Diagrammatic Computation of Morphisms Between Bott-Samelson Bimodules via Libedinsky's Light Leaves

Evan Chen   Francisco Unda

Research Science Institute

January 15, 2014

We consider the left/right action-preserving morphisms between Bott-Samelson bimodules from a combinatorial perspective; specifically, by using a diagrammatic representation. We examine the maps formed by a basis known as Libedinsky's light leaves, which constructs a map based on a string $\underline{r}$, composed of the letters s and t, and a binary string $\underline{b}$. The eventual goal is to find a formula for the composition of two such maps based on the two original binary strings. The paper provides a complete formula for the case $\underline{r} = sss\dots$, and shows that the possible structures of maps when $\underline{r} = ststs\dots$ are, in a certain sense, limited. We additionally provide a formula for the number of maps between two arbitrary Bott-Samelson bimodules.

1 Introduction

The Hecke Algebra (or Iwahori-Hecke Algebra) is an object of significant interest, studied in representation theory, algebraic number theory, and combinatorics. In 1979, Kazhdan and Lusztig [1, 2] gave a categorification of the Hecke algebra, using a technique involving *perverse sheaves on the flag variety.* The result became known as the Hecke category. Soergel [3, 4] simplified this construction significantly in 1990 with an algebraic approach, using what were later called Soergel bimodules. The Soergel bimodules permit an easier understanding of this categorification; their advantage is that they are easier to define and are combinatorial in nature. For a full treatise, we refer the reader to the introduction of Elias and Williamson [5].

Soergel bimodules can be described in terms of another class of bimodules, known as the *Bott-Samelson bimodules.* The morphisms of Bott-Samelson bimodules admit a combinatorial description. In particular, Elias and Williamson [5], building off earlier works such as Elias and Khovanov [6], give a presentation for the category of these morphisms in terms of colored planar diagrammatics (i.e. colored diagrams).

This paper investigates some of the properties of these diagrammatics from a combinatorial perspective. We consider only the special case where the underlying group W of the Hecke algebra is an infinite dihedral group with generators s and t (as opposed to an arbitrary Coxeter group W).

Given a string of length n composed of the letters s and t, as well as a binary string of length n, one can construct such a diagram using an algorithm known as Libedinsky's light leaves. The light leaves were derived by Libedinsky in [7] and are detailed in Section 2.4. These light leaves are interesting because, modulo certain "lower terms", they form a basis for a morphism spaces, as shown in [5]. In this paper, we develop tools to understand the composition of two such maps given the original binary strings, with the eventual goal of deriving a combinatorial formula.

Section 2 contains the preliminaries for the problem; it describes the light leaves and defines the relation between diagrams and maps. A minimal set of definitions used in this paper is given in Section 3. In Section 4, we give the problem statement in its full technicality.

The results begin by solving the special case of one color completely in Section 5. In Section 6, we also present a formula for the number of maps between two given bimodules. The most important results of this paper are presented in Sections 7 and 8, where we investigate a special "alternating" case. We show that the possible resulting structures are extremely limited in a certain sense. We then exploit these restrictions to provide a partial recursion to compute the compositions of light leaves in the alternating case.

2 Background

Let V be a (real) Euclidean space endowed with a positive definite symmetric bilinear form (μ, λ). Let W be a group with two generators s and t corresponding to reflecting over two hyperplanes in V, and let α_s and α_t be arbitrary vectors normal to these planes. We assume that the angle formed by α_s and α_t is not a rational multiple of π, so that the order of st in W is infinite.

In other words, W is the infinite dihedral group with the presentation

$$W = \left\langle s, t \mid s^2 = t^2 = 1 \right\rangle .$$

Then for any $a \in V$, reflection over the hyperplane normal to a can be written explicitly for each $v \in V$ as the map

$$v \mapsto v - \frac{2(v,a)}{(a,a)}a.$$

In particular, if W acts on V by this reflection, $s(\alpha_s) = -\alpha_s$ and $t(\alpha_t) = -\alpha_t$. Furthermore, using the above, we see that there exist fixed constants x and y such that

$$\begin{aligned} s(\alpha_t) &= \alpha_t + x\alpha_s \\ t(\alpha_s) &= \alpha_s + y\alpha_t. \end{aligned}$$

Explicitly, $x = \frac{-2(\alpha_s,\alpha_t)}{(\alpha_s,\alpha_s)}$ and $y = \frac{-2(\alpha_t,\alpha_s)}{(\alpha_t,\alpha_t)}$.

Now we define the *Coxeter ring* by $R = \mathbb{R}[\alpha_s, \alpha_t]$. Then W acts on R by precisely the same algorithm as described above.

Let R^s be the subring of R which is invariant under s. In other words,

$$R^s \stackrel{\text{def}}{=} \{r \in R \mid s(r) = r\}.$$

Define R^t similarly. Then a *Bott-Samelson bimodule* is a bimodule of the form

$$R \otimes_{R^{i_1}} R \otimes_{R^{i_2}} R \otimes_{R^{i_3}} \cdots \otimes_{R^{i_n}} R,$$

where each i_j is either s or t. We write elements of such a bimodule in the form $f_0 \mid f_1 \mid \cdots \mid f_n$, where $f_i \in R$.

We consider maps between these bimodules that preserve left and right actions; that is, we consider the maps σ such that

$$\sigma(rx) = r\sigma(x) \quad \text{and} \quad \sigma(xr) = \sigma(x)r \quad \text{for each } r \in R \text{ and } x \in \text{Domain } \sigma. \tag{2.1}$$

These maps may be represented diagrammatically.

2.1 Diagrammatics of Maps

These maps can be described in terms of diagrams. We take a detour and first describe the appearance of such diagrams; in the subsequent section we will explain their algebraic meaning. A reader who is not interested in this context can simply accept on faith the relations specified in Section 2.3.

Consider a category $\mathcal{D}$ whose elements are linear combinations $\sum c_i \Box$ of the diagrams described below (where the $\Box$'s are diagrams). The coefficients c_i belong to the ring $\mathbb{Z}[x, y]$.

The diagrams may be described as planar graphs, not necessarily connected, drawn in a rectangle, with the following properties.

(i) Vertices may lie on the upper or lower boundary, but not on the left or right boundary. In other words, the graph is embedded in $\mathbb{R} \times [0, 1]$.

(ii) Each vertex has degree 1 or 3.

(iii) Each connected component is colored either blue or red.

The vertices on the boundary are by convention not explicitly shown, but are nonetheless labelled s or t for blue or red, respectively.

An example of such a diagram is shown in Figure 1.

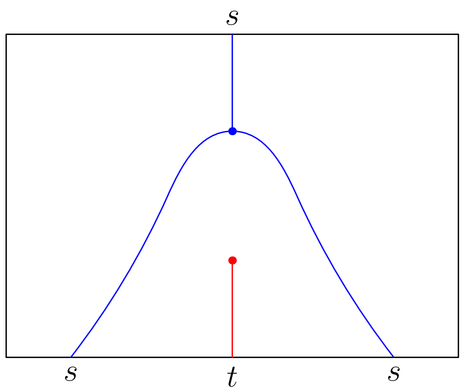


Figure 1: An example of a possible diagram.

The elements of $\mathcal{D}$ can be added by simply adding corresponding coefficients in front of each diagram; no other relations exist. Multiplication is defined as follows: the product of two diagrams is the composition of the diagrams if the labels on the bottom of the first coincide with those on the top of the second. By composition, we mean that the first diagram is vertically juxtaposed on top of the second diagram; see Figure 3 for an example. Otherwise, the product is 0.

2.2 Algebraic Context

The maps are conventionally read from bottom to top. The labels on the upper and lower boundaries specify the domain and codomain of the map by "transcribing" the tensor products to take; for example, a map with bottom stt and top s represents a map with domain $R \otimes_{R^s} R \otimes_{R^t} R \otimes_{R^t} R$ and codomain $R \otimes_{R^s} R$. An unlabelled domain corresponds to R.

Let us introduce one final definition.

Definition 2.1. The *Demazure operator* $\partial_s : R \to R$ is given by

$$\partial_s(f) = \frac{f - s(f)}{\alpha_s}.$$

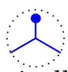
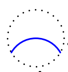
With this, we can now describe the meanings of each type of vertex (for blue lines) in Table 1. The corresponding equations hold for red in t.

Note that the descriptions are sufficient to determine outputs for all values because these morphisms respect left and right actions, as prescribed in (2.1).

Maps are composed by juxtaposition, and disjoint portions of the diagrams act independently. Therefore, by a combination of these structures, we can generate arbitrarily complicated maps.

