

Then we can use the directed graph to find a lowered upper bound of the step number of this braid. There are two ways of doing it. One is to trace the arrows from top to bottom and find the longest possible path on the graph. This is known as the *longest path problem* of directed acyclic graph, which has associated algorithms available providing the solution. The number of arrows on the longest path is one less than the minimal number of steps. The number for the diagram in Figure 7.41 is $2 + 1 = 3$. We can also find out the number by grouping. The strategy is that we start from the bottom and group every circle without arrows pointing out as one group. Then, we remove these points, and apply the same method to the points at the new bottom. Note that along the way, we also find out the simultaneous dance moves at each step. This grouping idea is called *layered graph drawing*. There are algorithms proposed for creating such drawings. Figure 7.42 shows that we get 3 as well by grouping.

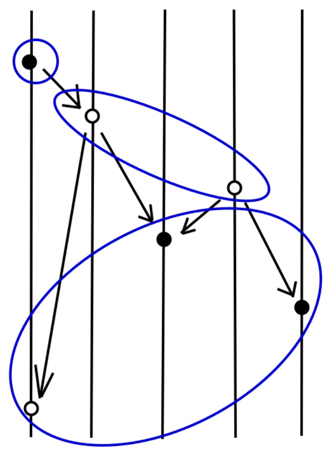

FIGURE 7.42. Grouping of the directed graph

Note that the method above considers neither the braid relation $\sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1}$ nor the equivalent braids with longer word length. The strategy illustrated in this section does not yield the step number of a given braid $\mathcal{A}$ since there are other words in the equivalence class that we cannot get from drawing the directed graph. However, it can efficiently lower the upper bound and give a better approximation of $\mathbf{D}(\mathcal{A})$, compared with the bounds provided in the previous section. If a given braid has a relatively short braid word, or a repetitive pattern like the ones produced in maypole dances, then this approach can be applied as a handy tool to quickly analyze the braid with pencil and paper.

An improved algorithm of approximating the step number.

On a directed graph, the two braid relations are shown as in Figure 7.43 and Figure 7.44.

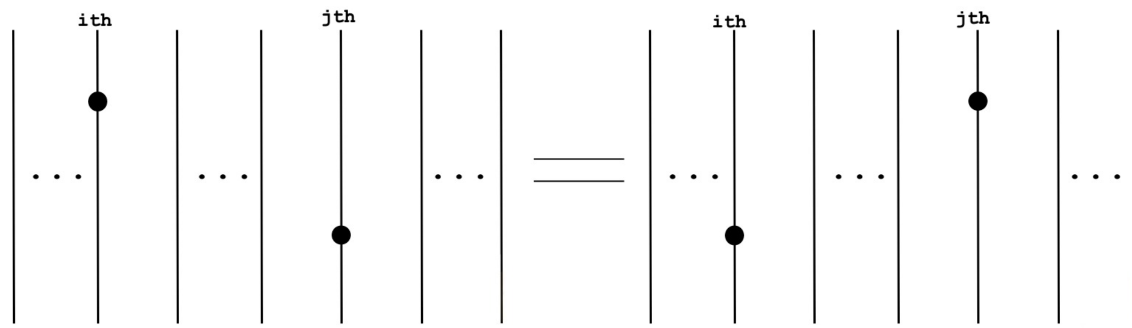

FIGURE 7.43. The braid relation $\sigma_i\sigma_j = \sigma_j\sigma_i$ on directed graph.

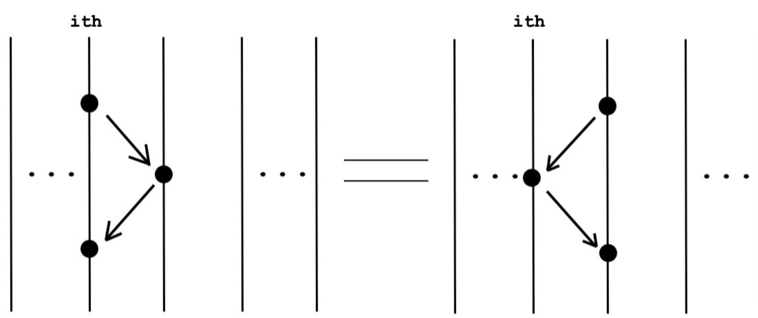

FIGURE 7.44. The braid relation $\sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1}$ on directed graph.

