A Puzzle-Based Synthesis Algorithm For a Triple Intersection of Schubert Varieties

by

Andrew Alexander Harold Brown

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Combinatorics & Optimization

Waterloo, Ontario, Canada, 2010

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

This thesis develops an algorithm for the Schubert calculus of the Grassmanian. Specifically, we state a puzzle-based, synthesis algorithm for a triple intersection of Schubert varieties. Our algorithm is a reformulation of the synthesis algorithm by Bercovici, Collins, Dykema, Li, and Timotin. We replace their combinatorial approach, based on specialized Lebesgue measures, with an approach based on the puzzles of Knutson, Tao and Woodward. The use of puzzles in our algorithm is beneficial for several reasons, foremost among them being the larger body of work exploiting puzzles. To understand the algorithm, we study the necessary Schubert calculus of the Grassmanian to define synthesis. We also discuss the puzzle-based Littlewood-Richardson rule, which connects puzzles to triple intersections of Schubert varieties. We also survey three combinatorial objects related to puzzles in which we include a puzzle-based construction, by King, Tollu, and Toumazet, of the well known Horn inequalities.

Acknowledgements

I would like to thank my advisor Kevin Purbhoo for his guidance and instruction. I also thank my Master's committee for their many useful suggestions. I am very grateful of Mathieu Guay-Paquet for his selfless help and advice with Latex and Postscript. I thank the Department of Combinatorics and Optimization for the academic opportunities afforded me as a graduate student. Lastly, I would like to acknowledge the financial support of the National Science and Engineering Research Council of Canada.

Contents

List of Figures

List of Symbols

$\wedge$	Meet, page 4
$\vee$	Join, page 4
c_{IJK}	A Littlewood-Richardson number, page 3
$\mathbb{C}^n$	Complex n-space, page 1
$\underline{d}$	A step sequence, page 2
$\mathcal{D}_0(P)$	A puzzles type 0 deflation, page 33
$\mathcal{D}_1(P)$	A puzzles type 1 deflation, page 32
$\mathcal{D}^\circ_\rightarrow(P)$	The rhombus mark deflation of P, page 79
$\mathcal{D}_\rightarrow(U)$	Rhombus mark deflation of U, page 61
$e \rightarrow f$	f is descended from e, page 35
EWF	End With Failure, page 72
$F_\bullet$	A flag in $\mathbb{C}^n$, page 3
$\mathcal{F}\ell(E)$	Flag manifold of complete flags, page 2
$\mathcal{F}\ell_{\underline{d}}(E)$	Flag manifold for step sequence $\underline{d}$, page 2
GL_n	The complex general linear group, page 14
$\mathrm{Gr}_d(\mathbb{C}^n)$	The complex Grassmanian, page 2
$H^*(\mathrm{Gr}_d(\mathbb{C}^n))$	The cohomology ring of $\mathrm{Gr}_d(\mathbb{C}^n)$, page 16

I	A cardinality d subset of $[n]$, page 3
I^*	The dual set of $I \in [n]_d$, page 7
I, J, K	Boundary of a puzzle, page 6
$\mathcal{I}_0(m^*)$	A dual measures inflations, page 33
$\mathcal{I}_1(m)$	A measures inflation, page 32
$I^{\max}$	The integers $n-d+1$ to n, page 10
$I^{\min}$	The integers $[d]$, page 10
$I \xrightarrow{m} J$	J is an m-shift of I, page 13
J_m	The unique m-shift of $I^{\max}$, page 13
$K^\vee$	The 'checked' version of $K \in [n]_d$, page 3
λ	A partition of N, page 19
$\lambda(I)$	The partition for $I \in [n]_d$, page 20
λ/ν	A skew partition, page 19
m	A measure in $\mathcal{M}_r$, page 30
$m(e)$	The edge density of e, page 30
$\mathcal{M}_r$	The set of measures in $\triangle_r$, page 31
$\mathcal{M}_s^*$	A set of dual measures, page 33
$[n]$	The integers 1 to n, page 3
$[n]_d$	All cardinality d subsets of $[n]$, page 3
P	A puzzle, page 5
$\mathrm{Part}_{n,d}$	Partitions which fit within an $n-d$ by d box, page 19
S_e	The descendance set of e, page 36
σ_I	A Schubert class, page 16

s_λ	The Schur function for λ, page 20
$s_{(m)}$	The homogeneous, symmetric polynomial of degree m, page 20
$\mathcal{S}(P)$	A stretched Puzzle, page 80
SSYT	Semistandard Young tableau, page 47
$\text{supp}(m)$	The support of a measure m, page 31
t	A semistandard Young tableau, page 47
T	A tri-region, page 55
$\triangle_r$	A size r equilateral triangle, page 29
w	A walker, page 52
$\omega(m)$	The weight of measure m, page 31
$w(t)$	The word of t, page 48
$\Omega_I^\circ(F_\bullet)$	A Schubert cell, page 10
$\Omega_I(F_\bullet)$	A Schubert variety, page 3
T_{IJK}	Short hand for $\Omega_I(F_\bullet) \cap \Omega_J(G_\bullet) \cap \Omega_K(H_\bullet)$, page 3
$\widetilde{\Omega}_J$	An opposite Schubert variety, page 12
$\Omega_{J_m}(G_\bullet)$	Special Schubert variety, page 13
$\Omega_I \cap \widetilde{\Omega}_J$	A Richardson variety, page 12

Chapter 1

Introduction

The main topic of this thesis is the *Schubert calculus* of the Grassmanian, a subdiscipline of enumerative geometry. Schubert calculus is the study of certain counting problems in linear algebra, which take forms similar to the following classical example studied by Hermann Schubert [19]:

> Let $d < n$ be positive integers and $N = d(n-d)$. Given N $(n-d)$-dimensional, linear subspaces of $\mathbb{C}^n$ in general position, how many d-dimensional subspaces intersect all N subspaces non-trivially?

It is natural to phrase these counting problems in terms of *Schubert varieties* (defined in Subsection 1.1.1), where the number of solutions is given as the size of an intersection of Schubert varieties. The number of solutions can also be calculated using a Littlewood-Richardson rule. Each such rule asserts that the size of a finite triple intersection of Schubert varieties is equal to the size of a certain set of combinatorial objects; three examples of Littlewood-Richardson rules are discussed in Chapter 3.

Here, we are interested in more than computing the size of intersections. We also interested in explicitly describing the points in a triple intersection of Schubert varieties in terms of spans and intersections of subspaces used to define the three Schubert varieties. A point described in this way is said to be *synthesized*.

As we will see in Subsection 2.1.2, if a point V in the triple intersection can be synthesized, then V is the unique point of that intersection. In this thesis, we will state an algorithm that establishes the converse: if V is the unique point of a triple intersection, then V can be synthesized.

The algorithm for synthesizing a point V was first published by Bercovici, Collins, Dykema, Li, and Timotin in [2]. They use a combinatorial approach based on specialized Lebesgue measures for which a Littlewood-Richardson rule can be stated. The measures are in bijection with *puzzles*, another important combinatorial object. Puzzles were introduced by Knutson, Tao, and Woodward in [15] to prove a claim pertaining to Horn's conjecture about eigenvalues of Hermitian matrices. The main result of this thesis is a reformulation of the algorithm in [2] using a puzzle-based approach in place of the measures used in [2].

We now introduce our basic notation and state the main definitions concerning Schubert varieties, synthesis, and puzzles.

1.1 Schubert Calculus

This section will define the necessary concepts from the Schubert calculus of the Grassmanian. We are actually restricting to a special case of Schubert calculus. More generally, Schubert calculus is the study of Schubert varieties in flag manifolds and generalizations of those manifolds. As the notation will be applied in later sections, we now define flag manifolds in general.

Let E be an n-dimensional vector space. A **step sequence**, $\underline{d}$, is a strictly increasing list of non-negative integers $\underline{d} = (d_0, d_1, \ldots, d_m)$ with $d_0 = 0$ and $d_m = n$. A **flag** $F_\bullet$ in E with step sequence $\underline{d}$ is a list of subspaces of E, $F_\bullet = (F_0, F_1, \ldots, F_m)$, such that $\dim F_\ell = d_\ell$ and $F_\ell \subset F_{\ell+1}$. The subspace F_ℓ is called the ℓth **step** of the flag, and a **complete flag** has step sequence $\underline{d} = (0, 1, 2, \ldots, n)$.

The **flag manifold** with step sequence $\underline{d}$, $\mathcal{F}\ell_{\underline{d}}(E)$, is the set of flags with step sequence $\underline{d}$, and the flag manifold of complete flags is denoted $\mathcal{F}\ell(E)$. It is beyond our scope to delve into the theory of general flag manifolds other than to note that they are smooth, projective varieties, and hence manifolds. A comprehensive discussion of flag manifolds and their Schubert calculus can be found in [6].

The Schubert calculus we are interested in is carried out in the complex **Grassmanian**, $\mathrm{Gr}_d(\mathbb{C}^n)$, which we noted is a special case of a flag manifold where $E = \mathbb{C}^n$:

$$\mathrm{Gr}_d(\mathbb{C}^n) := \mathcal{F}\ell_{(0,d,n)}(\mathbb{C}^n).$$

The complex Grassmanian can also be viewed as the set of d-dimensional subspaces of $\mathbb{C}^n$; indeed, this was how Hermann Grassman first introduced

$\mathrm{Gr}_d(\mathbb{C}^n)$. Since the first and last steps of every flag are fixed, it is clear that the two views of $\mathrm{Gr}_d(\mathbb{C}^n)$ are equivalent. We now develop the basic definitions from Schubert calculus of the Grassmanian.

1.1.1 Schubert Varieties

The Schubert calculus of $\mathrm{Gr}_d(\mathbb{C}^n)$ involves manipulating the Schubert varieties $\Omega_I(F_\bullet)$, the primary geometric objects of consideration. The input data for $\Omega_I(F_\bullet)$ are a complete flag $F_\bullet$ and a cardinality d set $I \subset [n]$. We will denote the set of all such subsets as $[n]_d$, and if $I \in [n]_d$, we index the elements of I as $I = \{i_1 < i_2 < \cdots < i_d\}$. The **Schubert variety** for a complete flag $F_\bullet$ and $I \in [n]_d$ is

$$\Omega_I(F_\bullet) := \big\{V \in \mathrm{Gr}_d(\mathbb{C}^n) \mid \dim(V \cap F_{i_\ell}) \geq \ell\big\}.$$

Our focus is the intersection of three Schubert varieties for complete flags $F_\bullet, G_\bullet, H_\bullet$, and $I, J, K \in [n]_d$:

$$T_{IJK} := \Omega_I(F_\bullet) \cap \Omega_J(G_\bullet) \cap \Omega_K(H_\bullet).$$

The majority of our discussion requires that T_{IJK} be transverse. In Section 2.1.2, we discuss how the flags are selected so that the intersection is transverse. For now, it suffices to note why transversality is desirable. If T_{IJK} is transverse for the flags $F_\bullet, G_\bullet, H_\bullet$ and the flags $F'_\bullet, G'_\bullet, H'_\bullet$, then $|T_{IJK}| = |T'_{IJK}|$, where T'_{IJK} is the triple intersection for $F'_\bullet, G'_\bullet, H'_\bullet$.