Example. The diagram in Figure 1 represents a map

$$R \otimes_{R^s} R \otimes_{R^t} R \otimes_{R^s} R \to R \otimes_{R^s} R \quad \text{by} \quad a \mid b \mid c \mid d \mapsto a\partial_s(bc) \mid d.$$

Henceforth, we will make the convenient abbreviation of  as . It will also be understood that edges need not be straight lines, but any topologically equivalent deformation shall represent the same graph. With this understanding, any of the graphs described in the previous section can be viewed as compositions of the primitive structures we describe here.

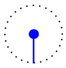
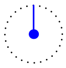
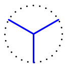
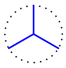
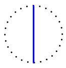
Map	Modules	Description
	$R \otimes_{R^s} R \to R$	$1 \mid 1 \mapsto 1$
	$R \to R \otimes_{R^s} R$	$1 \mapsto \frac{1}{2}(1 \mid \alpha_s + \alpha_s \mid 1)$
	$R \otimes_{R^s} R \to R \otimes_{R^s} R \otimes_{R^s} R$	$1 \mid 1 \mapsto 1 \mid 1 \mid 1$
	$R \otimes_{R^s} R \otimes_{R^s} R \to R \otimes_{R^s} R$	$1 \mid g \mid 1 \mapsto \partial_s(g) \mid 1$
	$R \otimes_{R^s} R \to R \otimes_{R^s} R$	$1 \mid 1 \mapsto 1 \mid 1$

Table 1: Describing the maps.

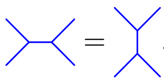
2.3 Operations for Diagrammatics

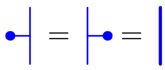
From the algebraic context given above, one can derive the following relationships, which are here described completely graphically. These rules are sufficient to compute the maps we are interested in.

Unless multiple colors appear in an equation, blue represents a generic color.

Operation 0 (Isotropy)**.** If two diagrams can be continuously deformed into each other (in the topological sense), then they are equivalent. In other words, the maps are isotropy invariant.

Remark. The justification for the above operation is not obvious; see page 7 of [5] for an explanation.

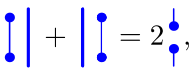
Operation 1 (Associativity)**.** We have  .

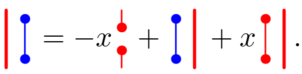
Operation 2 (Contraction)**.** We have  .

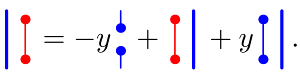
Operation 3 (The Needle)**.** We have  $= 0$.

Remark. Using contraction and the above, one can show that the diagram  is zero as well. In fact, in general, *any* map which contains a "closed" and empty region is the zero map.

Operation 4 (Barbell-Forcing Rules)**.** We have the following three equalities:

(a) , and the similar equation for red.

(b) .

(c) .

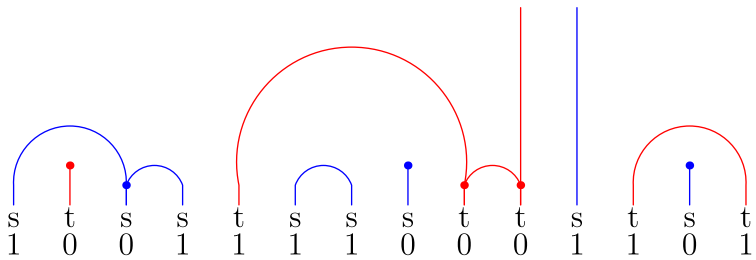


Figure 2: An example of a light leaf.

2.4 Libedinsky's Light Leaves

We now present the algorithm for Libedinsky's light leaves, in the special case where W is the infinite dihedral group. A full construction for more general Coxeter groups is given in [5].

Fix an expression $\underline{r}$ of length n. Then for any binary string $\underline{b}$ of length n we construct a diagram based on a series of rules.

Each bit of $\underline{b}$ is read from left to right in order and designated either open or closed; the state of a bit can change more than once. Initially all bits are closed. We say that a bit $\underline{b}[j]$ is an *open neighbor* of $\underline{b}[k]$ if

(i) $\underline{b}[j]$ is marked open

(ii) $\underline{b}[j]$ and $\underline{b}[k]$ have the same color, and

(iii) there are no other bits marked open between $\underline{b}[j]$ and $\underline{b}[k]$.

Note that each bit will have at most one open neighbor.

Now for $k = 1, 2, \ldots, n$, in that order,

- If the current bit $\underline{b}[k]$ is 1 and has an open neighbor $\underline{b}[j]$, then draw an arc joining the two bits, and mark both bits as closed. Otherwise, designate $\underline{b}[k]$ as open.

- If the current bit $\underline{b}[k]$ is 0 and has an open neighbor $\underline{b}[j]$, then draw an arc joining the two bits, mark $\underline{b}[j]$ as closed, and $\underline{b}[k]$ as open. Otherwise, create a "stub", i.e. a single vertex of degree one, and mark $\underline{b}[k]$ as down.

Finally, join any bits which are open at the end of the process to the upper boundary of the diagram. A map which arises in such a manner is called a *light leaf.* An example of such a light leaf is shown in Figure 2.

The *top* of such a map is the expression which appears on the upper boundary; i.e. the bits which are still up at the end of the algorithm. For example, the map in Figure 2 has top ts. Note that any top is a reduced expression.

3 Preliminaries and Definitions

In Section 2, we provided the background behind the Bott-Samelson bimodules, and the correspondence between our maps and diagrams. We then introduced Libedinsky's light leaves, a basis by which maps can be constructed using binary strings.

In Section 3.1 we introduce several of our definitions which we use throughout the paper. Next, in Section 3.2, we describe how these maps can be represented as polynomials. The actual problem statement is then deferred to Section 4.

3.1 Definitions and Notations

Take an infinite dihedral group $W = \langle s, t \mid s^2 = t^2 = 1 \rangle$ and let $R = \mathbb{R}[\alpha_s, \alpha_t]$ be the Coxeter ring as defined in Section 2, where α_s denotes reflection over the hyperplane normal to s.

3.1.1 Notations for Strings

For each nonnegative integer n, let us define for convenience the strings

$$\mathcal{S}_n \stackrel{\text{def}}{=} \underbrace{s \dots s}_{n\ s\text{'s}}$$

and the "alternating" string

$$\mathcal{A}_n \stackrel{\text{def}}{=} \begin{cases} \underbrace{sts \dots s}_{n \text{ letters}} & \text{if } n \equiv 1 \pmod 2 \\ \underbrace{sts \dots t}_{n \text{ letters}} & \text{if } n \equiv 0 \pmod 2. \end{cases}$$

Similarly, let $\mathcal{A}'_n$ denote the string obtained by switching s with t in $\mathcal{A}_n$. We take $\mathcal{S}_0 = \mathcal{A}_0 = \mathcal{A}'_0 = \varnothing$ where $\varnothing$ denotes the empty string.

In general, a string of s's and t's will be referred to as an *expression*. In the case where there are no consecutive identical characters (in other words, the string is either $\mathcal{A}_n$ or $\mathcal{A}'_n$), we refer to them as *reduced expressions*.

Next, let $\mathfrak{B}_n$ denote the set of binary strings of length n.

Finally, given a string $\underline{x}$ (either a binary string or an expression), let $\underline{x}[i]$ denote the i^{th} character. In general, underlined Roman letters denote strings.

3.1.2 Definitions for Maps and Diagram

Consider an expression $\underline{r}$. Given a binary string $\underline{b}$, let $\mathcal{M}_{\underline{r}}(\underline{b})$ denote the map formed by the light leaves, as described in Section 2.4. We refer to maps and their associated diagrams interchangeably.

Maps constructed by the light leaves are called *half-maps* or *half-diagrams* (since subsequently we compose pairs of them). Figure 2 (in Section 2.4, on page 6) gives an example of such a half-map.

Additionally, let $\operatorname{Top}_{\underline{r}}(\underline{b})$ denote the *top* of the half-diagram $\mathcal{M}_{\underline{r}}(\underline{b})$, as defined in Section 2.4. Note that the top of any half-map is always a reduced expression. In fact, one can verify that the top of a map is given explicitly by

$$\operatorname{Top}_{\underline{r}}(\underline{b}) = \prod_{i=1}^{n} (\underline{r}[i])^{\underline{b}[i]} \in W. \tag{3.1}$$

Two maps $\mathcal{M}_{\underline{r}}(\underline{a})$ and $\mathcal{M}_{\underline{r}}(\underline{b})$ with identical tops may be composed by juxtaposing $\mathcal{M}_{\underline{r}}(\underline{a})$ with a flipped copy of $\mathcal{M}_{\underline{r}}(\underline{b})$; an example of such a composition is given in Figure 3. We denote this product by $\mathcal{M}_{\underline{r}}(\underline{a}, \underline{b})$. If the two half-maps are not compatible (that is, $\operatorname{Top}_{\underline{r}}(\underline{a}) \neq \operatorname{Top}_{\underline{r}}(\underline{b})$), then we simply set $\mathcal{M}_{\underline{r}}(\underline{a}, \underline{b}) = 0$.