One tricky thing about finding the step number is that we cannot apply the "greedy algorithm" which only executes the optimal choice at each step when finding the globally optimal solution. Figure 7.45 is a counterexample showing a case where we do not necessarily get the step number of a braid by only going for better cases. For example, if we start from the braid diagram 1, the greedy algorithm will tell us to stop and conclude that 5 is the best we can do since applying braid relations can only get us to either 2 or 8, which will be worse. However, from the figure, we can see that the step number should be 4. Thus, there are cases where we have to go worse to get to the optimal.

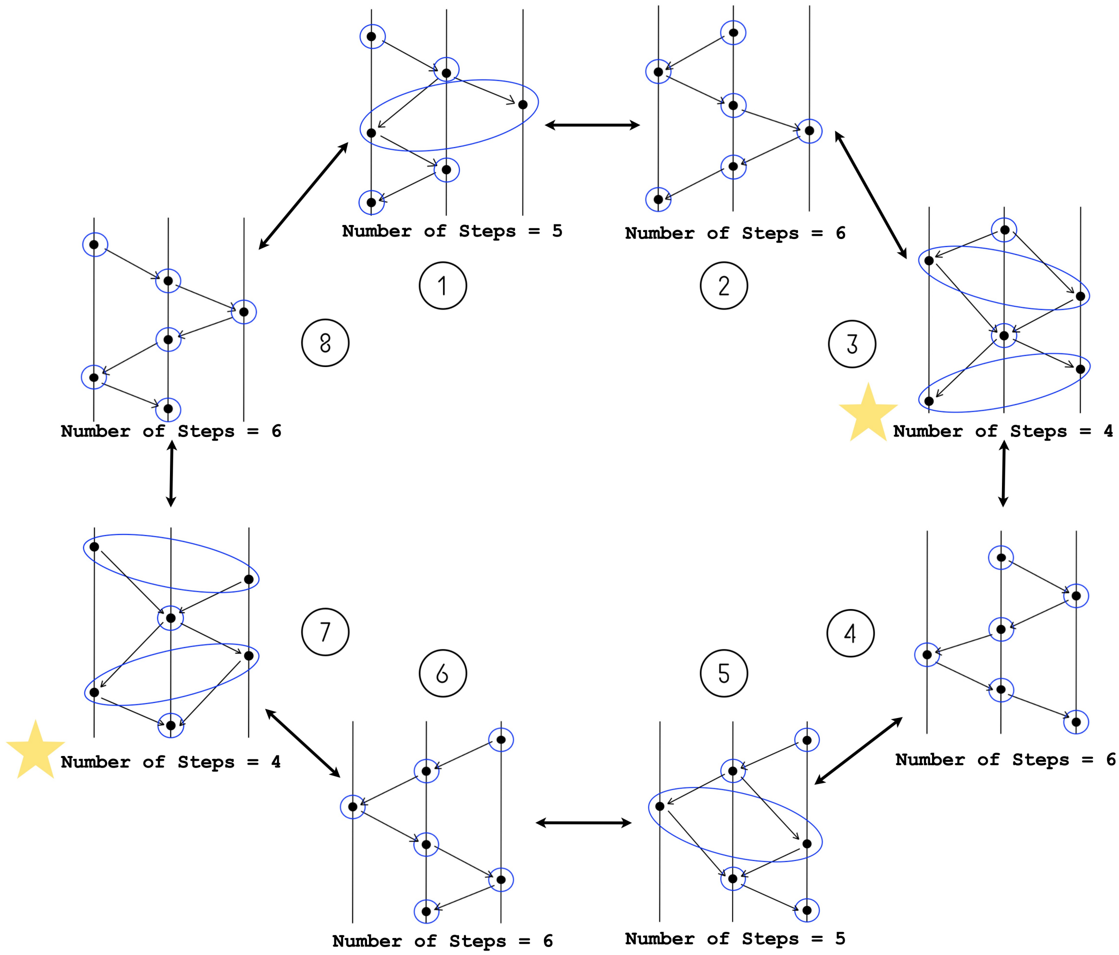

FIGURE 7.45. Counterexample showing the greedy algorithm does not work

In order to get a lowered number of steps, what we did before drawing the directed graph was to cancel as many inverses as possible. This leads us to the question of the number of crossings. Even though the shortest word problem in B_n is proved to be at least as hard as an NP-complete problem, we can apply the following algorithm to help get an approximation of $\mathbf{D}(\mathcal{A})$ for a given braid $\mathcal{A}$. We first apply the braid relation to gather all the τ-generators to the back of the given word. Then, we take the front part with only σ-generators and use the greedy algorithm to cancel as many pairs of generators as we can. We will be left with the smallest number of crossings that we can get at this point. Next, we apply the two braid relations to do an exhaustive search on this remaining word. For each transformation, we apply the method of directed graph to find out the number of groupings of the associated word. Since the number of crossings is finite and definite, there is only a finite number of cases to check. The minimum number from all of these braids is an upper bound of the step number.

In short, what we are essentially doing is to first get down to the word with local minimal length, and then enumerate all possible transformations to find the minimum number among the corresponding grouping numbers. Even though the difficulty in finding the word with global minimal length prevents us from checking all equivalent braids with the same length, our hypothesis is that this algorithm can at least get us the local minimum.