Let $F_\bullet, G_\bullet, H_\bullet$ be any flags such that T_{IJK} is transverse. We define the **Littlewood-Richardson number** to be the intersection's size in this case when T_{IJK} is finite and 0 otherwise:

$$c_{IJK} = \begin{cases} \big|T_{IJK}\big| & \text{if } T_{IJK} \text{ is finite,} \\ 0 & \text{otherwise.} \end{cases}$$

The Littlewood-Richardson number is often written with an up index, c_{IJ}^K; however, the two notations are not synonymous; rather they are related by $c_{IJK} = c_{IJ}^{K^\vee}$, where the set $K^\vee$ is derived from $K \in [n]_d$ by

$$K^\vee = \{n + 1 - k \mid k \in K\}.$$

1.1.2 Synthesis

Our aim is to state an algorithm that synthesizes the unique point of T_{IJK} when $c_{IJK} = 1$. To write the result, we use the lattice theory operations, meet and join, to create **lattice polynomials**. The set of vector subspaces of $\mathbb{C}^n$ is a lattice where intersection is denoted by **meet**, $\wedge$, and span is denoted by **join**, $\vee$. The lattice operations can be used to construct lattice polynomials where $\wedge$, and $\vee$ replace the usual polynomial operations $+$ and $*$. For example,

$$V = (F_4 \wedge G_9 \wedge H_2) \vee (F_3 \wedge (G_4 \vee H_1)) \vee (H_4 \wedge (G_4 \vee H_1))$$

is a lattice polynomial in the flags $F_\bullet, G_\bullet, H_\bullet$. We define synthesis as follows: $V \in T_{IJK}$ is **synthesized** if there exists a lattice polynomial q such that

$$V = q(F_\bullet, G_\bullet, H_\bullet) \in \Omega_I(F_\bullet) \cap \Omega_J(G_\bullet) \cap \Omega_K(H_\bullet)$$

for all *generic*, complete flags $F_\bullet, G_\bullet, H_\bullet$.

We will discuss the generiticity condition in Subsection 2.1.2. For the moment, it is enough to note that using generic flags ensures that the subspaces we construct have the 'correct' dimension. In particular, any intersection or span of two steps has the expected dimension: $\dim F_i \wedge G_j = \max\{i+j-n, 0\}$ and $\dim F_i \vee G_j = \min\{i+j, n\}$.

Let us illustrate a very simple case of synthesis. Let $n = 5$, $d = 2$, $I = \{2, 3\}$, $J = \{3, 4\}$ and $K = \{4, 5\}$. Further, let $F_\bullet, G_\bullet, H_\bullet$ be generic, complete flags in $\mathbb{C}^5$. We claim that

$$G_4 \wedge F_3 \in T_{IJK} \subset \mathrm{Gr}_2(\mathbb{C}^5).$$

We check that $G_4 \wedge F_3$ is in the first Schubert variety, $\Omega_I(F_\bullet)$, by checking the dimension condition $\dim(V \wedge F_{i_\ell}) \geq \ell$. Generiticity ensures that

$$\begin{aligned} \dim (G_4 \wedge F_3) \wedge F_{i_2} &= \dim G_4 \wedge F_3 = 2 \\ \dim (G_4 \wedge F_3) \wedge F_{i_1} &= \dim G_4 \wedge F_2 = 1 \end{aligned}$$

where $i_1 = 2$, and $i_2 = 3$. Hence, $G_4 \wedge F_3 \in \Omega_I(F_\bullet)$. Similarly, $G_4 \wedge F_3$ is also in the other two varieties.

1.2 Puzzles

The central combinatorial object of this thesis is the puzzle, introduced in [15]. Puzzles are one of several combinatorial objects with a Littlewood-Richardson rule. As such, puzzles are in bijection with the objects discussed in Chapter 3: Littlewood-Richardson tableaux, Berenstein and Zelevinsky's hives, and the measures in [2]. However, puzzles offer a different combinatorial view that has been put to several important uses.

It was shown in [15] that puzzles entirely determined by their boundaries, called *rigid* puzzles, correspond to facets of the Littlewood-Richardson cone. Puzzles also give rise to Horn inequalities via a construction demonstrated in [10], which we detail in Section 3.2. This construction was used to prove a factorization result for Littlewood-Richardson numbers [10]. A generalization of puzzles has even been used for computation in the equivariant cohomology of the Grassmanian; this equivariant puzzle rule was proved by Knutson and Tao in [14]. We now define puzzles and their relevant notation.

1.2.1 Labelled Puzzles

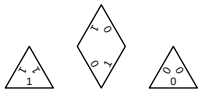


Figure 1.1: The three edge labelled puzzle pieces.

A **labelled puzzle** P sits in a size n equilateral triangle, and it is filled with labelled pieces. The individual pieces are shown in Figure 1.1. The pieces and their rotations are placed in the triangle such that the edge labels of adjacent pieces match and there are no empty positions. Figure 1.2 shows a complete puzzle composed of labelled pieces. The number of 1s on each side of P is the same, so in addition to n, we use another parameter, d, to specify the number of 1 labels per side. Hence, the number of 0 labels per side is $n - d$. The puzzle in Figure 1.2 has $d = 4$.

The borders of P can be viewed as three 01-strings of length n with d ones. We say P has **boundary** π, σ, ρ if π, σ and ρ are strings written clockwise on P's Northwest (NW), Northeast (NE), and south (S) borders respectively. There is a straightforward bijection between $[n]_d$ and 01-strings with d ones: i_ℓ determines the position of the ℓth 1 in a string. The Littlewood-Richardson numbers can be given in terms of these 01-strings; if π_X is the string associ-

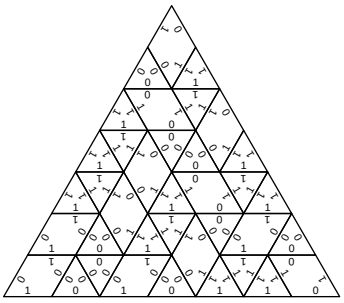


Figure 1.2: A completed puzzle with labelled pieces.

ated to $X \in [n]_d$, then we write $c_{IJK} = c_{\pi_I \pi_J \pi_K}$.

A Littlewood-Richardson rule for $c_{\pi\sigma\rho}$ can be stated for puzzles, which we do now:

Littlewood-Richardson rule 1. *Let π, ρ, σ be n length 01-strings with d 1s. Then*

$$c_{\pi\sigma\rho} = \#\ \textit{puzzles with boundary}\ \pi, \sigma, \rho.$$

This rule is proved in Subsection 2.3.2.

1.2.2 Shaded Puzzles

Although more commonly used, labelled puzzles involve too many symbols for our purposes, as we require puzzles with less adornment to avoid overcrowding our figures. To this end, we introduce **shaded puzzles** with shaded pieces as in Figure 1.3. The piece conversion is as follows: 1-triangles become dark triangles, 0-triangles become white triangles, and labelled rhombi become gray rhombi. Figure 1.3 also shows the puzzle from Figure 1.2 converted to a shaded puzzle. The triangle pieces are called white or dark in the obvious way. A piece edge is called dark if a dark triangle may be placed adjacent to it; otherwise, the edge is called white.

We describe the boundary of shaded puzzles using the sets $I, J, K \in [n]_d$. If I determines the NW boundary of P, then i_ℓ is the position of the ℓth dark edge counting clockwise from the lower left corner. The sets $J, K \in [n]_d$ determine the NE and S boundaries similarly. Unless otherwise required, we always label the NW side by a set I, NE by J, and S by K.

The rotational symmetry of puzzles makes some symmetries of the Littlewood-Richardson numbers clear:

$$c_{IJK} = c_{KIJ} = c_{JKI}.$$

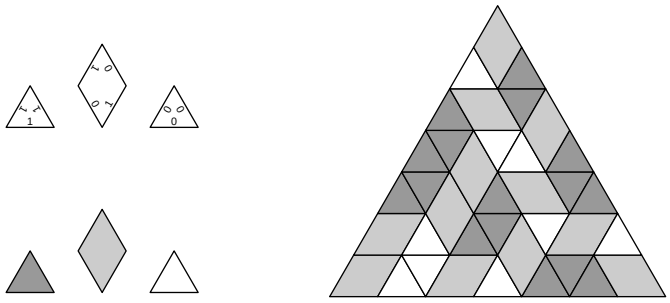


Figure 1.3: The puzzle piece shading.

Puzzles have another symmetry: reflection about a center line. To make the reflection a puzzle, all dark triangles must change to white triangles and vice-versa. The boundary of this new puzzle is given by the sets $J^*, I^*, K^* \in [n]_{n-d}$. These sets determine the positions of the $n-d$ white edges of P counter clockwise, so $I^* = \{i^* \mid i^* \neq n+1-i \text{ for } i \in I\}$. These sets should not be confused with $I^\vee \in [n]_d$. By Littlewood-Richardson rule 1,

$$c_{IJK} = c_{J^*I^*K^*}.$$

1.3 Our Aim and Outline

The goal of this thesis is to state an algorithm that synthesizes a point in

$$\Omega_I(F_\bullet) \cap \Omega_J(G_\bullet) \cap \Omega_K(H_\bullet)$$

when $c_{IJK} = 1$. As noted, our algorithm is a reformulation, using puzzles, of the algorithm in [2], which used specialized Lebesgue measures. These measures are in bijection with puzzles, but we prefer an algorithm in terms of puzzles for several reasons.

Firstly, the use of puzzles makes certain aspects of the proof of correctness for our algorithm are more transparent. Second, the need to switch between two sets of combinatorial objects, measures and their duals, is avoided, as puzzle encode both. Third, puzzles are strongly connected to Horn inequalities, and in fact, our algorithm can be used to find essential Horn inequalities. Lastly, puzzles seem central among the combinatorial objects connected to Littlewood-Richardson rules in that there are straightforward bijections between those other objects and puzzles. It is perhaps because of this that

puzzles have been better studied and more widely applied than the measures in [2], so a synthesis algorithm stated in terms of puzzles more clearly connects with that previous work.

Chapter 2 delves into the triple intersection of Schubert varieties and the relation to puzzles in greater detail. The first section examines the Grassmanian, and Schubert varieties. We will prove some facts about Richardson varieties. An important fact about the intersection of a Richardson variety and a special Schubert variety is also stated. The claim that synthesis is not possible when $c_{IJK} \geq 2$ is justified in Subsection 2.1.2. In the following section, we explore the cohomology of the Grassmanian. Some facts about the Schubert classes are proven using the results for Schubert varieties from Subsection 2.1.1. In Section 2.2.2, the relationship between Schur functions and Schubert classes is made explicit when a linear map from the ring of symmetric polynomials to the cohomology ring is shown to be a ring homomorphism. Finally, the puzzle-based Littlewood-Richardson rule is proved in Section 2.3.

In Chapter 3, we discuss three combinatorial devices related to puzzles: the measures from [2], hives, and Littlewood-Richardson tableaux. In each case, we state the associated Littlewood-Richardson rule and prove a bijection with puzzles. The measures are discussed at length, as the facts we state in Section 3.1 are used in the proofs of correctness in Chapter 4. With hives, we include a discussion of the Horn inequalities, and we describe a construction of the inequalities combining hives d puzzles.

Our synthesis algorithm is stated in Chapter 4. The algorithm consists of three 'layers': walkers, the Puzzle Skeleton algorithm, and the Lattice Polynomial algorithm. A section is devoted to each. Section 4.1 will define the walkers, a method of marking puzzle pieces. The Puzzle Skeleton algorithm of Section 4.2 uses walkers to search for a desirable set of marks in a puzzle. Finally, the Lattice Polynomial algorithm employs the desirably marked puzzle to synthesize the point in a triple intersection of Schubert varieties.

Chapter 2

The Triple Intersection of Schubert Varieties

The focus of this chapter is the triple intersection of Schubert varieties. As mentioned in Section 1.1.1, Schubert calculus is the study of linear algebra problems phrased as intersections of subvarieties of flag manifolds. In modern terms, this is the study of Schubert classes in the cohomology ring of a flag manifold, the complex Grassmanian in our case. Products of Schubert classes in the cohomology ring correspond to transverse intersections of the underlying Schubert varieties. As such, the product structure of these classes can be used to study the intersection theory of the manifold and vice-versa.

In Section 2.1, we develop the necessary ideas about Schubert varieties. In particular, we discuss Richardson varieties and their intersection with a special Schubert variety in Subsection 2.1.1. We include a discussion of synthesis, by a lattice polynomial, of points in a triple intersection of Schubert varieties in Subsection 2.1.2, and we demonstrate why synthesis is not possible when the intersection has size greater than one. We then move to looking at the cohomology ring $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$ in Section 2.2. Specifically, we study the Schubert classes which are the cohomology classes of Schubert varieties. The chapter finishes with a discussion of puzzles and a proof of the puzzle-based Littlewood-Richardson rule. The majority of the discussion comes from [6]. We begin with the Schubert varieties of the Grassmanian.