Such structures are called *full-maps* or *full-diagrams* for clarity. If $\underline{r}$ is clear from context, we abbreviate $\mathcal{M}_{\underline{r}}$ as simply $\mathcal{M}$ and $\operatorname{Top}_{\underline{r}}$ as Top.

Recall that the light leaves are based off a sequence of vertices at the bottom boundary of a half-map. We call these vertices, whether on the bottom boundary of a half-map or the center of a full-map, the *anchors* for that map, and the associated sequence of letters the *base*.

An example of a full-map with base $\mathcal{A}_5$ is given in Figure 4.

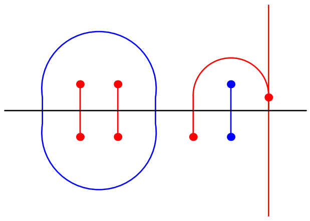


Figure 3: An example of composing two maps, $\mathcal{M}_{sttstst}(1001100)$ and $\mathcal{M}_{sttstst}(1001001)$.

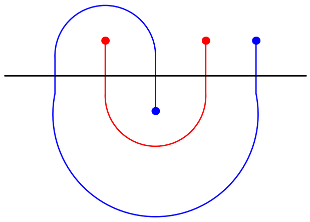


Figure 4: A full-map with base $ststs$, hence with five anchors. The top is $\varnothing$.

3.1.3 Nomenclature for Certain Structures in Diagrams

Let us make a few convenient definitions for some recurring characters in our full-diagrams.

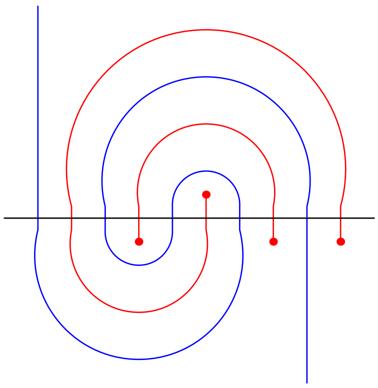


Figure 5: Two red barbells with a very twisted fence, which creates two pastures.

Definition 3.1. A connected component which (i) is acyclic, and (ii) does not touch the top or bottom boundaries is called a *barbell.*

Notice that, by combining homotopy and contraction (see Section 2.3), every blue barbell is simply . Similarly, every red barbell is simply .

Definition 3.2. A *fence* is a contiguous path which runs from the top of the boundary to the bottom of the boundary (i.e. paths between the labelled vertices). The diagram is divided by these fences into *pastures.*

Definition 3.3. A component is called *attached* if it is connected to a fence. Otherwise, it is called *free.*

An example of a fence complete with two barbells is given in Figure 5.

Next, we name some recurrent characters in our work.

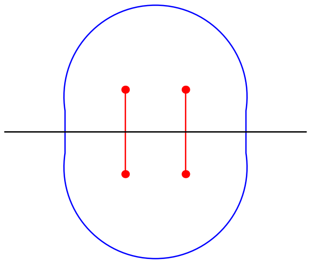


Figure 6: A blue bubble with two barbells inside it.

Definition 3.4. A *bubble* is a bounded face along with any components contained within the face. The *color* of the bubble is the color of the edges which form the bounded face.

Definition 3.5. We say that the bubble B *holds* a component c if c is inside B, but there is no other bubble B' inside B which contains c.

An example of a bubble is given in Figure 6.

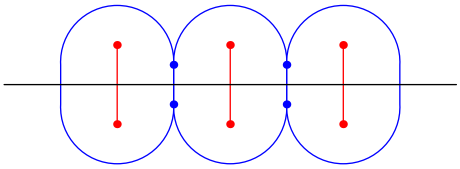


Figure 7: A caterpillar made of three bubbles.

Definition 3.6. A *caterpillar* consists of a free component and the contents of any bubbles formed by its edges. A caterpillar *holds* a component if some bubble of the caterpillar holds it.

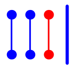
An example of a caterpillar is given in Figure 7. Notice that any barbell or bubble is a special case of a caterpillar.

3.2 Representations of Maps as Polynomials

Recall that every diagram represents a map using the generators and relations in Section 2.2 and Section 2.3.

In order to study these maps as algebraic structures, we wish to consider full-maps as polynomials in $\mathbb{R}[x,y][\alpha_s,\alpha_t]$. Hence, we establish the following conventions:

1. For any $f \in R$, the map $x \mapsto fx$ (i.e. multiplication on the left) will be abbreviated as f from this point forward.

   In particular, the map  is α_s. This follows from a straightforward computation. Furthermore, one can check that, for example,  $= \alpha_s^2\alpha_t$. Note that $x \mapsto xf$ does not get a similar abbreviation. So,  $\neq \alpha_s$.

2. We work *modulo lower terms*. That is, any map which contains a connected component that touches any boundary only once (total) is considered zero. So, for example, we have  $= 0$.

Henceforth, we refer to diagrams and their associated polynomials interchangeably.

One can verify that, using the barbell-forcing rules, we can recursively move all barbells to the left, and any "broken" walls are reduced to zero, or are contracted. Hence, any full-diagram can be reduced to linear combinations of maps of the form $f \mapsto \alpha_s^m \alpha_t^n f$. In other words, every full-diagram can be represented as a polynomial in $\mathbb{R}[x, y][\alpha_s, \alpha_t]$.

In fact, we can even show, with this new convention, that the following identity holds for every f:

$$\left| \, f = \partial_s(f) \quad + s(f) \, \right| \tag{3.2}$$

where ∂_s is the Demazure operator defined in Section 2.3.

For reference, Appendix A contains a complete table of all maps of the form $\mathcal{M}_{\mathcal{A}_5}(\underline{a}, \underline{b})$.

3.2.1 Quantum Numbers

A particularly nice sequence of polynomials can be defined recursively. These polynomials are termed *two-color quantum numbers*.

Definition 3.7 (Quantum Numbers)**.** For each integer $n \geq 0$, define two polynomials $[n]_x, [n]_y \in \mathbb{Z}[x, y]$ recursively by $[0]_x = [0]_y = 0$, $[1]_x = [1]_y = 1$, $[2]_x = x$, $[2]_y = y$, and

$$[2]_x[n]_y = [n+1]_x - [n-1]_x$$
$$[2]_y[n]_x = [n+1]_y - [n-1]_y.$$

Finally, $[-n]_x = -[n]_x$ and $[-n]_y = -[n]_y$ for each $n \geq 0$.

As an example, $[3]_x = [2]_x[2]_y + [1]_x = xy + 1$.

The quantities $[n]_x$ and $[n]_y$ are closely related. In particular, $[2k+1]_x = [2k+1]_y$ and $[2]_x[2k]_y = [2]_y[2k]_x$ for all integers k. This follows immediately by induction.

Recall that the generators s and t of W act on R by the action given in Section 2. The quantum numbers are relevant to the problem because of the identity

$$s([n-1]_x\alpha_s + [n]_y\alpha_t) = [n]_x\alpha_s + [n+1]_y\alpha_t \tag{3.3}$$

and its analogous form for t. One can verify that (3.3) immediately follows from definitions. This gives us a robust way to express the results of action on α_s or α_t by the elements of W. For example, $ststst(\alpha_s) = ststst([1]_x\alpha_s + [0]_y\alpha_t) = [7]_x\alpha_s + [6]_y\alpha_t$.

Now (3.3) motivates us to make the following definition, which we use in Section 8.

Definition 3.8. For any reduced expression $\underline{r}$, we define the *quantum polynomial* $Q_{\underline{r}}$ by $Q_{\mathcal{A}_n} = [n+1]_x\alpha_s + [n]_y\alpha_t$ when $\underline{r} = \mathcal{A}_n$, and otherwise, $Q_{\mathcal{A}'_n} = [n]_x\alpha_s + [n+1]_y\alpha_t$.

The significance is that, if $\underline{r}$ is a reduced expression of length $n+1$, then

$$\mathcal{M}_{\underline{r}}(\underbrace{11\ldots11}_{n\text{ 1's}}0, \underbrace{11\ldots11}_{n\text{ 1's}}0) = Q_{\underline{r}}.$$

An illustration of such a full-map is given in Figure 8. This follows from (3.2).

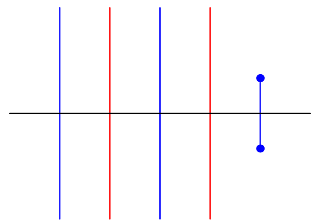


Figure 8: $\mathcal{M}(11110, 11110)$.