Specifically, our conjecture is: Given an initial braid word A_n^0 with length n, let $A_n^1, A_n^2, A_n^3, \ldots$ be a sequence of transformations of words of length n, attained after applying any sequence of the two braid relations on A_n^0. Let A_{n-2}^0 be the word obtained after canceling one pair of inverses in A_n^i for some $i \in \{0\} \cup \mathbb{N}^+$. Then, similarly, denote the transformations of A_{n-2}^0 of words of length $n-2$ by $A_{n-2}^1, A_{n-2}^2, A_{n-2}^3, \ldots$. We analogously continue the definition until there is no possible cancellation in any transformations of the words. Suppose the cancellation terminates at the sequence $A_{n-2k}^0, A_{n-2k}^1, A_{n-2k}^2, A_{n-2k}^3, \ldots$ for some $k \in \{0\} \cup \mathbb{N}^+$ and $k \leq \frac{n}{2}$. (Note: k is the number of cancellations of inverses from A_n^0 to A_{n-2k}^0 .) For all $j \in \{0, 1, 2, \ldots, k\}$, define $d_j = \min\{D(A_{n-2j}^u) | u \in \{0\} \cup \mathbb{N}^+\}$. Then, for any $v, w \in \{0, 1, 2, \ldots, k\}$ where $v \geq w$, we have $d_v \leq d_w$.

Note that in the process mentioned in the conjecture, we strictly control the length of the braid words. Even though among the transformations of A_n^0 with length n, there might be multiple ones being able to be cancelled by one pair of inverses and shortened to a word with length $n-2$, we only choose one of them and shorten its word to A_{n-2}^0. Once we cancel a pair of generators, we never allow the word length to come back up again. The same control applies up to the end when we get A_{n-2k}^0. Thus, each time, we are only considering one subgroup of all words that can be obtained by strict reduction in length. However, the validity of this conjecture would guarantee that our improved algorithm can yield a local minimum of the step number.

8. Conclusion and Future Work

In this thesis, we have discussed and compared three presentations of the annular braid group. Specifically, we modified the definition of the new set of generators for annular braid group mentioned in [TB15], proved algebraically that there is an isomorphism between this new group and the annular braid group, and provided the formal presentation of this group.

Furthermore, we used three levels - the instructions, the overhead moves and the braid diagram - to describe the traditional maypole dances. The concise descriptions allow us to capture the essentials of the dances and further derive new dances more easily using the abstract languages.