2.1 Schubert Varieties

For ease of reference, we recall the definition of Schubert varieties from Subsection 1.1.1. Let $I \in [n]_d$, and $F_\bullet$ be a complete flag. The Schubert variety for I and $F_\bullet$ is

$$\Omega_I(F_\bullet) := \big\{V \in \mathrm{Gr}_d(\mathbb{C}^n) \mid \dim(V \cap F_{i_\ell}) \geq \ell\big\}.$$

The two extreme cases for this definition are the set $\{1, 2, \ldots, d\} \in [n]_d$, which we denote $I^{\min}$, and $\{n-d+1, \ldots, n\} = I^{\max}$. The Schubert variety for $I^{\min}$ is $\Omega_{I^{\min}}(F_\bullet) = \{F_d\}$, a single point, as only $V = F_d$ satisfies $\dim(V \cap F_d) \geq d$. For $I^{\max}$, $\Omega_{I^{\max}}(F_\bullet) = \mathrm{Gr}_d(\mathbb{C}^n)$, as every d-dimensional subspace of $\mathbb{C}^n$ obeys

$$\begin{aligned}\dim(V \cap F_{n-d+\ell}) &\geq \dim(V) + \dim(F_{n-d+\ell}) - n \\ &= (d) + (n-d+\ell) - n \\ &= \ell.\end{aligned}$$

Schubert varieties are also commonly indexed by 01-strings. We write the Schubert variety $\Omega_I(F_\bullet)$ as $\Omega_\pi(F_\bullet)$ when π is the 01-string corresponding to I under the bijection in Subsection 1.2.2. We note that the use of the term variety is no misnomer: $\Omega_I(F_\bullet)$ is indeed a projective subvariety of $\mathrm{Gr}_d(\mathbb{C}^n)$. We do not include the details here; they can be found in [6].

The **Schubert cells** are affine subvarieties of Schubert varieties. They are given by

$$\Omega_I^\circ(F_\bullet) := \big\{V \mid \dim(V \cap F_k) = \ell \text{ for } i_\ell \leq k < i_{\ell+1}\big\}.$$

If V is in $\Omega_I^\circ(F_\bullet)$, the sequence $\dim(V \cap F_k)$ stabilizes between consecutive values in I. We note two facts about $\Omega_I^\circ(F_\bullet)$ from [6]. First, $\Omega_I^\circ(F_\bullet)$ is open in the Zariski topology relative to $\Omega_I(F_\bullet)$, so it is dense in $\Omega_I(F_\bullet)$. Second, Schubert varieties can be expressed as a disjoint union of Schubert cells:

$$\Omega_I(F_\bullet) = \coprod_{J \leq I} \Omega_J^\circ(F_\bullet),$$

where $J \leq I$ if $I, J \in [n]_d$ and $j_\ell \leq i_\ell$ for $1 \leq \ell \leq d$.

Another way to define the Schubert cells is by using a basis for $V \in \mathrm{Gr}_d(\mathbb{C}^n)$. Select a basis of $\mathbb{C}^n$, $\{f_1, \ldots, f_n\}$ such that $F_\ell = \mathrm{span}\{f_1, \ldots, f_\ell\}$. Then for $V \in \mathrm{Gr}_d(\mathbb{C}^n)$, select a basis $\{v_1, \ldots, v_d\}$. We have that $v_\ell =$

$$A \longrightarrow \begin{bmatrix} * & 0 & * & * & 0 & * & 0 & 1 \\ * & 0 & * & * & 0 & * & 1 & 0 \\ * & 0 & * & * & 1 & 0 & 0 & 0 \\ * & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 2.1: Gaussian elimination from the right for $V \in \Omega^\circ_{2,5,7,8}(F_\bullet)$.

$\sum A_{\ell k} f_k$, so we can form the matrix $A = A_{\ell k}$. The matrix A is then put in reduced form using Gaussian elimination from the right, as in Figure 2.1. The reduced matrix has d pivot columns, and if column i_ℓ is the ℓth pivot column, then $\dim(V \cap F_{i_\ell}) = \ell$. Hence, $\Omega^\circ_I(F_\bullet)$ can be defined as the set of subspaces whose reduced matrices have pivot columns given by I. In Figure 2.1, we see the general form of a matrix for a subspace $V \in \Omega^\circ_{2,5,7,8}(F_\bullet)$. The stars represent arbitrary complex entries. The dimension of $\Omega^\circ_I(F_\bullet)$ is the number of stars in the general form:

$$\dim\big(\Omega^\circ_I(F_\bullet)\big) = \sum_{\ell=1}^{d} \big(i_\ell - \ell\big).$$

As $\Omega^\circ_I(F_\bullet)$ is dense in $\Omega_I(F_\bullet)$, the dimension of $\Omega_I(F_\bullet)$ is the same as that of $\Omega^\circ_I(F_\bullet)$. In particular, the dimension of $\mathrm{Gr}_d(\mathbb{C}^n)$ is $d(n-d)$ as $\mathrm{Gr}_d(\mathbb{C}^n) = \Omega_{I^{\max}}(F_\bullet)$ and $i^{\max}_\ell = n - d + \ell$.

2.1.1 Notable Intersections

Products in the cohomology ring of $\mathrm{Gr}_d(\mathbb{C}^n)$ can be computed by intersection theory, so the intersections of Schubert varieties are of special interest. In particular, we want to know the size of the intersection

$$T_{IJK} := \Omega_I(F_\bullet) \cap \Omega_J(G_\bullet) \cap \Omega_K(H_\bullet)$$

for complete flags $F_\bullet, G_\bullet, H_\bullet$. The focus of this section is on developing the necessary facts about Schubert varieties needed to compute $|T_{IJK}|$ in general. Clearly, we need $|T_{IJK}|$ to be finite to say anything meaningful about it. Two conditions are needed for finiteness.

The first is a condition on the sets I, J, K. If the sets I, J, K obey

$$2d(n-d) = \sum_{\ell=1}^{d} \big(i_\ell + j_\ell + k_\ell - 3\ell\big),$$

then the intersection T_{IJK} is possibly finite. We assume this condition for the remainder of this chapter. The second condition is transversality, discussed in Subsection 2.1.2, which ensures that T_{IJK} is in indeed finite.

The computation of $|T_{IJK}|$ in general will be accomplished using the cohomology ring, but some initial calculations must be done directly with the Schubert varieties. In particular, we prove two facts about Richardson varieties, and then state a fact about the size of an intersection of a Richardson variety with a special Schubert variety.

The first intersection we look at uses an **opposite flag**. Select a basis for $\mathbb{C}^n$: $\{f_1, \ldots, f_\ell\}$. Let $F_\bullet$ be the complete flag with $F_\ell = \text{span}\{f_1, \ldots, f_\ell\}$. The opposite flag in the basis $\{f_1, \ldots, f_\ell\}$ is $\widetilde{F}_\bullet$ where the ℓth step is given by $\widetilde{F}_\ell = \text{span}\{f_{n-\ell+1}, \ldots, f_n\}$. For a Schubert variety $\Omega_I(F_\bullet)$, an **opposite variety** is $\Omega_J(\widetilde{F}_\bullet)$. When the flags are understood, we instead write Ω_I and $\widetilde{\Omega}_J$ for a variety and an opposite variety. A **Richardson variety** is the intersection $\Omega_I \cap \widetilde{\Omega}_J$. We state and prove two theorems for Richardson varieties.

Theorem 2.1.1. *$\Omega_I \cap \widetilde{\Omega}_J \neq \emptyset$ if and only if $n+1 \leq i_\ell + j_{d+1-\ell}$ for $\ell = 1, \ldots, d$.*

Proof. Let $j_{d+1-\ell} = k_\ell$. $F_{i_\ell} \cap \widetilde{F}_{k_\ell}$ is spanned by $\{f_t \mid n+1-k_\ell \leq t \leq i_\ell\}$, which may be empty, and it has dimension $i_\ell + k_\ell - n$ when non-empty. Suppose $V \in \Omega_I \cap \widetilde{\Omega}_J$. Then $\dim(V \cap F_{i_\ell}) \geq \ell$ and $\dim(V \cap \widetilde{F}_{k_\ell}) \geq d+1-\ell$. V is d-dimensional, so

$$1 \leq \dim\left((V \cap F_{i_\ell}) \cap (V \cap \widetilde{F}_{k_\ell})\right) \leq \dim(F_{i_\ell} \cap \widetilde{F}_{k_\ell}) = i_\ell + k_\ell - n,$$

as required.

Conversely, if $n+1 \leq i_\ell + k_\ell$ for $\ell = 1, \ldots, d$, then $1 \leq \dim(F_{i_\ell} \cap \widetilde{F}_{k_\ell})$. Define the vectors

$$v_\ell = f_{n-k_\ell+1} + \cdots + f_{i_\ell}.$$

Let V be their span. It is easily seen that $\dim(V) = d$, $\dim(V \cap F_{i_\ell}) \geq \ell$ and $\dim(V \cap \widetilde{F}_{k_\ell}) \geq d+1-\ell$, so $\Omega_I \cap \widetilde{\Omega}_J \neq \emptyset$, as required. □

The next theorem involves the set derived from $I \in [n]_d$ defined in Subsection 1.1.1: $I^\vee = \{n+1-i_\ell \mid i_\ell \in I\}$. It is easy to see that $i^\vee_{d+1-\ell} = n+1-i_\ell$.

Theorem 2.1.2. *If $\sum_I i + \sum_J j = d(n+1)$, then*

$$\Omega_I \cap \widetilde{\Omega}_J = \begin{cases} \text{a point if } J = I^\vee \\ \emptyset \text{ otherwise,} \end{cases}$$

and the intersection is transverse at the point when non-empty.

Proof. If $J \neq I^\vee$, then for some ℓ, $i_\ell + j_{d+1-\ell} < n+1$, so $\Omega_I \cap \widetilde{\Omega}_J = \emptyset$ by the previous theorem.

Say $J = I^\vee$. Then $j_{d+1-\ell} = n+1-i_\ell$, so $\Omega_I \cap \widetilde{\Omega}_J \neq \emptyset$. Let $V \in \Omega_I \cap \widetilde{\Omega}_J$. Then by the same argument as in the previous theorem,

$$1 \leq \dim(V \cap F_{i_\ell} \cap \widetilde{F}_{n+1-i_\ell}) \leq \dim(F_{i_\ell} \cap \widetilde{F}_{n+1-i_\ell}) = i_\ell + j_{d+1-\ell} = 1,$$

for $1 \leq \ell \leq d$. We also have for each ℓ that $F_{i_\ell} \cap \widetilde{F}_{n+1-i_\ell} = \operatorname{span}(f_{i_\ell})$. There is a basis of V including the vector f_{i_ℓ} for $1 \leq \ell \leq d$. Hence, $\Omega_I \cap \widetilde{\Omega}_J$ consists of the single point $V = \operatorname{span}\{f_{i_1}, \ldots, f_{i_d}\}$. □

We delay proof that $\Omega_I \cap \widetilde{\Omega}_J$ is transverse until Section 2.2.1.

Now, let $I, J \in [n]_d$. We say that J is an m-**shift** of I, denoted $I \xrightarrow{m} J$, if $m = m_1 + \cdots + m_d$, each $m_\ell \geq 0$, and

$$i_{\ell-1} < j_\ell = i_\ell - m_\ell.$$

A special Schubert variety is indexed by the unique m-shift of $I^{\max}$, which we denote J_m. A **special Schubert variety** is $\Omega_{J_m}(G_\bullet)$. The next intersection we discuss is between a Richardson variety and a special Schubert variety.