4 Problem Statement

Definition 4.1. A map is *idempotent* if multiplying it with itself returns the original map. Two idempotents are *orthogonal* if their product is zero.

If we can find two half-maps A and B for which AB is the identity, then $(BA)^2 = BABA = BA$ is an idempotent. More generally, however, if $AB = c$ for some scalar $c \neq 0$, then

$$\left(\frac{BA}{c}\right)\left(\frac{BA}{c}\right) = \frac{BABA}{c^2} = \frac{BA}{c}$$

is also an idempotent.

However, if we are interested in finding sets of *pairwise orthogonal* idempotents, then this becomes a linear algebra problem in which we require a matrix of all products. The construction of such a set of idempotents is closely related to the decomposition of these bimodules, which in turn gives information about the Soergel bimodules[1].

This motivates the main objective of our study.

Question. For a fixed $\underline{r}$ and two binary sequences $\underline{a}$ and $\underline{b}$ with $\mathrm{Top}_{\underline{r}}(\underline{a}) = \mathrm{Top}_{\underline{r}}(\underline{b})$, find a formula for $\mathcal{M}_{\underline{r}}(\underline{a}, \underline{b})$ in terms of $\underline{r}$, $\underline{a}$, and $\underline{b}$.

In this paper, we wish to compute the polynomial which corresponds to the map.

5 A Complete Characterization of the One-Color Case

In this section we provide a complete characterization for $\mathcal{M}_{\mathcal{S}_n}(\underline{a}, \underline{b})$.

First, we need a criteria to determine the top of such compositions.

Definition 5.1. For a binary string $\underline{b}$, let $\pi_1(\underline{b})$ denote the number of 1's in $\underline{b}$.

Proposition 5.2. *For any $\underline{b} \in \mathfrak{B}_n$,*

$$\mathrm{Top}_{\mathcal{S}_n}(\underline{b}) = \begin{cases} \varnothing & \text{if } \pi_1(\underline{b}) \equiv 0 \pmod 2 \\ s & \text{if } \pi_1(\underline{b}) \equiv 1 \pmod 2. \end{cases}$$

Proof. This is a direct application of (3.1). $\square$

Subsequently, we make the following definition for this section only.

[1] Actually, the Soergel bimodules may be defined as the images of idempotents in the endomorphism rings of the Bott-Samelson bimodules.

Definition 5.3. For a binary string $\underline{b}$, define the *partial-sum string* of $\underline{b}$, denoted $\underline{b}^*$, as follows:

$$\underline{b}^*[i] = \begin{cases} 1 & \text{if } \underline{b}[1] + \underline{b}[2] + \cdots + \underline{b}[i] \equiv 1 \pmod 2 \\ 0 & \text{otherwise.} \end{cases}$$

We are now ready to state our main result for this section.

Theorem 5.4 (One-Color Theorem)**.** *If $\underline{a}, \underline{b} \in \mathfrak{B}_n$ satisfy $\mathrm{Top}_{\mathcal{S}_n}(\underline{a}) = \mathrm{Top}_{\mathcal{S}_n}(\underline{b})$, then*

$$\mathcal{M}_{\mathcal{S}_n}(\underline{a}, \underline{b}) = \begin{cases} 0 & \text{if there exists } 1 \le i \le n-1 : \ \underline{a}^*[i] = \underline{b}^*[i] = 1 \\ \alpha_s^{n - \pi_1(\underline{a}^*) - \pi_1(\underline{b}^*) + \underline{a}^*[n]} & \text{otherwise.} \end{cases}$$

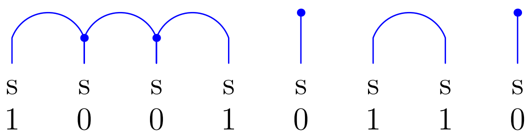


Figure 9: A half-map with base $\mathcal{S}_8$.

Proof of Theorem 5.4. We will only consider the case where $\mathrm{Top}(\underline{a}) = \varnothing$, so that $\underline{a}^*[n] = \underline{b}^*[n] = 0$; the other case is essentially identical. See Figure 9 for an example of such a map.

Number the anchors $1, 2, \ldots, n$ from left to right. For a half-map $\mathcal{M}(\underline{b})$, define the *antennae* of a connected component to be the anchors which it touches. Note that the antennae of a component always form a single contiguous sequence of anchors.

The key observation is that for $1 \le i \le n-1$, we have that $\underline{b}^*[i] = 1$ if and only if i and $i+1$ are both antennae of some component in $\mathcal{M}(\underline{b})$. This follows by construction.

Let us now compose $M_a \overset{\text{def}}{=} \mathcal{M}(\underline{a})$ and $M_b \overset{\text{def}}{=} \mathcal{M}(\underline{b})$ to obtain a full-map. Two components (one from M_a and one from M_b) are said to *feel* each other k times if they have k antennae in common. Notice that if any two components feel each other at least twice, then there exist two adjacent common antennae, forcing the composition of the entire map to be zero because the two antennae combine to form an empty bubble. Furthermore, it is evident that any bubbles must be formed in this manner. In combination with the claim above, this implies the first case of the theorem.

On the other hand, suppose this product is nonzero, and consider a component from the top with k antennae. It suffices to show that there are $n - \pi_1(\underline{a}^*) - \pi_1(\underline{b}^*)$ barbells. Notice that $\pi_1(\underline{a}^*) + \pi_1(\underline{b}^*)$ counts the number of edges between adjacent anchors (i and $i+1$). Because there are no cycles, deleting one of these edges increases the number of connected components by exactly one (since all the connected components are barbells). If we delete all such edges, then we obtain n barbells. Therefore there must have been $n - \pi_1(\underline{a}^*) - \pi_1(\underline{b}^*)$ barbells to begin with, as desired. $\square$

6 Counting Maps

We give a formula for the number of binary strings $\underline{b}$ such that $\mathrm{Top}_{\underline{r}}(\underline{b}) = \underline{w}$ for each expression $\underline{w}$ and $\underline{r}$.

Definition 6.1. Let $\mathrm{Reduce}(\underline{r})$ denote $\underline{r}$ with any consecutive blocks of s's replaced by a single s, and similarly for t.

For example, $\mathrm{Reduce}(sststttts) = ststs$.

Proposition 6.2. *Let $\underline{r}$ be an expression of length ℓ, and suppose* $\mathrm{Reduce}(\underline{r}) = \mathcal{A}_n$.

(i) If $n = 2k+1$ is odd, then for each $0 \le m \le k$, there are $2^{\ell-n}\binom{n-1}{m+k}$ binary strings $\underline{b}$ yielding each of $\mathcal{A}_{2m+1}$, $\mathcal{A}_{2m}$, $\mathcal{A}'_{2m-1}$, and $\mathcal{A}'_{2m}$ as $\mathrm{Top}_{\underline{r}}(\underline{b})$.

(ii) If $n = 2k$ is even, then for each $0 \le m \le k-1$, there are $2^{\ell-n}\binom{n-1}{m+k}$ binary strings $\underline{b}$ yielding each of $\mathcal{A}_{2m+1}$, $\mathcal{A}_{2m+2}$, $\mathcal{A}'_{2m}$, and $\mathcal{A}'_{2m+1}$ as $\mathrm{Top}_{\underline{r}}(\underline{b})$.

If $\underline{r}$ begins with the character t, then one can simply reverse the roles of s and t in the above theorem to obtain an analogous formula.

Proof of Proposition 6.2. The proof proceeds in two steps. The first is to resolve the case $\ell = n$. We do so by using induction on n, and applying (3.1). By splitting into the case where $\underline{b}[n] = 1$ and $\underline{b}[n] = 0$, we get two binomial coefficients; the computation is to verify that we can then apply Pascal's identity. The second step is to reduce the case where $\ell \neq n$ to the previous step using Proposition 5.2.

First we resolve the case where $\underline{r} = \mathcal{A}_n$, so that $\ell = n$ and $2^{\ell-n} = 1$. We proceed by induction on $n \ge 1$ and consider eight cases. (The base case is trivial.) The strategy for each is that, for the binary strings of length n and top $\underline{w}$, we consider those binary strings which end with 1 and those which end with 0. Using (3.1), we may then write the new quantity in terms of the number of strings of length $n-1$ with a different top.