Finally, we proposed invariants of the annular braids - the crossing number and the step number - with inspirations from the maypole dances. The crossing number is analogous to the shortest word problem in B_n. For the step number, we developed a method of producing the directed graph of a given word, which graphically gives a lowered upper bound of the step number of a given braid. The step number is the minimum grouping number among all words in the equivalence class. By asserting a conjecture, we drafted a greedy algorithm to approximate the step number.

The shortest word problem is a historically hard question, for which no algorithm has been provided to efficiently give a solution. A greedy algorithm is oftentimes a good approximation, but we do not know when it will break and how close its solution is to the real one. The application of braid word in cryptography is based on the difficulty of this problem. Further exploration of the shortest word problem can not only reinforce the security of a cryptosystem that employs braid words, but also reveal associated properties of knots and links. Moreover, We were not able to provide a thorough proof for our conjecture, which states that among the equivalent braid words obtained from one branch of strict reduction of the original word, no word with a larger crossing number can have a smaller number of steps than the minimum of all the numbers of steps of the words with a smaller crossing number. If this conjecture is proved to be true, then we are at least confident to attain a local minimum number of steps for a braid. In the future, it will be interesting to further investigate and complete the solutions of finding the crossing number and the step number.

References

[Ada94] Colin C. Adams. *The knot book*. W. H. Freeman and Company, New York, 1994. An elementary introduction to the mathematical theory of knots.

[Art47] E. Artin. Theory of braids. *Ann. of Math. (2)*, 48:101–126, 1947.

[Ban09] Patrick D. Bangert. Braids and knots. In *Lectures on topological fluid mechanics*, volume 1973 of *Lecture Notes in Math.*, pages 1–73. Springer, Berlin, 2009.

[BB05] Joan S. Birman and Tara E. Brendle. Braids: a survey. In *Handbook of knot theory*, pages 19–103. Elsevier B. V., Amsterdam, 2005.

[BKL98] Joan Birman, Ki Hyoung Ko, and Sang Jin Lee. A new approach to the word and conjugacy problems in the braid groups. *Adv. Math.*, 139(2):322–353, 1998.

[BR04] Elizabeth L. Bouzarth and David Richeson. Topological helicity for framed links. *J. Knot Theory Ramifications*, 13(8):1007–1019, 2004.

[BS10] A.J. Berrick and N.U.S.I.M. Sciences. *Braids: Introductory Lectures on Braids, Configurations and Their Applications*. Lecture Notes Series - National University of Singapore, Institute for Mathematical Sciences. World Scientific, 2010.

[Cro04] Peter R. Cromwell. *Knots and links*. Cambridge University Press, Cambridge, 2004.

[Fed03] Elie Feder. *Algorithmic problems in the braid group*. ProQuest LLC, Ann Arbor, MI, 2003. Thesis (Ph.D.)–City University of New York.

[KP02] Richard P. Kent, IV and David Peifer. A geometric and algebraic description of annular braid groups. *Internat. J. Algebra Comput.*, 12(1-2):85–97, 2002. International Conference on Geometric and Combinatorial Methods in Group Theory and Semigroup Theory (Lincoln, NE, 2000).

[Kra05] Daan Krammer. MA4F2 Braid groups (lecture notes), March 2005. `http://homepages.warwick.ac.uk/~masbal/MA4F2Braids/braids-lectures.pdf`. Last accessed on April 14, 2019.

[KT08] Christian Kassel and Vladimir Turaev. *Braid groups*, volume 247 of *Graduate Texts in Mathematics*. Springer, New York, 2008. With the graphical assistance of Olivier Dodane.

[Liv93] Charles Livingston. *Knot theory*, volume 24 of *Carus Mathematical Monographs*. Mathematical Association of America, Washington, DC, 1993.

[Sti93] John Stillwell. *Classical topology and combinatorial group theory*, volume 72 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, second edition, 1993.

[TB15] Jean-Luc Thiffeault and Marko Budišić. Braidlab: A Software Package for Braids and Loops, 2015. `https://arxiv.org/abs/1410.0849`. Last accessed on April 14, 2019.