Theorem 2.1.3. *Let $m \geq 0$, $I \geq K^\vee$ and $\sum_\ell i_\ell - \sum_\ell k_\ell^\vee = m$. Then*

1. *if $K^\vee$ is not an m-shift of I, then there exists a flag $G_\bullet$ such that*

$$\Omega_I \cap \widetilde{\Omega}_K \cap \Omega_{J_m}(G_\bullet) = \emptyset.$$

2. *if $I \xrightarrow{m} K^\vee$, then there exists a flag $G_\bullet$ such that*

$$\Omega_I \cap \widetilde{\Omega}_K \cap \Omega_{J_m}(G_\bullet) = \{V\},$$

a single point, and the intersection is transverse at V.

The proof, though slightly more involved, follows lines similar to the two proofs above, so it is omitted here. It can be found in [6].

2.1.2 Monodromy Prevents Synthesis

We now discuss synthesis in the Grassmanian. Recall that a subspace

$$V \in T_{IJK} := \Omega_I(F_\bullet) \cap \Omega_J(G_\bullet) \cap \Omega_K(H_\bullet)$$

is said to be **synthesized** if it can be described in terms of a lattice polynomial built from the steps of the complete flags $F_\bullet, G_\bullet, H_\bullet$. We show that synthesis is not possible when $c_{IJK} > 1$.

We examine the monodromy of the intersection points of a triple intersection as we continuously deform the three given flags $F_\bullet, G_\bullet, H_\bullet$. Consider a continuous function from $[0,1] \to \mathcal{F}\ell(\mathbb{C}^n)$ given by $t \mapsto F_\bullet(t)$ with $F_\bullet(0) = F_\bullet(1) = F_\bullet$. Consider $G_\bullet(t)$ and $H_\bullet(t)$ similarly. Each intersection point is then defined by a continuous function from $[0,1] \to \mathrm{Gr}_d(\mathbb{C}^n)$ where $t \mapsto V(t)$ with

$$V(0), V(1) \in T_{IJK}.$$

The monodromy of T_{IJK} is given by the group of all ways the intersection points can be permuted in this way. The salient point is that if the monodromy of a point V is non-trivial, then $V(0)$ need not be equal to $V(1)$. We show below that when $c_{IJK} > 1$, the monodromy of every point in T_{IJK} is non-trivial, and thus, a deformation can be found with $V(0) \neq V(1)$.

There is the danger that the deformations we perform change the size of $|T_{IJK}|$. To understand how this might occur, consider the intersection of two circles. If the circles are tangent, then a perturbation can change a single point of intersection into two or none. The size of T_{IJK} can exhibit similar sensitivity to perturbation. However, the term 'non-tangentially' is not the correct notion for general manifolds; instead, we desire T_{IJK} to be a transverse intersection. We do not define transversality here. For us, it is enough to note that when an intersection of Schubert varieties is *transverse*, infinitesimal perturbations of the flags do not change $|T_{IJK}|$. Let $F_\bullet(t), G_\bullet(t), H_\bullet(t)$ be such that T_{IJK} is transverse for all t. Then $|T_{IJK}| = c_{IJK}$ for all t.

However, we may not have such flags to choose from for a given I, J, K. Fortunately, it is always possible to choose three flags such that T_{IJK} is transverse. This follows from Kleiman's transversality theorem [11] which, in this case, states that for any three complete flags $F_\bullet, G_\bullet, H_\bullet$ there is an open, dense subset $\mathcal{U} \subseteq GL_n^3$ such that

$$\Omega_I(g_1 F_\bullet), \Omega_J(g_2 G_\bullet), \text{ and } \Omega_K(g_3 H_\bullet)$$

have transverse intersection for all $(g_1, g_2, g_3) \in \mathcal{U}$. In other words, a *generic* choice of three flags produces a transverse Schubert variety intersection. Hence, we can select $F_\bullet(t), G_\bullet(t), H_\bullet(t)$ such that the triple intersection they define is transverse for all t, which we do for the remainder of this discussion. See [8] for more details about transversality and Kleiman's theorem.

We now prove that when $c_{IJK} \geq 2$ and the monodromy of an intersection point is non-trivial, we cannot synthesize that point. Let $V, W \in T_{IJK}$, and suppose that the continuous function with $V(0) = V$ has $V(1) = W \neq V$. Suppose further that V is given by a lattice polynomial p: $V = p(F_\bullet, G_\bullet, H_\bullet)$. We have that $V(t) = p(F_\bullet(t), G_\bullet(t), H_\bullet(t))$ as spans and intersections change continuously with t, so $p(F_\bullet(1), G_\bullet(1), H_\bullet(1)) = V(1) = W$. We also have $F_\ell(0) = F_\ell(1)$ and similarly for the other flags, so $p(F_\bullet(0), G_\bullet(0), H_\bullet(0)) = p(F_\bullet(1), G_\bullet(1), H_\bullet(1))$; a contradiction, as $V \neq W$. Hence, if V is synthetic, then the monodromy of the intersection point V is trivial.

Of course, when $c_{IJK} = 1$, the monodromy of the unique intersection point is trivial, so synthesis may be possible in this case. We now show that when $c_{IJK} \geq 2$ the monodromy of every intersection point is non-trivial [18], so synthesis is impossible. Define

$$\mathcal{U} = \big\{(F_\bullet, G_\bullet, H_\bullet) \mid \Omega^\circ_I(F_\bullet) \cap \Omega^\circ_J(G_\bullet) \cap \Omega^\circ_K(H_\bullet) \text{ is transverse}\big\}.$$

The set $\mathcal{U}$ is open and dense in $\mathcal{F}\ell(\mathbb{C}^n)^3$ by Kleiman's theorem, and it is therefore connected, as $\mathcal{F}\ell(\mathbb{C}^n)^3$ is connected. Let I, J, K be such that $c_{IJK} \geq 2$. Define the space

$$\mathcal{T} = \big\{(V, F_\bullet, G_\bullet, H_\bullet) \mid V \in T_{IJK},\ (F_\bullet, G_\bullet, H_\bullet) \in \mathcal{U}\big\}.$$

Also, define two projections:

$$\begin{aligned} q(V, F_\bullet, G_\bullet, H_\bullet) &= (F_\bullet, G_\bullet, H_\bullet), \\ r(V, F_\bullet, G_\bullet, H_\bullet) &= V. \end{aligned}$$

We have that $q^{-1}(F_\bullet, G_\bullet, H_\bullet) = T_{IJK}$. It is not hard to show that $\mathcal{T}$ is connected; however, it requires showing that the fibers are Schubert cells in $\mathcal{F}\ell_{\underline{d}}(\mathbb{C}^n)$ for some step sequence $\underline{d}$, which are connected. Consider a path $e_t \in \mathcal{T}$, $t \in [0, 1]$ such that $q(e_0) = q(e_1) = (F_\bullet, G_\bullet, H_\bullet)$. Since $\mathcal{T}$ is connected and $q^{-1}(F_\bullet, G_\bullet, H_\bullet)$ has more than one point, we can choose e_t such that $r(e_0) \neq r(e_1)$. Hence, moving around the loop $q(e_t)$ in $\mathcal{U}$ permutes the points of T_{IJK} non-trivially. We record this as a theorem.

Theorem 2.1.4 ([18]). *Let $I, J, K \in [n]_d$, and $F_\bullet, G_\bullet, H_\bullet$ be complete flags such that T_{IJK} is transverse. If there exists a lattice polynomial p such that $p(F_\bullet, G_\bullet, H_\bullet) \in T_{IJK}$, then $c_{IJK} = 1$.*

2.2 The Cohomology Ring $H^*(\mathbf{Gr}_d(\mathbb{C}^n))$

Cohomology associates a ring, $H^*(X)$, to a topological space X, and elements of that ring to certain subobjects of the space. In the case of $\mathrm{Gr}_d(\mathbb{C}^n)$, the subobjects are subvarieties of $\mathrm{Gr}_d(\mathbb{C}^n)$, and certain classes of the ring $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$ are associated with the subvarieties of $\mathrm{Gr}_d(\mathbb{C}^n)$. The classes defined by Schubert varieties form a linear basis for $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$. The intersections of the previous section can be interpreted as products of these Schubert classes. The aim of this section is to exploit this connection between the intersection theory of $\mathrm{Gr}_d(\mathbb{C}^n)$ and the ring $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$. In particular, a Pieri rule is proved for $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$, and a surjection from the well understood ring of symmetric functions to the cohomology ring follows.

2.2.1 Schubert Classes

The **Schubert class** for I, denoted by σ_I, is the class in $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$ of a corresponding Schubert variety:

$$\sigma_I := \big[\Omega_I(F_\bullet)\big].$$

These classes form a basis for $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$. We now show that this definition does not depend on the choice of flag.

Proposition 2.2.1. *Let $F_\bullet$ and $H_\bullet$ be complete flags. Then $[\Omega_I(F_\bullet)] = [\Omega_I(H_\bullet)]$.*

Proof. There exists $h \in GL_n$ such that $hF_\bullet = H_\bullet$, where the action is on each step of $F_\bullet$. The action of GL_n on Schubert varieties is then given by

$$h\Omega_I(F_\bullet) = \Omega_I(H_\bullet).$$

Define $h_t : [0,1] \to GL_n$ to be continuous such that $h_0 =$Id and $h_1 = h$. This gives a homotopy from the map

$$\text{Id: } \Omega_I(F_\bullet) \to \Omega_I(F_\bullet) \text{ to the map } h : \Omega_I(F_\bullet) \to \Omega_I(H_\bullet).$$

Hence, $[\Omega_I(F_\bullet)] = [\Omega_I(H_\bullet)]$, as homotopic varieties represent the same cohomology class. □

We turn now to the product structure of $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$. The product of two Schubert classes, $\sigma_I\sigma_J$, is represented by the intersection of the Schubert varieties

$$\sigma_I\sigma_J = [\Omega_I(F_\bullet) \cap \Omega_J(G_\bullet)],$$

provided the intersection is transverse. Since $\Omega_{I^{\max}}(F_\bullet) = \mathrm{Gr}_d(\mathbb{C}^n)$, $\sigma_{I^{\max}}$ is the identity element in $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$.

We also require transversality when calculating the pairing between homology and cohomology. The pairing between the fundamental class of $\mathrm{Gr}_d(\mathbb{C}^n)$ and a cohomology class σ is commonly denoted by $\int_{\mathrm{Gr}_d(\mathbb{C}^n)} \sigma$. The integral notation comes from the de Rham theory of cohomology, where it is indeed an integral. Here, it is simply notation for the pairing. In the case of a triple product of Schubert classes, the pairing computes the size of a transverse intersection of the three Schubert varieties:

$$\int_{\mathrm{Gr}_d(\mathbb{C}^n)} \sigma_I\sigma_J\sigma_K := \left|\Omega_I(F_\bullet) \cap \Omega_J(G_\bullet) \cap \Omega_K(H_\bullet)\right| = c_{IJK}.$$

Since σ_I is independent of flag choice, we may choose generic flags $F_\bullet$, $G_\bullet$, and $H_\bullet$ to ensure that the intersection is transverse.

In the case of a Richardson variety, if $F_\bullet$ and $G_\bullet$ are two generic flags, it is possible to select a basis such that $G_\bullet = \widetilde{F}_\bullet$. Multiplying by $h \in GL_n$, we see that any pair of opposite flags produces transverse intersections, as multiplication by h maintains transversality. We have the following corollaries to Theorem 2.1.2.

Corollary 2.2.2. *If $\sum_I i + \sum_J j = d(n+1)$, then*

$$\int_{Gr_d(\mathbb{C}^n)} \sigma_I\sigma_J = \delta_{IJ^\vee}.$$

Corollary 2.2.3.

$$\sigma_I\sigma_J = \sum_L c_{IJL}\ \sigma_{L^\vee}.$$

Proof. Write the product $\sigma_I\sigma_J$ in the Schubert basis: $\sigma_I\sigma_J = \sum_L d_{IJL}\ \sigma_{L^\vee}$.