When $n = 2k+1$, we have $\mathcal{A}_n[n] = s$ and $n - 1 = 2k$. Let $\underline{b}'$ denote the first $n-1$ characters of $\underline{b}$. Then,

- The top $\mathcal{A}_{2m+1}$ ends in s. So, if $\underline{b}[n] = 1$ then we have top $\mathcal{A}_{2m} = \mathcal{A}_{2(m-1)+2}$ for $\underline{b}'$, and otherwise $\underline{b}'$ has the same top $\mathcal{A}_{2m+1}$. In total, we get $\binom{n-2}{(m-1)+k} + \binom{n-2}{m+k} = \binom{n-1}{m+k}$.
- The top $\mathcal{A}_{2m}$ ends in t. So, if $\underline{b}[n] = 1$ then we have top $\mathcal{A}_{2m+1}$ for $\underline{b}'$, and otherwise $\underline{b}'$ has the same top $\mathcal{A}_{2m} = \mathcal{A}_{2(m-1)+2}$. In total, we get $\binom{n-2}{m+k} + \binom{n-2}{(m-1)+k} = \binom{n-1}{m+k}$.
- The top $\mathcal{A}'_{2m-1}$ ends in t. So, if $\underline{b}[n] = 1$ then we have top $\mathcal{A}'_{2m}$ for $\underline{b}'$, and otherwise $\underline{b}'$ has the same top $\mathcal{A}'_{2m-1} = \mathcal{A}'_{2(m-1)+1}$. In total, we get $\binom{n-2}{m+k} + \binom{n-2}{(m-1)+k} = \binom{n-1}{m+k}$.
- The top $\mathcal{A}'_{2m}$ ends in s. So, if $\underline{b}[n] = 1$ then we have top $\mathcal{A}'_{2m-1} = \mathcal{A}'_{2(m-1)+1}$ for $\underline{b}'$, and otherwise $\underline{b}'$ has the same top $\mathcal{A}'_{2m}$. In total, we get $\binom{n-2}{(m-1)+k} + \binom{n-2}{m+k} = \binom{n-1}{m+k}$.

The case where $n = 2k$ is similar, but we now have $n - 1 = 2(k-1) + 1$. Now the last bit changes the top by t.

- The top $\mathcal{A}_{2m+1}$ ends in s. So, if $\underline{b}[n] = 1$ then we have top $\mathcal{A}_{2m+2} = \mathcal{A}_{2(m+1)}$ for $\underline{b}'$, and otherwise $\underline{b}'$ has the same top $\mathcal{A}_{2m+1}$. In total, we get $\binom{n-2}{(m+1)+(k-1)} + \binom{n-2}{m+(k-1)} = \binom{n-1}{m+k}$.
- The top $\mathcal{A}_{2m+2}$ ends in t. So, if $\underline{b}[n] = 1$ then we have top $\mathcal{A}_{2m+1}$ for $\underline{b}'$, and otherwise $\underline{b}'$ has the same top $\mathcal{A}_{2m+2} = \mathcal{A}_{2(m+1)}$. In total, we get $\binom{n-2}{m+(k-1)} + \binom{n-2}{(m+1)+(k-1)} = \binom{n-1}{m+k}$.
- The top $\mathcal{A}'_{2m}$ ends in s. So, if $\underline{b}[n] = 1$ then we have top $\mathcal{A}'_{2m+1} = \mathcal{A}'_{2(m+1)-1}$ for $\underline{b}'$, and otherwise $\underline{b}'$ has the same top $\mathcal{A}'_{2m}$. In total, we get $\binom{n-2}{(m+1)+(k-1)} + \binom{n-2}{m+(k-1)} = \binom{n-1}{m+k}$.
- The top $\mathcal{A}'_{2m+1}$ ends in t. So, if $\underline{b}[n] = 1$ then we have top $\mathcal{A}'_{2m}$ for $\underline{b}'$, and otherwise $\underline{b}'$ has the same top $\mathcal{A}'_{2m+1} = \mathcal{A}'_{2(m+1)-1}$. In total, we get $\binom{n-2}{m+(k-1)} + \binom{n-2}{(m+1)+(k-1)} = \binom{n-1}{m+k}$.

This completes the inductive step. Hence, the proposition is true for $\ell = n$.

To finish the case $\ell \neq n$, we note that we can treat each consecutive block of s's of length m as a single s, where there are 2^{m-1} ways to get a 1 and 2^{m-1} ways to get a 0. (This follows from Proposition 5.2.) This reduces to the previous case $\underline{r} = \mathcal{A}_n$ multiplied by additional factors of 2; aggregating all of them yields the factor of $2^{\ell-n}$, as desired. $\square$

7 Colorful Lemmata for the Alternating Case

We now turn our attention to the case of $\underline{r} = \mathcal{A}_n$.

Below we present three results which show that the possible outputs of a map with base $\mathcal{A}_n$ all take certain forms.

Lemma 7.1 (Bubble Lemma). *Consider a bubble in a full-diagram with base $\mathcal{A}_n$. The bubble holds exactly one caterpillar, whose color is opposite that of the bubble.*

Lemma 7.2 (Caterpillar Lemma). *Any caterpillar in a full-diagram with base $\mathcal{A}_n$ evaluates in the form $x^m y^n \alpha_s$ if it is blue, or $x^m y^n \alpha_t$ if it is red.*

Lemma 7.3 (Pasture Lemma). *Consider a full-diagram with $N+1$ pastures (numbered 0, 1, ..., N) with base $\mathcal{A}_n$ and top $\underline{w}$. Suppose further that $N \geq 1$ (or equivalently, $\underline{w} \neq \varnothing$). Then*

(i) Pastures 1 through $N-1$ contain no caterpillars.

(ii) Pasture N is empty if $\underline{w}[N] = \mathcal{A}_n[n]$, and contains a single caterpillar otherwise. The caterpillar is blue if $\mathcal{A}_n[n]$ is s, and red otherwise.

Included in Figure 10 is an example of a map with base $\mathcal{A}_{27}$. The Bubble Lemma and Pasture Lemma can be readily observed in this example.

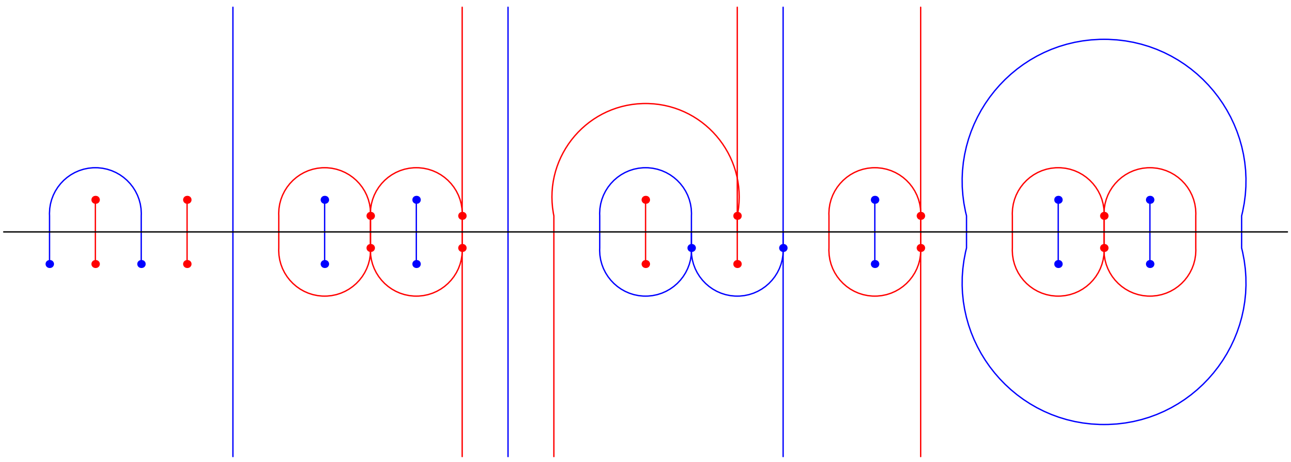


Figure 10: A large map with base $\mathcal{A}_{27}$.

The Bubble Lemma is the key combinatorial observation, which hinges on the fact that any two anchors with the same color have a third anchor of the other color between them. The other two lemmas follow directly from it.

7.1 Proof of the Bubble Lemma

We begin with a proof of the Bubble Lemma[2], which is the fundamental lemma for the alternating case. Indeed, both of the subsequent lemmas are fairly straightforward consequences of this one.

Proof of Lemma 7.1. Take a bubble B and assume without loss of generality that it is blue. By construction all caterpillars held inside B must be red. This implies that there must be at least one red caterpillar, because there is at least one red anchor inside B.

Define a pair of red anchors u, v, with u to the left of v, to be *quasi-connected* if (i) one can draw an arc from u to v contained entirely in one half-map which does not intersect blue lines, and (ii) there is no red anchor w strictly between u and v for which one can draw a similar arc on the same half-map joining w and v.

Now we claim that **any quasi-connected pair of red anchors is actually joined by a red arc**. To prove this, take such a quasi-connected pair u and v. Now we will focus only on the half-map in which the prescribed arc is drawn. We thus may state that an arc *shields* any anchors which lie between its endpoints.

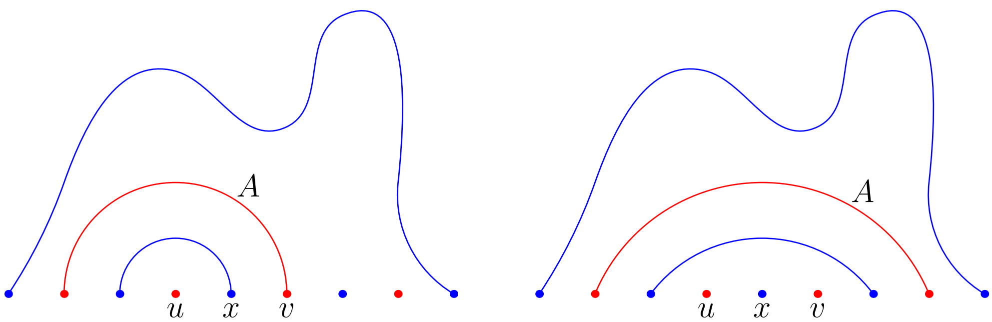


Figure 11: Proving the bubble lemma with quasi-connected vertices.

Because the string is alternating, we can take a blue anchor x immediately to the left of v. Now, there must be some blue arc shielding u and v (since u, v are inside a blue bubble). By construction, there must be some red arc A covering x. If this arc joins u and v, then we are done. Otherwise, if it has endpoint v then it must completely cover u by condition (ii) of quasi-connectedness; if not, then it must completely cover v. This leads us to two cases.

- If the arc covers exactly one of u or v, say u, then there must be a blue arc shielding u from A. But then this contradicts the quasi-connectedness, because now it is not possible to join u and v by an arc on this half-map. This case is illustrated by the left half of Figure 11.

- If the arc covers both u and v, then there must be a blue arc which shields both u and v, which is itself shielded by A. Then we can repeat the same procedure with this new blue arc. This case is illustrated by the right half of Figure 11.

Therefore any pair of quasi-connected anchors must be connected by an arc. However, one can verify that the red anchors in B but not in any other bubble are "connected" under quasi-

[2]We should remark that attempting to construct counterexamples to the Bubble Lemma leads to a far better understanding than trying to read the following proof.

connectedness.[3] Therefore all such red anchors must be connected. So there is a single caterpillar, as desired. □

7.2 Proof of the Caterpillar Lemma

We will now derive the Caterpillar Lemma as a consequence of the Bubble Lemma.

It is worth intuitively explaining the main idea of the proof; the rest is book-keeping. Essentially, because caterpillars may only hold a single other caterpillar, we may "descend down" until we find a caterpillar which contains barbells in its bubbles. Now we break each of the bubbles of this caterpillar one at a time using the barbell forcing rule. This generates several x and y terms, but the barbell disappears at each such move. Hence, we break all cycles in the caterpillar, at which point it becomes a barbell (modulo the addition of new x and y terms). Rinse and repeat.

First, a few definitions for this section only.

Definition 7.4. Consider a caterpillar C in a full-diagram with base $\mathcal{A}_n$. Define the *nesting level* of the caterpillar as the largest integer n for which there exists caterpillars $C = C_0$, C_1, C_2, $\dots$, C_n such that C_i holds C_{i+1} for each $i = 0, 1, \dots, n-1$. Define the *size* of the caterpillar to be the number of bubbles it contains.

Proof of Lemma 7.2. The proof is by double induction. First we will consider caterpillars with nesting level 0.

We perform induction on the size of the caterpillar. If there are no bubbles, then the caterpillar is simply a barbell. This is the base step. For the inductive step, consider an arbitrary bubble. Inside it is exactly one caterpillar, say c, by the Bubble Lemma. Because the original caterpillar has nesting level 0, we deduce c is merely a barbell, and hence is either α_s or α_t. Applying the barbell-forcing rule of equation (3.2) generates two terms. One of them is zero because it creates a new empty bubble. The other is the original diagram with the cycle cut open and the contents removed, multiplied by one of $\{x, y\}$. The inductive hypothesis now applies directly, and this completes the first induction.

Now suppose a caterpillar has nesting level $k \geq 1$. We simply apply the exact same induction on size as before. The only modification is that, instead of a barbell, we have a caterpillar with nesting level strictly less than k. However, these caterpillars all evaluate as a polynomial again of the form $x^m y^n \alpha_s$ or $x^m y^n \alpha_t$, by the outer inductive hypothesis. So applying equation (3.2) in the same manner completes this induction as well. □

7.3 Proof of the Pasture Lemma

The Bubble Lemma and the Pasture Lemma are structurally very similar. In fact, we will now show a somewhat surprising proof of the Pasture Lemma via the Bubble Lemma.

Proof of Lemma 7.3. Again, let $\underline{w}$ denote the top, with length N. In the first case, suppose $\underline{w}[N] \neq \mathcal{A}_n[n]$; that is, the last characters of $\underline{w}$ and $\mathcal{A}_n$ differ. Then it must be the case that the rightmost anchor is not part of a fence. Now suppose the map arose from $\mathcal{M}_{\mathcal{A}_n}(\underline{a}, \underline{b})$. Let $\underline{a}'$ be the string formed by appending N 1's to $\underline{a}$, and define $\underline{b}'$ similarly. Now consider the map

$$\mathcal{M}_{\mathcal{A}_{n+N}}\left(\underline{a}', \underline{b}'\right).$$

[3] That is, one can follow a sequence of red anchors, each consecutive pair quasi-connected, from any red anchor to any other red anchor.

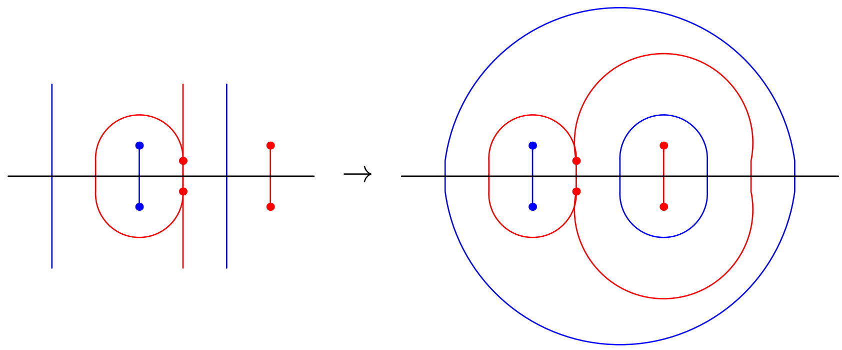


Figure 12: In the first diagram, $\underline{a} = \underline{b} = 110010$. In the second, $\underline{a}' = \underline{b}' = 110010111$. Closing the pastures allows us to deduce the Pasture Lemma from the Bubble Lemma.

See Figure 12 for an example.

Clearly the top of this new map is zero, say by (3.1). Now what we have done is enclose the contents of each pasture into bubbles. The Bubble Lemma tells us that the innermost bubble can have exactly one caterpillar, while the other bubbles must contain only a single caterpillar, which was once a fence. This implies the Pasture Lemma in the first case.

The case where $\underline{w}[N] = \mathcal{A}_n[n]$ is similar. We simply append a zero followed by N 1's and repeat the same procedure. The only difference is that the innermost barbell was generated by the 0 we appended, and thus was not part of the original diagram. $\square$

8 Algebraic Computations for the Alternating Case

We now turn to using the tools developed in Section 7 in order to compute polynomials.

8.1 Restrictions and Definitions for the Final Output

Let us now endow our maps with base $\mathcal{A}_n$ with some additional structure.

Theorem 8.1. *For any binary strings $\underline{a}, \underline{b} \in \mathfrak{B}_n$, if $\mathcal{M}_{\mathcal{A}_n}(\underline{a}, \underline{b})$ is nonzero and has top $\underline{w}$, then for some nonnegative integers m_x, m_y, m_s, and m_t, we have*

$$\mathcal{M}_{\mathcal{A}_n}(\underline{a}, \underline{b}) = x^{m_x} y^{m_y} \alpha_s^{m_s} \alpha_t^{m_t} \cdot \begin{cases} 1 & \textit{if } \underline{w} \textit{ and } \mathcal{A}_n \textit{ have the same last character} \\ Q_{\underline{w}} & \textit{otherwise.} \end{cases}$$

Here, $Q_{\underline{w}}$ is the quantum polynomial defined in Section 3.2.1.