Then we have that

$$\begin{aligned}\int_{\mathrm{Gr}_d(\mathbb{C}^n)} \sigma_I\sigma_J\sigma_K &= \sum_L d_{IJL} \int_{\mathrm{Gr}_d(\mathbb{C}^n)} \sigma_{L^\vee}\sigma_K \\ &= \sum_L d_{IJL}\ \delta_{L^\vee K^\vee} \\ &= d_{IJK}.\end{aligned}$$

Hence, $d_{IJK} = c_{IJK}$. □

The intersection of the Richardson variety and a special variety is also transverse by Kleiman's theorem giving us a corollary to Theorem 2.1.3.

Corollary 2.2.4. *Let $m \geq 0$, $I \geq K^\vee$, and $\sum_I i - \sum_{K^\vee} k^\vee = m$. Then*

$$\int_{Gr_d(\mathbb{C}^n)} \sigma_I\sigma_{J_m}\sigma_K = \begin{cases} 1 & \text{if } I \overset{m}{\to} K^\vee \\ 0 & \text{otherwise.}\end{cases}$$

This gives the **Pieri Rule** for Schubert classes.

Corollary 2.2.5. *Let $m \geq 0$. Then*

$$\sigma_I\sigma_{J_m} = \sum_{I \overset{m}{\to} K} \sigma_K.$$

Proof. Consider the product $\sigma_I\sigma_{J_m} = \sum_L c_{IJ_mL}\ \sigma_{L^\vee}$. Combining Corollaries 2.2.2 and 2.2.4, we get that

$$c_{IJ_mK} = \int_{\mathrm{Gr}_d(\mathbb{C}^n)} \sigma_I\sigma_{J_m}\sigma_K = 1$$

when $I \overset{m}{\to} K^\vee$, and 0 otherwise. Hence,

$$\sigma_I\sigma_{J_m} = \sum c_{IJ_mK}\sigma_K^\vee = \sum_{I \overset{m}{\to} K^\vee} \sigma_{K^\vee}.$$

□

2.2.2 Map From Schur Polynomials to Schubert Classes

In this section, we describe the connection between the ring of symmetric polynomials in d variables, $\Lambda(d)$, and the cohomology ring for the Grassmanian. In particular, we will show that a linear map taking the Schur basis to the Schubert basis is in fact a ring homomorphism. The ring homomorphism allows Schubert classes to be represented by Schur polynomials. The homomorphism also makes clear how the Littlewood-Richardson numbers, defined in Subsection 1.1.1 relate to the classical definition: Littlewood-Richardson numbers are the connection coefficients of a product of Schur polynomials in the ring of symmetric polynomials. The Schur polynomials are indexed by partitions. We now recall the definition of partitions and their diagrams, and we define Schur polynomials after that.

A **partition** $\lambda = (\lambda_1, \ldots, \lambda_t)$ is a weakly decreasing, finite sequence of non-negative integers which sum to an integer $N > 0$. The partition λ is said to partition N, denoted $\lambda \vdash N$. The integer λ_ℓ is called the ℓth **part** of λ. A partition can also be given by its diagram. The **diagram** of a partition λ is drawn by placing t rows of boxes with the ℓth row having λ_ℓ boxes, as in Figure 2.2. A partition is said to fit within an $n - d$ by d box if the partition's diagram fits within that box. We denote the partitions that fit within an $n - d$ by d box by $\mathrm{Part}_{n,d}$.

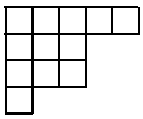


Figure 2.2: The diagram for $\lambda = (5, 3, 3, 1)$.

We also need to define the **skew partition** λ/ν. Suppose ν is a partition whose diagrams fits within λ's. The diagram of the skew partition λ/ν is constructed by laying the diagram of ν on λ and removing any boxes that overlap. A skew partition diagram is pictured in Figure 2.3.

Now, the Schur polynomial indexed by the partition (m) for $m > 0$ is the

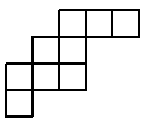


Figure 2.3: The diagram for the skew partition $\lambda/\nu = (5, 3, 3, 1)/(2, 1)$.

homogeneous, symmetric polynomial of degree m:

$$s_{(m)} := \sum_{1 \le i_1 \le i_2 \le \cdots \le i_m \le d} x_{i_1} x_{i_2} \cdots x_{i_m}.$$

For a general partition λ, the **Schur polynomial** for λ is the following:

$$s_\lambda := \det(s_{(\lambda_i + j - i)})_{i,j=1,\ldots,d}.$$

The Schur polynomials form a basis of $\Lambda(d)$, so the definition for s_λ makes it clear that $s_{(m)}$ for $m \ge 0$ generate the ring $\Lambda(d)$.

There is a bijection between $[n]_d$ and $\mathrm{Part}_{n,d}$ that allows us to index Littlewood-Richardson numbers by partitions. For $I \in [n]_d$, $\lambda(I)$ is the partition where

$$\lambda_\ell(I) = n + 1 - i_{d+1-\ell} - \ell.$$

The indices of a Littlewood-Richardson number are simply replaced using this bijection: $c_{IJK^\vee} = c_{IJ}^K = c_{\lambda(I)\lambda(J)}^{\lambda(K)}$. The Littlewood-Richardson numbers are often defined as the connection coefficients of a product of Schur polynomials in $\Lambda(d)$, and we record this here as a theorem.

Theorem 2.2.6. *Let μ, ν, λ be partitions. Then*

$$s_\mu s_\nu = \sum_\lambda c_{\mu\nu}^\lambda s_\lambda.$$

Proof. As we did for Schubert classes, we can state a Pieri rule for Schur polynomials. We say that λ/ν is an m**-strip** if the diagram of λ/ν has at most one box per column. We say that an m-strip was added to ν to form λ if λ/ν is an m-strip. We denote this by $\nu \xrightarrow{m} \lambda$. The Pieri rule for Schur functions is

$$s_\nu s_{(m)} = \sum_{\nu \xrightarrow{m} \lambda} s_\lambda.$$

Using the bijection between $[n]_d$ and $\mathrm{Part}_{n,d}$, it is easily checked that $\lambda(J_m) = (m)$, and that adding an m-strip corresponds to an m-shift. Let

Figure 2.4: Diagram of (5), and a 5-strip.

$I \xrightarrow{m} K$. Then we have that $i_{\ell-1} < i_\ell - m_\ell = k_\ell$ for $\ell = 1, \ldots, d$ and $m = m_1 + \cdots + m_d$. Hence, $\lambda_\ell(I) = \lambda_\ell(K) + m_\ell \leq \lambda_{\ell-1}(I)$, and so

$$\lambda(I) \xrightarrow{m} \lambda(K) \text{ if and only if } I \xrightarrow{m} K.$$

We define the unique, surjective, linear map $\Lambda(d) \to H^*(\mathrm{Gr}_d(\mathbb{C}^n))$ given by

$$s_\nu \mapsto \begin{cases} \sigma_I & \text{if } \nu = \lambda(I), \\ 0 & \text{otherwise.} \end{cases}$$

Since $\Lambda(d)$ is generated by $s_{(m)}$, the Pieri rules on both sides confirm that this linear map is a surjective ring homomorphism; specifically,

$$s_{\lambda(I)} s_{(m)} \mapsto \sigma_I \sigma_{J_m},$$

and

$$\sum_{\lambda(I) \xrightarrow{m} \lambda(K)} s_{\lambda(K)} \mapsto \sum_{I \xrightarrow{m} K} \sigma_K.$$

Hence, the connection coefficients of $\sigma_I \sigma_J$ in $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$ are the connection coefficients for a product of Schur polynomials in $\Delta(d)$. □

The proof of Theorem 2.2.6 also shows us that $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$ is generated by σ_{J_m}, as $\Lambda(d)$ is generated by $s_{(m)}$.

2.3 The Puzzle-Based Littlewood-Richardson Rule

In this section, we prove the puzzle-based Littlewood-Richardson rule. Let $I, J, K \in [n]_d$. Then

$$c_{IJK} = |\Omega_I(F_\bullet) \cap \Omega_J(G_\bullet) \cap \Omega_K(H_\bullet)| = \#\text{ puzzles with boundary } I, J, K.$$

Our ultimate aim is to provide a synthesized description of an intersection point, so we are interested in the case where there is a single puzzle with boundary I, J, K. This occurs when the boundary allows no choice in the placement of puzzle pieces. We call such puzzles **rigid**. In [15], a puzzle P with boundary I, J, K is shown to be rigid if and only if P has no gentle loops, which are gentle paths with the same start and end. Gentle paths are also central to the proof of the puzzle Littlewood-Richardson rule. We define gentle paths and loops next, and prove the Littlewood-Richardson rule in Subsection 2.3.2.

2.3.1 Gentle Paths

We follow the definition of gentle paths from [15]. Let g be a path that walks through a puzzle P along puzzle piece edges, obeying the following conditions:

1. g can never use an edge between pieces of the same type or an edge on the boundary,
2. dark triangles are always to the right of g, and
3. white triangles are always to the left.

A **gentle turn** is a $\pm 60^\circ$ turn in such a path, and such a path is a **gentle path** if all its turns are gentle. A **gentle loop** is a closed gentle path. Figure 2.5 shows a gentle path and a gentle loop in a puzzle.

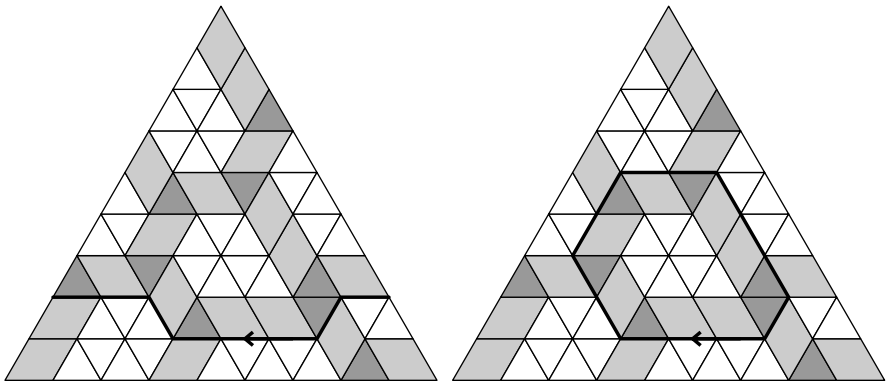


Figure 2.5: A puzzle with a gentle path (left), and gentle loop (right).

Figure 2.6: The four possible piece arrangements around a gentle path edge oriented right to left.

Figure 2.7: The four arrangements after breathing and reorienting the loop from left.to right.

Now, consider all the pieces with either a side or a corner adjacent to a gentle path. These pieces can be partitioned into one of the four arrangements shown in Figure 2.6, where the bold edge is the only edge of the gentle path in the arrangement. A gentle path allows the repositioning of pieces adjacent to the path without affecting the rest of the puzzle. This is called **breathing** a gentle path, and it can be done to closed paths or to paths between two points on the boundary of P. At each edge e of the gentle path, the piece arrangement at e is changed as in going from Figure 2.6 to Figure 2.7: that is, the triangular cases are turned so the other triangle piece is adjacent to the gentle path edge, and the other two cases are flipped about the gentle path edge. In the following theorem, we prove that a puzzle is rigid if and only if it has no gentle loops.

Theorem 2.3.1. *Let P be a puzzle. Then P is non-rigid if and only if there is a gentle loop in P.*

Proof. Suppose P has a gentle loop. Breathing that gentle loop produces another puzzle with the same boundary as P, so P is not rigid.

Conversely, if P is non-rigid, then there are at least 2 puzzle with the same boundary as P. We find a gentle loop in P. Let $P_1 = P$, and let P_2 be another puzzle with the same boundary such that the number of positions that differ from P_1 is minimal. Remove from P_1 all the pieces that differ from P_2. As P_2 is minimally different, there is only one empty region in P_1. Let R be that empty region. If we walk along an edge of R, the puzzle piece rules force our turns to be $\pm 60°$, as any turn of $120°$ has a single piece that may be placed at that turn.