Proof. This is an immediate consequence of the Caterpillar Lemma and the Pasture Lemma. Everything in the first pasture is a product of monomials by the Caterpillar Lemma. If $\underline{w}$ and $\mathcal{A}_n$ agree, then the last pasture is empty and hence we obtain the first case. Otherwise, there is a single caterpillar of color opposite to $\underline{w}$, and hence after evaluating that caterpillar, we obtain some monomial times a single barbell. This generates the final $Q_{\underline{w}}$ term after the barbell is pushed through all the fences; the broken fences are all zero because they are lower terms. $\square$

Theorem 8.1 motivates the following definition, which is relevant for Proposition 8.3.

Definition 8.2. If a nonzero map M with base $\mathcal{A}_n$ is presented as in Theorem 8.1, then we call the term $x^{m_x}y^{m_y}\alpha_s^{m_s}\alpha_t^{m_t}$ the *fluff* of M, and call the polynomial (either 1 or $Q_{\underline{w}}$) the *face* of M.

Remark. According to Theorem 8.1, a nonzero map $\mathcal{M}_{\mathcal{A}_n}$ has face 1 if and only if its top has the same last character as $\mathcal{A}_n$.

8.2 A Partial Recursion

In this section we present a partial recursion for computing $\mathcal{M}_{\mathcal{A}_n}(\underline{a}, \underline{b})$.

Proposition 8.3. *Let $n \geq 2$ be an odd integer and consider two binary strings $\underline{a}, \underline{b} \in \mathfrak{B}_n$ such that $\underline{w} = \mathrm{Top}_{\mathcal{A}_n}(\underline{a}) = \mathrm{Top}_{\mathcal{A}_n}(\underline{b})$. Let $\underline{a}'$ and $\underline{b}'$ be the first $n-1$ bits of $\underline{a}$ and $\underline{b}$, respectively. Suppose $M_1 = \mathcal{M}_{\mathcal{A}_{n-1}}\left(\underline{a}', \underline{b}'\right)$ is a nonzero map with fluff F.*

(i) If $\underline{a}[n] = \underline{b}[n] = 0$, then

$$\mathcal{M}_{\mathcal{A}_n}(\underline{a}, \underline{b}) = \begin{cases} \alpha_s M_1 & \text{if } \underline{w} = \varnothing \\ -xF & \text{if } \underline{w} \text{ ends in } s \\ Q_{\underline{w}} M_1 & \text{if } \underline{w} \text{ ends in } t. \end{cases}$$

(ii) If $\underline{a}[n] = \underline{b}[n] = 1$, then

$$\mathcal{M}_{\mathcal{A}_n}(\underline{a}, \underline{b}) = \begin{cases} -x\alpha_s F & \text{if } \underline{w} = \varnothing \\ M_1 & \text{if } \underline{w} \text{ ends in } s \\ -xFQ_{\underline{w}} & \text{if } \underline{w} \text{ ends in } t. \end{cases}$$

Analogous statements hold when n is an even integer, with the roles of x and y interchanged, as well as the roles of s and t.

The recursion arises because, using the lemmata of Section 7, we are able to see exactly what happens in the rightmost pasture if a pair of 1's or a pair of 0's is added. In short, either a barbell or a fence is added on the far right, or a caterpillar is enclosed in a bubble; each of these leads to one of the cases above.

Notice that this recursion is sufficient to calculate $\mathcal{M}_{\mathcal{A}_n}(\underline{b}, \underline{b})$.

Proof of Proposition 8.3. We will only resolve the case where n is odd; that is, $\mathcal{A}_n$ terminates in s. The other case is identical.

(i) Suppose $\underline{a}[n] = \underline{b}[n] = 0$. Note that the top of M_1 is also $\underline{w}$.

- If $\underline{w} = \varnothing$, then M_1 also had top $\varnothing$. Therefore, the addition of the 0's at the end simply generates an extra blue barbell. So the resulting map is simply a multiple by α_s; i.e. $M = \alpha_s M_1$.
- The case where $\underline{w}$ ends in t is analogous; we simply obtain a new barbell. However, because $\underline{w} \neq \varnothing$, the extra factor is $Q_{\underline{w}}$ instead of simply α_t. Hence, $M = Q_{\underline{w}} M_1$.
- Now suppose $\underline{w}$ ends in s. Then the addition of the 0's resulted in enclosing a caterpillar of M_1 in an attached bubble, as per Figure 13. Breaking out of the additional blue loop now adds a factor of x, but the red caterpillar is annihilated in the process; therefore, only the fluff remains, because there is nothing to push through the fences. Hence, $M = -xF$.

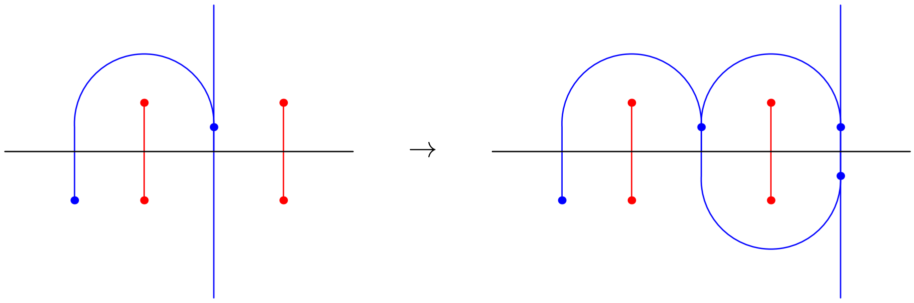


Figure 13: $\mathcal{M}(1000, 0010)$ on the left, and $\mathcal{M}(10000, 00100)$ on right.

(ii) Now suppose that $\underline{a}[n] = \underline{b}[n] = 1$.

- First, suppose $\underline{w}$ ends in s. Then the additional 1's added simply create a fence on the far right, because there are no "up" blue anchors to interact with. Because this fence is already on the far right, it does not affect the product, hence $M = M_1$.

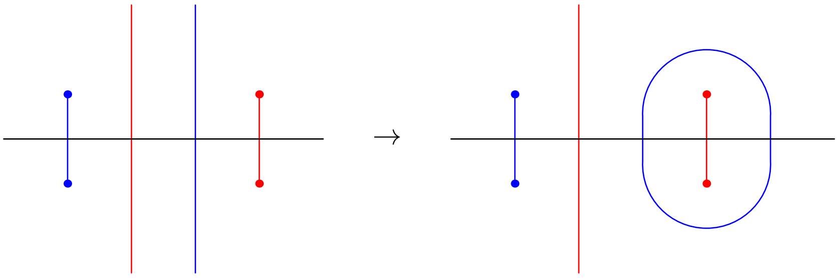


Figure 14: $\mathcal{M}(0110, 0110)$ on the left, and $\mathcal{M}(01101, 01101)$ on right.

- Conversely, suppose that $\underline{w}$ ends in t. See Figure 14 for an example.

  In that case, M_1 must have had the top $\underline{w}$ with an s appended. Once again, we find that the red caterpillar from M_1 is enclosed in a bubble. This time, however, the bubble is not attached. Therefore, while we still obtain the extra factor of x and the caterpillar is still removed, we remain with a blue barbell on the far right pasture. So we must push this blue barbell through $\underline{w}$. This adds a factor of $Q_{\underline{w}}$, and so we obtain the desired form $M = -xFQ_{\underline{w}}$.

- The case where $\underline{w} = \varnothing$ is analogous to the previous case; in a sense, we merely have a degenerate $Q_\varnothing = \alpha_s$. Specifically, the blue barbell is already on the correct pasture, so we simply add a factor of α_s. Therefore, $M = -xF\alpha_s$.

This exhausts all the cases in the proposition, hence we are done. □

9 Conclusion

The overarching goal of the paper was to compute $\mathcal{M}_{\underline{r}}(\underline{a}, \underline{b})$ in terms of $\underline{r}$, $\underline{a}$, and $\underline{b}$. In this paper, we solved the case where $\underline{r} = \mathcal{S}_n$ completely, and showed that outputs in the case $\underline{r} = \mathcal{A}_n$ are substantially constrained. We also provided a partial recurrence for $\mathcal{A}_n$. An obvious next direction would be to attempt to finish the recursion for $\mathcal{A}_n$ so that $\mathcal{M}_{\mathcal{A}_n}(\underline{a}, \underline{b})$ can be determined completely.

What about the original problem of arbitrary $\underline{r}$? It was initially hoped that, by understanding $\mathcal{A}_n$ and $\mathcal{S}_n$, one could derive the general formula (perhaps by viewing strings of s's in $\underline{r}$ as a single s with a formula for $\mathcal{S}_n$, and then aggregating all this information with a formula for $\mathcal{A}_n$). Unfortunately, the lemmata of Section 7 show that in fact, maps with base $\mathcal{A}_n$ have very special properties. So, either the results of Section 8 will have to be generalized significantly (in the context of maps outside those with base $\mathcal{A}_n$), or an entirely new approach will be required.