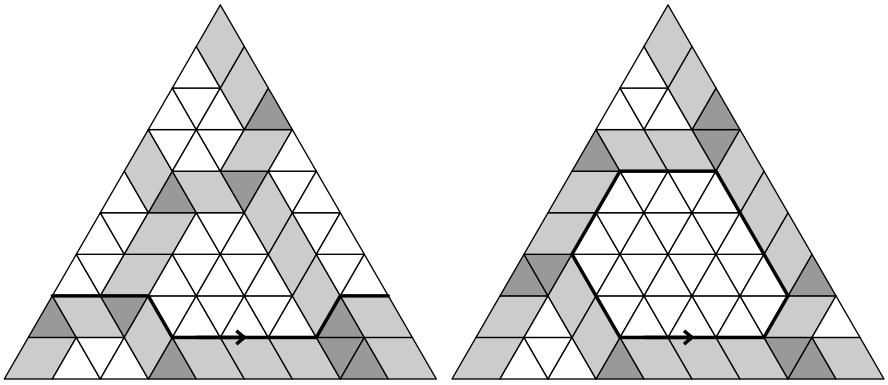


Figure 2.8: The puzzles from Figure 2.5 after breathing the gentle paths.

Let r be a rhombus placed with a side adjacent to a piece p on R's outer boundary, properly or not. The piece r is only adjacent to p. Otherwise, there can only be one type of puzzle piece placed next to p, contradicting that this is a spot where P_1 and P_2 differ. Therefore, at least one of the piece arrangements from Figure 2.6 can be placed next to p. An easy, though lengthy, case analysis suffices to show that, regardless of an inner boundary of R, the piece arrangements can be placed around the outer boundary of R without overlap. The minimality of R ensures that these arrangements fill R, and the bold edges form a gentle loop in P_1. □

2.3.2 Proof of The Puzzle-Based Littlewood-Richardson Rule

Our final task in this section is to prove the Littlewood-Richardson rule for puzzles:

$$c_{IJK} = \#\text{ puzzles with boundary } I, J, K \in [n]_d.$$

We begin by defining a ring Q.

Let Q be a vector space with basis q_I for $I \in [n]_d$, and let $q_I q_J = \sum_K d_{IJK^\vee}\, q_K$, where $d_{IJK^\vee}$ is the number of puzzles with boundary $I, J, K^\vee$. We first show that this is an associative ring by specifying a puzzle bijection using m-shifts. Our proof follows the outline of the proof of the Littlewood-Richardson rule for $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$ given in [5], but here, we replace the tableaux with puzzles. The techniques we use on puzzles in this proof are modifications of the techniques used on mosaics in [17], and on equivariant puzzles in [14].

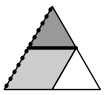
Theorem 2.3.2. *Let $I, J, K \in [n]_d$. Then*

$$\sum_{J \xrightarrow{m} \widetilde{J}} d_{I\widetilde{J}K} = \sum_{I \xrightarrow{m} \widetilde{I}} d_{\widetilde{I}JK}.$$

Proof. We show that for every puzzle with boundary $I, \widetilde{J}, K$ for some m-shift $\widetilde{J}$ of J there is a corresponding puzzle with boundary $\widetilde{I}, J, K$ for some m-shift $\widetilde{I}$ of I, and that the operation connecting the puzzles is invertible. First, consider the case where $m = 1$, so $\widetilde{J}$ is a 1-shift of J. Let P be the associated puzzle with boundary $I, \widetilde{J}, K$. There is some j_ℓ such that $\widetilde{j_\ell} = j_\ell - 1$. By definition of J, $j_{\ell+1} > j_\ell$, so the $\widetilde{j_\ell}$ edge of P is dark, and the $\widetilde{j_\ell} + 1$ edge is white. A gentle path can be started at the point s between the dark and

white edges for $\widetilde{j}_\ell$ and $\widetilde{j}_\ell + 1$. Indeed, if a gentle path can be found from s to the NW boundary of P, then breathing that gentle path will change the NE boundary to J and the NW boundary to some 1-shift of I.

Here, we construct such a path. Note that at every point g between consecutive edges of a gentle path there is at least one dark edge and one white edge meeting at g which form a line. Further, there is a unique line formed this way that is also not parallel to any of the gentle path edges meeting at g.



The diagram shows such a dark and white edge pair at a point on a gentle path running along the bold edge. The edge pair is indicated by the dotted line. That line will be denoted by L_g.

Now, start a gentle path G at the point s found above. If G has not reached the NW boundary, we add an edge to the end of G such that the end point g has L_g parallel to either the NW or NE sides of the puzzle. Figure 2.9 shows an example of a gentle path G with the point s and the lines L_g. If G hits a boundary, it must be the NW boundary, as otherwise some line L_g was parallel to the S side.

We now show that G always reaches the NW side. Let g and h be the end points of the last edge added to G, with h being the current end point of G. The lines L_g and L_h are parallel unless g is the central point of one of the diagrams below.

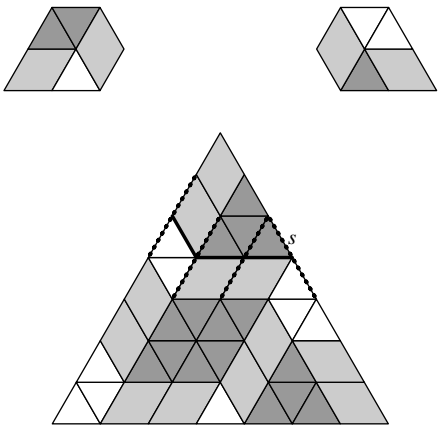


Figure 2.9: The path G in a puzzle with start point s. The lines L_g are dotted.

If the point g is at the center point of one of the two arrangements, then we need to show there is always a choice for the next point h such that L_h is parallel to the NW or NE sides. A simple case analysis shows that if L_g is parallel to the NW or NE, then there is a unique choice for h with L_h parallel to the NW or NE and the turn at g is gentle. Hence, G is the unique path from the point s on the NE side to the NW side.

Breathing the gentle path G produces a new puzzle P'. The NW boundary of P' is given by a 1-shift $\widetilde{I}$, as a dark edge on the NW is moved from above G to below G. If we build a gentle path H in P' from the NW to the NE with the same rules, then $H = G$ but with the opposite direction. To see this, note that the lines L_g mark the borders of the piece arrangements from Figure 2.6 in P, so there is a gentle path in P' that follows G in reverse. To deviate from that gentle path, H would have to hit a point h with L_h parallel to the S side. Each step in H is also unique, so this process is invertible for 1-shifts.

Now, take $m > 1$. P is again the puzzle with boundary $I, \widetilde{J}, K$ for an m-shift $\widetilde{J}$ of J. We repeat the procedure for $m = 1$ in three steps to find a succession of gentle paths, G_i, to breath and produce a puzzle with boundary $\widetilde{I}, J, K$ for some m-shift $\widetilde{I}$. First, find the least element in $\widetilde{J}$ such that $\widetilde{j}_\ell < j_\ell$. Second, find G_1 as in the $m = 1$ case. Finally, breath G_1. Repeat these three steps in the puzzle produced at the end of them until the NE boundary is J. Figure 2.10 shows the puzzle after the gentle path in Figure 2.9 is breathed. The next gentle path is bold, and the final puzzle after breathing that path is shown on the right of Figure 2.10. Between the two examples, this is an example of a 2-shift.

It is clear that the NW boundary is an m-shift of I, as there were m moves of dark edges counter clockwise. Further, a dark edge was never moved past another, as a gentle path cannot exit between dark edges. Note also that

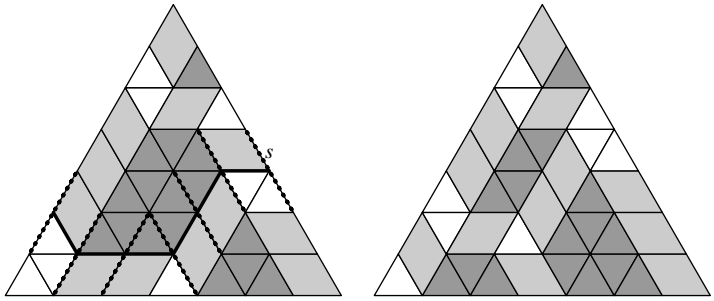


Figure 2.10: The example from Figure 2.9 continued.

G_{i+1} never intersects G_i. If it were to do so, then G_{i+1} either runs along G_i or crosses it. The dark edge of each line L_g for G_i and G_{i+1} is above the gentle path, but after breathing G_i, the dark edge is below. Hence, G_{i+1} cannot run along G_i. To cross G_i, a case analysis shows that G_{i+1} must have a point h where the line L_h is parallel to the S side, which is not possible. Therefore, successive gentle paths do not intersect. The argument in the $m = 1$ case suffices to show that each of the gentle paths G_i is unique, and the inverse process consists of finding the first dark edge clockwise on the $\widetilde{I}$ boundary with position $\widetilde{i}_\ell < i_\ell$, starting the gentle path H, and breathing H.

This constitutes a bijection between puzzles with boundary $I, \widetilde{J}, K$ and $\widetilde{I}, J, K$, which proves the result. □

We get the Pieri rule for Q as a corollary.

Corollary 2.3.3.

$$q_I q_{J_m} = q_{J_m} q_I = \sum_{I \xrightarrow{m} \widetilde{I}} q_{\widetilde{I}}.$$

Proof. Recall that J_m is the unique m-shift of $I^{\max}$. Hence,

$$d_{IJ_mK} = \sum_{I \xrightarrow{m} \widetilde{I}} d_{\widetilde{I}I^{\max}K}.$$

It is an easy matter to check that there is a unique puzzle with boundary $J, I^{\max}$, K, and that $K = J^\vee$. So we have that

$$d_{IJ_mK} = \begin{cases} 1 & I \xrightarrow{m} K^\vee \\ 0 & \text{otherwise.} \end{cases}$$

Hence, $q_I q_{J_m} = \sum_K d_{IJ_mK} q_{K^\vee} = \sum_{I \xrightarrow{m} \widetilde{I}} q_{\widetilde{I}}$.

For $q_{J_m} q_K$, we note two facts. First, $I \xrightarrow{m} K^\vee$ is equivalent to $K \xrightarrow{m} I^\vee$. Second, $d_{IJ_mK} = d_{J_mKI}$, as puzzles are rotationally symmetric. Hence,

$$q_{J_m} q_K = \sum_I d_{J_mKI} q_{I^\vee} = \sum_{K \xrightarrow{m} \widetilde{K}} q_{\widetilde{K}}.$$

□

A product with q_{J_m} is associative:

$$\begin{aligned}(q_I q_{J_m}) q_K &= \sum_L q_L \sum_{I \overset{m}{\to} \widetilde{I}} d_{\widetilde{I} K L^\vee} \\ &= \sum_L q_L \sum_{K \overset{m}{\to} \widetilde{K}} d_{I \widetilde{K} L^\vee} \\ &= q_I (q_{J_m} q_K).\end{aligned}$$

Consider the surjective linear map given by $\sigma_I \mapsto q_I$. There is a Pieri rule on both sides of the map, so Q is generated by q_{J_m}, as $H^*(\mathrm{Gr}_d(\mathbb{C}^n))$ is generated by σ_{J_m}. Hence, the product in Q is associative, so Q is a ring. It follows that the map is a surjective ring homomorphism. Therefore, the Littlewood-Richardson rule for puzzles is proven, as the structure constants are equal:

$$c_{IJK} = d_{IJK} = \text{ the number of puzzles with boundary } I, J, K.$$

Chapter 3

Combinatorics Related to Puzzles

In this chapter, we undertake a survey of some combinatorial objects related to puzzles and the Littlewood-Richardson rule. In particular, we introduce the measures developed in [2], Berenstein and Zelevinsky's hive model, and a classical skew tableau formulation of the Littlewood-Richardson rule. For each, we describe a Littlewood-Richardson rule and a bijection with puzzles. The discussion of the measures includes the development of a notion of measure rigidity, and a method for decomposing the measures via submeasures known as skeletons. For hives, we include the construction of the Horn Inequalities from [10], as it uses both hives and puzzles.

3.1 Measures in $\triangle_r$

The first combinatorial objects of our survey are the Lebesgue measures defined in [2]. These measures are supported in an equilateral triangle in the plane. We denote this triangle by $\triangle_r$, where $r \in \mathbb{N}$ is the side length. The measures are supported on the intervals between the triangular lattice points in $\triangle_r$. Figure 3.1 shows the **lattice points** of $\triangle_5$ on the left. On the right, the allowed intervals are shown by dashed lines between lattice points. These intervals are called **small edges**. We require the density of a measure on a small edge to be a non-negative integer and constant.

The measures of interest have a further requirement that the densities on the small edges meeting at a lattice point in $\triangle_r$ must obey the **balance**

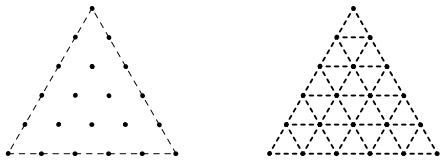


Figure 3.1: The triangle region $\triangle_5$, and the intervals in $\triangle_5$ which can receive non-zero measure.

condition. Let m be a measure in $\triangle_r$ and a, b, c, d, e, f be the six small edges about a non-boundary lattice point of $\triangle_r$, as in Figure 3.2. The balance condition at a non-boundary lattice point is obeyed if the edge densities of a, b, c, d, e, f obey the following:

$$m(a) - m(d) = m(b) - m(e) = m(c) - m(f). \tag{3.1}$$

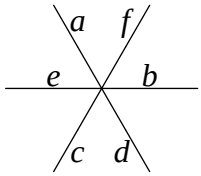


Figure 3.2: The six small edges meeting at a lattice point in $\triangle_r$.

For the balance condition at boundary points, we introduce the **exit edges** of $\triangle_r$. The exit edges of $\triangle_5$, marked a_i, b_j, c_k, are shown in Figure 3.3. The edge densities at each point on the boundary must obey the balance condition, Equation 3.1, with an additional requirement that the exit edges are the only edges outside of $\triangle_r$ that can receive non-zero density. For example, if the lattice point in Figure 3.2 is a boundary point on the NW side, then the edge e must have zero density. A boundary point is said to meet the balance condition if the small edge densities around the point obey

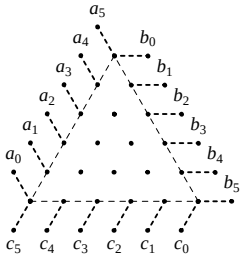


Figure 3.3: The exit edges of measures in $\triangle_5$.

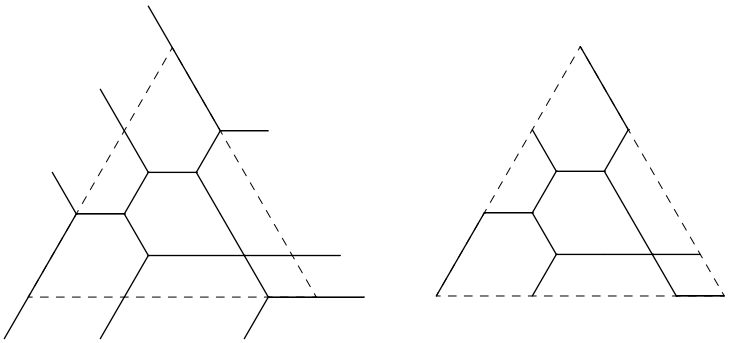


Figure 3.4: A measure in $\mathcal{M}_6$ with and without its exit edges.

Equation 3.1 and obey this additional requirement. The balance condition shows that the density of edges a, b, c, d and f obey

$$m(a) = m(b) + m(d)$$
$$m(c) = m(b) + m(f)$$

with $m(f) = 0$ at the top corner point. The other two sides have similar equations specifying the densities on small edge around the boundary points.

We denote by $\mathcal{M}_r$ the set of integer measures in $\triangle_r$ meeting the balance condition at each lattice point. Figure 3.4 shows two diagrams of a measure in $\mathcal{M}_6$. The left diagram has the exit edges displayed, and their densities are all equal to the densities of the adjacent small edges in $\triangle_6$. In future, the exit edges will be suppressed, as in the right diagram.

For $m \in \mathcal{M}_r$, we define $\mu \in \mathcal{M}_r$ to be a **submeasure** of m, denoted $\mu \leq m$, if for every small edge e in $\triangle_r$, $\mu(e) \leq m(e)$. Clearly, the support of μ is contained in the support of m, that is $\text{supp}(\mu) \subseteq \text{supp}(m)$. A measure m is called **extremal** if all the submeasures of m are multiples of m, that is, there is some constant c such that $c\mu(e) = m(e)$ for all small edges $e \in \text{supp}(m)$.

As with puzzles, we describe measures in $\mathcal{M}_r$ by their boundary, specifically, by their exit edge densities. Let $m \in \mathcal{M}_r$ and $I, J, K \in [n]_r$, and let

$$A_t = \{1 + t + M_t, 2 + t + M_t, \ldots, m(a_t) + t + M_t\},$$

where $M_t = \sum_{u=0}^{t-1} m(a_u)$ is the sum of exit densities for exit edges a_0 to a_{t-1}. We say m has NW boundary I if $I = \bigcup_{t=0}^{r} A_t$. The exit edge densities can be computed by $|A_t| = m(a_t)$. We define the sets J and K similarly for B_t and C_t. The **weight** of a measure in $\mathcal{M}_r$ is

$$\omega(m) = \sum_t |A_t| = \sum_t |B_t| = \sum_t |C_t|.$$

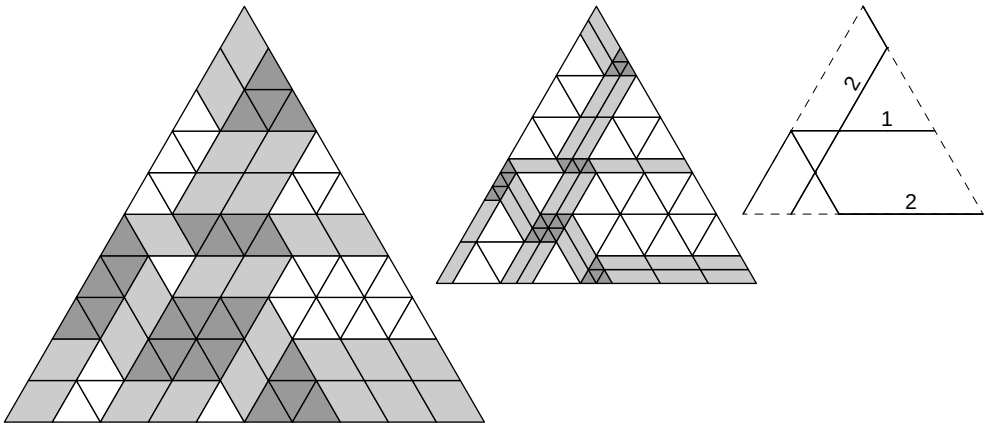


Figure 3.5: The deflation $\mathcal{D}_1(P)$ process shown in two stages.

If $I, J, K \in [n]_d$, then a measure m with that boundary has $\omega(m) = d$.

3.1.1 Puzzle Bijection

In this section, we describe the bijection used in [2] between the measures of $\mathcal{M}_r$ and puzzles. The bijection is called deflation. The process of deflation is the same as the deflation used by Knutson, Tao, and Woodward in [15] to create a bijection between puzzles and honeycombs. Except for the need to orient the honeycomb appropriately, the two deflation processes are identical. The measures described here can be viewed as the result of applying the honeycomb configuration to the honeycomb tinkertoy from [13]. For more information about how Knutson and Tao define honeycombs and their deflations, see [13] and [15]. We now describe the bijection.

Let P be a size n puzzle with d dark edges on each boundary. The **type 1 deflation** of P is denoted $\mathcal{D}_1(P)$, and it is obtained by shrinking all dark edges to length zero and dropping the boundaries on the white triangles. The density of each small edge or exit edge is given by the number of dark edges collapsed in forming that edge. The measure produced is in a triangle of size equal to the number of white edges on a boundary of P, so $\mathcal{D}_1(P) \in \mathcal{M}_{n-d}$ with $\omega\big(\mathcal{D}_1(P)\big) = d$. Figure 3.5 shows the process in three steps. The labels in the last step are edge densities that can be used to determine the densities everywhere in the measure by repeatedly applying the balance condition.

The **type 1 inflation** of $m \in \mathcal{M}_r$, denoted $\mathcal{I}_1(m)$, expands each small edge e to a number of rhombi equal to $m(e)$. The puzzle is completed by filling the holes with dark triangle pieces. It is clear that $\mathcal{I}_1(\mathcal{D}_1(P)) = P$,

and $\mathcal{D}_1(\mathcal{I}_1(m)) = m$. Also, P and $\mathcal{D}_1(P)$ have the same boundary I, J, K; the exit edges with non-zero density inflate to dark edges on P's boundary. The type 1 deflation and inflation operations constitute a bijection between size n puzzles with d dark edges per side and the measures $m \in \mathcal{M}_{n-d}$ with $\omega(m) = d$. The Littlewood-Richardson rule for measures follows.

Littlewood-Richardson rule 2. *Let $I, J, K \in [n]_d$. Then*

$$c_{IJK} = \#m \in \mathcal{M}_{n-d} \text{ with boundary } I, J, K.$$

There is another pair of operations similar to the type 1 operations where the roles of the dark and white edges are exchanged. To define these, we first take a moment to discuss the set of *dual measures*.

Let $\mathcal{M}_s^*$ be the set of measures in the triangle $\triangle_s$ that obey the balance condition at every lattice point in $\triangle_s$ but have the exit edges as in Figure 3.6. The boundary of dual measures in $\mathcal{M}_s^*$ is defined similarly to the boundary of measures in $\mathcal{M}_r$. Note that for $m \in \mathcal{M}_r$, the reflection of m about the center vertical line is a dual measure in $\mathcal{M}_r^*$. To differentiate, the diagrams of measures in $\mathcal{M}_r$ are drawn with a white background, and the dual measures in $\mathcal{M}_s^*$ are drawn with a dark gray background.

The dual measures allow us to define the type 0 deflation and inflation operations. The type 0 deflation, $\mathcal{D}_0(P)$, of a puzzle P is similar to the type 1 deflation save that it is the white edges that are shrunk to zero length. This creates a measure $\mathcal{D}_0(P) = m^* \in \mathcal{M}_d^*$. The type 0 inflation of $m^* \in \mathcal{M}_d^*$, $\mathcal{I}_0(m^*)$, is similar to $\mathcal{I}_1(P)$ but the rhombi created are inflated along their white edges and the holes created are filled with white triangles.

Recall the definition of the sets $I^*, J^*, K^* \in [n]_{n-d}$ from Section 1.2.2. The measure $m^* \in \mathcal{M}_d^*$ is dual to the measure $m \in \mathcal{M}_{n-d}$ in the following

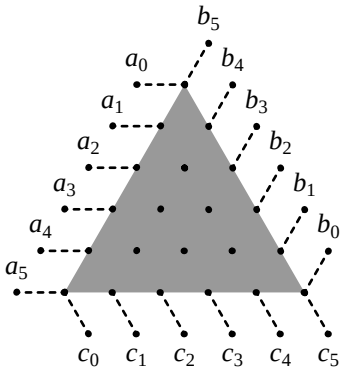


Figure 3.6: The exit edges of dual measures.

way: if m has boundary $I, J, K \in [n]_d$, then the dual measure of m is

$$m^* = \mathcal{D}_0(\mathcal{I}_1(m)),$$

and m^* has boundary $I^*, J^*, K^* \in [n]_{n-d}$. As noted, $\mathcal{D}_1(P) \in \mathcal{M}_{n-d}$ with weight d, while its dual $\mathcal{D}_0(P) \in \mathcal{M}_d^*$ has weight $n - d$. This duality makes it clear that $c_{IJK} = c_{I^*J^*K^*}$.