Yet another possible direction is to search for similar results when the order of st in W is finite, or even if W is an arbitrary Coxeter group.

10 Acknowledgments

The authors would like to thank Professor Benjamin Elias, MIT, for providing this project and for often meeting personally to discuss it.

Furthermore, we would like to thank the RSI head mentor Dr. Tanya Khovanova, MIT, for providing individual discussion with the first author, and the MIT Math Department for making the project possible. We also wish to thank Mr. Antoni Rangachev, Northeastern University, for his invaluable advice in preparing this paper and the associated presentation, as well as Felipe Hernández and Charlie Pasternak for their revisions.

The first author would also like to thank the following RSI sponsors: Ms. Alexa Margalith, The Leonetti/O'Connell Family Foundation; Mr. Arvind Parthasarathi, YarcData; Mr. Samuel Chen and Ms. Kathy Chang; Dr. Robert E. Curry; Mr. and Mrs. William Hellman; Mr. Ronald Houhauser; Ms. Chienlan Hsu-Hoffman; and Mr. and Mrs. George Keiter.

Finally, we also thank the Center for Excellence in Education and the Research Science Institute, as well as the Massachusetts Institute of Technology, for generously providing the facilities for the research.

References

[1] D. Kazhdan and G. Lusztig. Representations of Coxeter groups and Hecke algebras. *Invent. Math.*, 53(2):165–184, 1979.

[2] D. Kazhdan and G. Lusztig. Schubert varieties and Poincaré duality. In *Geometry of the Laplace operator (Univ. Hawaii, 1979)*, Proc. Sympos. Pure Math., XXXVI, pages 185–203. Amer. Math. Soc., Providence, R.I., 1980.

[3] W. Soergel. Kategorie $\mathcal{O}$, perverse Garben und Moduln über den Koinvarianten zur Weylgruppe. *J. Amer. Math. Soc.*, 3(2):421–445, 1990.

[4] W. Soergel. Kazhdan-Lusztig-Polynome und unzerlegbare Bimoduln über Polynomringen. *J. Inst. Math. Jussieu*, 6(3):501–525, 2007.

[5] B. Elias and G. Williamson. Generators and relations for Soergel bimodules. In preparation.

[6] B. Elias and M. Khovanov. Diagrammatics for Soergel categories, 2009. arXiv:0902.4700v1.

[7] N. Libedinsky. Sur la catégorie des bimodules de Soergel. *J. Algebra*, 320(7):2675–2694, 2008.

A Complete Product Matrix for ststs

For reference, here is a complete product matrix for maps with base $ststs$. The blank entries are 0 maps. The top is recorded as the title of each table.

stst	11110
11110	$[5]\alpha_s + [4]_y\alpha_t$

ststs	11111
11111	1

tst	01110
01110	$[3]\alpha_s^2 + [4]_y\alpha_s\alpha_t$

tsts	01111
01111	α_s

ts	01100	01001	00011	10111
01100	$-[2]_x\alpha_s$	α_s		
01001	α_s	$-[2]_y\alpha_s$	α_s^2	α_s
00011		α_s^2	$\alpha_s^2\alpha_t$	$\alpha_s\alpha_t$
10111		α_s	$\alpha_s\alpha_t$	$-[2]_x\alpha_s$

st	11000	10010	00110	11101
11000	$-[2]_y[3]\alpha_s - [2]_y[2]_y\alpha_t$	$[3]\alpha_s + [2]_y\alpha_t$		$[3]\alpha_s + [2]_y\alpha_t$
10010	$[3]\alpha_s + [2]_y\alpha_t$	$-[2]_x[3]\alpha_s - [2]_x[2]_y\alpha_t$	$[3]\alpha_s\alpha_t + [2]_y\alpha_t^2$	
00110		$[3]\alpha_s\alpha_t + [2]_y\alpha_t^2$	$[3]\alpha_s^2\alpha_t + [2]_y\alpha_s\alpha_t^2$	
11101	$[3]\alpha_s + [2]_y\alpha_t$			$-[2]_x[3]\alpha_s - [2]_x[2]_y\alpha_t$

t	01000	00010	10110	01101
01000	$-[2]_y\alpha_s^2 - [2]_y[2]_y\alpha_s\alpha_t$	$\alpha_s^3 + [2]_y\alpha_s^2\alpha_t$	$\alpha_s^2 + [2]_y\alpha_s\alpha_t$	$\alpha_s^2 + [2]_y\alpha_s\alpha_t$
00010	$\alpha_s^3 + [2]_y\alpha_s^2\alpha_t$	$\alpha_s^3\alpha_t + [2]_y\alpha_s^2\alpha_t^2$	$\alpha_s^2\alpha_t + [2]_y\alpha_s\alpha_t^2$	
10110	$\alpha_s^2 + [2]_y\alpha_s\alpha_t$	$\alpha_s^2\alpha_t + [2]_y\alpha_s\alpha_t^2$	$-[2]_x\alpha_s^2 - [2]_x[2]_y\alpha_s\alpha_t$	
01101	$\alpha_s^2 + [2]_y\alpha_s\alpha_t$			$-[2]_x\alpha_s^2 - [2]_x[2]_y\alpha_s\alpha_t$

sts	11100	11001	10011	00111
11100	$-[2]_x$	1		
11001	1	$-[2]_y$	1	
10011		1	$-[2]_x$	α_t
00111			α_t	$\alpha_s\alpha_t$

∅	00000	10100	01010	10001	00101	11011
00000	$\alpha_s^3\alpha_t^2$	$\alpha_s^2\alpha_t^2$	$\alpha_s^3\alpha_t$	$\alpha_s\alpha_t^2$	$\alpha_s^2\alpha_t^2$	$\alpha_s^2\alpha_t$
10100	$\alpha_s^2\alpha_t^2$	$-[2]_x\alpha_s^2\alpha_t$	$\alpha_s^2\alpha_t$	$-[2]_x\alpha_s\alpha_t$	$\alpha_s\alpha_t^2$	$\alpha_s\alpha_t$
01010	$\alpha_s^3\alpha_t$	$\alpha_s^2\alpha_t$	$-[2]_y\alpha_s^2\alpha_t$	$\alpha_s\alpha_t$	$\alpha_s^2\alpha_t$	$-[2]_y\alpha_s\alpha_t$
10001	$\alpha_s\alpha_t^2$	$-[2]_x\alpha_s\alpha_t$	$\alpha_s\alpha_t$	$[2]_x[2]_x\alpha_s$	$-[2]_x\alpha_s\alpha_t$	$-[2]_x\alpha_s$
00101	$\alpha_s^2\alpha_t^2$	$\alpha_s\alpha_t^2$	$\alpha_s^2\alpha_t$	$-[2]_x\alpha_s\alpha_t$	$-[2]_x\alpha_s^2\alpha_t$	$\alpha_s\alpha_t$
11011	$\alpha_s^2\alpha_t$	$\alpha_s\alpha_t$	$-[2]_y\alpha_s\alpha_t$	$-[2]_x\alpha_s$	$\alpha_s\alpha_t$	$[2]_x[2]_y\alpha_s$

s	10000	00100	11010	00001	10101	01011
10000	$[2]_x[2]_x$	$-[2]_x\alpha_t$	$-[2]_x$	α_t^2	$-[2]_x\alpha_t$	α_t
00100	$-[2]_x\alpha_t$	$-[2]_x\alpha_s\alpha_t$	α_t	$\alpha_s\alpha_t^2$	α_t^2	$\alpha_s\alpha_t$
11010	$-[2]_x$	α_t	$[2]_x[2]_y$	$\alpha_s\alpha_t$	α_t	$-[2]_y\alpha_t$
00001	α_t^2	$\alpha_s\alpha_t^2$	$\alpha_s\alpha_t$	$\alpha_s^2\alpha_t^2$	$\alpha_s\alpha_t^2$	$\alpha_s^2\alpha_t$
10101	$-[2]_x\alpha_t$	α_t^2	α_t	$\alpha_s\alpha_t^2$	$-[2]_x\alpha_s\alpha_t$	$\alpha_s\alpha_t$
01011	α_t	$\alpha_s\alpha_t$	$-[2]_y\alpha_t$	$\alpha_s^2\alpha_t$	$\alpha_s\alpha_t$	$-[2]_y\alpha_s\alpha_t$

B GitHub Repository

For the project, the first author wrote several scripts to produce and/or compute diagrams. The source code can be checked out here: `https://github.com/vEnhance/lightleaves`.