3.1.2 Evil Loops

The bijection with puzzles in Section 3.1.1 gives a natural idea of measure rigidity: the measure $m \in \mathcal{M}_r$ is rigid if and only if the puzzle $\mathcal{I}_1(m)$ is rigid. In [2], the authors define measure rigidity this way, and then give a characterization of measure rigidity wholly in terms of the measures using what are called evil loops. Evil loops are similar to gentle loops in puzzles in that evil loops are composed of evil turns, as gentle loops are gentle turns. We now state the definition of an evil loop from [2].

Definition 3.1.1. *Let e and f be adjacent small edges in the support of $m \in \mathcal{M}_r$ arranged as shown in one of the five diagrams in Figure 3.7. The turn from e to f is **evil** if supp(m) contains the small edges shown in that diagram, and possibly others. An **evil loop** in a measure is a sequence of evil turns that start and end with the same small edge.*

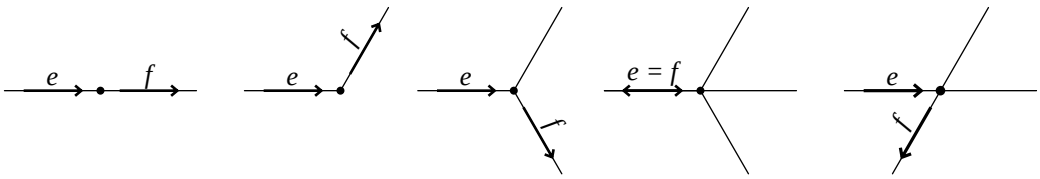


Figure 3.7: The five evil turns from e to f.

We define measure rigidity as follows. The measure m is **rigid** if m does not contain an evil loop or a lattice point with all six adjacent small edges in supp(m). An example of an evil loop is given in Figure 3.8. It is pictured along with another measure, which is obtained by an operation similar to breathing a gentle loop. Both measures have the same boundary I, J, K, so $c_{IJK} \geq 2$.

We have two remarks about this definition. First, the evil turns in dual measures reverse the arrows of the evil turns pictured in Figure 3.7, so for

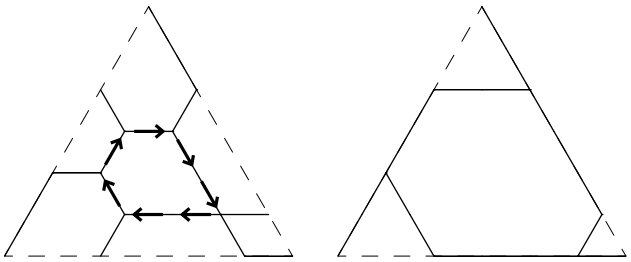


Figure 3.8: An evil loop, and another measure with the same boundary.

a dual measure, the evil turn can be viewed as going from the edge f to e. Second, for a measure $m' \in \mathcal{M}_r$ with $m' \leq m$, if m is rigid, then m' is also rigid, as an evil loop in m' gives an evil loop in m.

3.1.3 Skeletons

We now turn to the decomposition of rigid measures by submeasures supported on skeletons. Let $m \in \mathcal{M}_r$. The support of m is a **skeleton** if m is an extremal measure. If m is rigid and not extremal, then $\text{supp}(m)$ can be decomposed into skeletons. Figure 3.9 shows a measure with the three skeletons that compose it. The aim of this section is to describe the decomposition method in [2].

We say that a small edge $f \in \text{supp}(m)$ is a **descendant** of e, denoted $e \to f$, if e and f are adjacent in $\text{supp}(m)$ and, up to rotation or reflection, are arranged as in Figure 3.10. For $e \to f$, the solid edges must be in $\text{supp}(m)$, and at least one of the dotted edges must not be in supp(m). The small edge e is the **ancestor** of f. A **descendance path** from e to f is a sequence of small edges, $e = e_1, e_2, \ldots, e_m = f$, such that

$$e = e_1 \to e_2 \to \cdots \to e_m = f.$$

Note that if $e \to f$, the turn from e to f is evil, as both descendance relation-

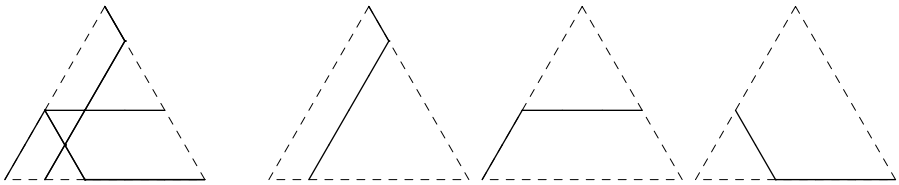


Figure 3.9: The measure from Figure 3.5, and the three skeletons composing it.

Figure 3.10: $e \to f$ if the solid edges are in $\text{supp}(m)$ and at least one of the dotted lines is not.

ships shown in Figure 3.10 are evil turns. Hence, if there is a descendance path from e to f and a different descendance path from f to e, then m has an evil loop.

To track the descendants, we mark the small edges of a measure with the arrows in Figure 3.10 where the arrow points from an ancestor edge to a descendant. Let v be a lattice point in $\triangle_r$, and let E be the set of small edges incident to v that are marked with an arrow pointing at v. We say v is **descendance marked** if for every $e \in E$, all the descendants of e incident to v are marked with an arrow pointing away from v. Note that we do allow a small edge to be marked with arrows of opposite direction. A directed path along such arrows is a descendance path.

A **descendance set** is the set of all descendants of a particular small edge. The descendance set for a small edge e is denoted S_e. The set S_e can be found by marking the descendants of e with arrows pointing away from e, and then repeatedly descendance marking any lattice points that are incident to arrows which are not yet descendance marked. During this process, we never place an arrow on a small edge f such that the new arrow has the same direction as an already existing arrow on f; as such, the marking process cannot cycle. An example of a descendance set marked in this way is shown in Figure 3.11.

Our first task toward the decomposition of m is to show that the density increases along descendance paths. Let $e \to f$. We show that either $m(e) = m(f)$ or $m(e) < m(f)$. The cases when $m(e) = m(f)$ are pictured in Figure

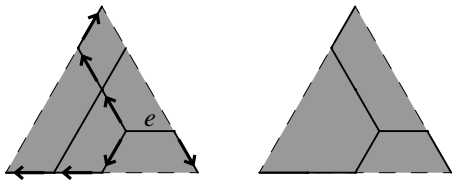


Figure 3.11: The marked descendants of an edge e.

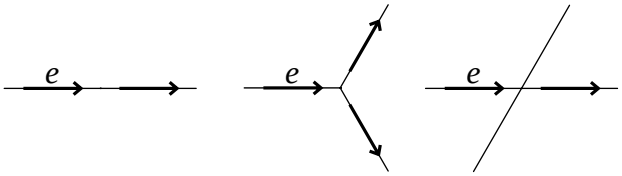


Figure 3.12: Three descendance marked points where the densities of all arrow marked edges is equal.

3.12. The balance condition, Equation 3.1, requires that $m(e) = m(f)$ in each case. Note that regardless of which descendant is designated f, we also have $f \to e$.

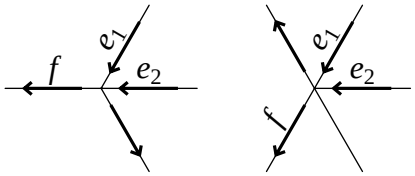


Figure 3.13: The two ways a descendant, f, can have greater density than an ancestor, e_1 and e_2.

The two lattice points shown in Figure 3.13 have $m(e) < m(f)$ for $e = e_1$ or $e = e_2$. This is because in both cases, $m(f) = m(e_1) + m(e_2)$, so that $m(f)$ is strictly greater than either.

Figure 3.14 shows lattice points for which an ancestor edge could have no descendants. Together, Figures 3.12, 3.13, and 3.14 show all the ways an arrow marked edge e can be adjacent to a lattice point. Since $e \to f$, the small edge e cannot be arrow marked as in Figure 3.14. Hence, the edge e must be the arrow marked edge in a case of Figure 3.12 or e is one of e_1 or e_2 in a case of Figure 3.13. The paragraphs above then establish that $m(e) \leq m(f)$. Further, the only lattice points where the density along a descendance path can increase are those in Figure 3.13.

Now, let $m \in \mathcal{M}_r$ be rigid. Let e have minimal density in m, and find S_e using arrow marks. We aim to define a non-zero measure in $\mathcal{M}_r$ supported

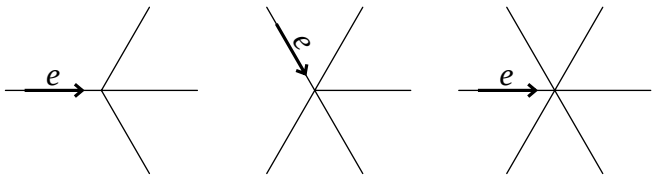


Figure 3.14: The edge e has no descendants in the direction of the arrow.

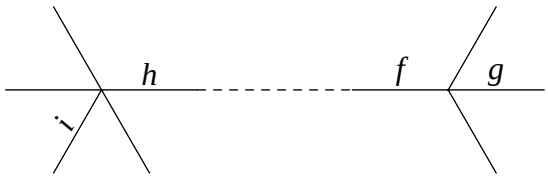


Figure 3.15: The descendance path from h to f.

by the skeleton S_e. To do this, we must ensure that it is possible for the balance condition to be met at each lattice point in $\triangle_r$. If a lattice point in $\triangle_r$ was marked as in Figure 3.14, the balance condition cannot be met at that lattice point, as there is a single small edge included in S_e at that point. We show there can be no lattice point marked as in Figure 3.14 if m is rigid.

Suppose otherwise. Let f be the arrow marked edge as in one of the diagrams in Figure 3.14. The small edge f cannot be incident to a lattice point where six edges meet, as m is rigid. In the other two cases of the figure, f is adjacent to an edge g as in Figure 3.13 where g plays the role of either e_1 or e_2. Hence, $m(f) > m(g)$. Figure 3.15 shows an example of the lattice point where f and g meet.

Since e has minimal density, we have that $m(f) > m(g) \geq m(e)$; therefore, the densities along the descendance path from e to f must have increased. The lattice points pictured in Figure 3.13 are the only points where a descendance path increases in density. Hence, there is a pair of edges h and i in the descendance path where the density strictly increased from i to h. Figure 3.15 shows an example of the lattice point where h and i meet. Further, we can assume without loss of generality that $m(h) = m(f)$.

The small edges f and h are both on the descendance path from e to f, so there is a descendance path from h to f. Figure 3.15 shows this as a dashed line. Since $m(h) = m(f)$, every edge along the path between h and f has the same density. In particular, each turn along that path is evil in both directions, as per the argument made for Figure 3.12. We note that there is an evil turn at the end points, and the evil turn in each case is the fourth evil turn pictured in Figure 3.7, which uses a single small edge to reverse direction. Hence, there is an evil loop in m, contradicting the rigidity of m.

We can now define a measure μ_e in $\triangle_r$ that is supported on S_e. First, set the density of e to be $\mu_e(e) = m(e)$. Then follow the arrows outward from e, increasing the density of small edge f by $m(e)$ each time f is encountered. Figure 3.11 shows an example of S_e in a measure, with the measure μ_e on the right.

We note some facts about the measure μ_e. Firstly, the process of building μ_e does not cycle; otherwise, there are two different descendance paths from two antipodal edges on that cycle, which forms an evil loop. Secondly, $\mu_e \in \mathcal{M}_r$. To see this, we need only check the balance condition at each lattice point. To that end, we keep the arrows of S_e on the measure μ_e. Each lattice point where small edges of μ_e meet is descendance marked, so the procedure outlined above ensures that the balance condition is met at each lattice point. Thirdly, S_e is indeed a skeleton as the measure created by removing a proper subset of small edges from μ_e has at least one lattice point that is not descendance marked, that is at least one lattice point where the balance condition cannot be met. Hence, μ_e is extremal. The third point is in fact enough to show that a descendance set S_e that is without a lattice point marked as in Figure 3.14 is a skeleton.

The measure m is decomposed as follows. Let e be a minimal density edge in m. Find S_e by arrow marking as above, and form the measure μ_e. Then, we do the following:

1. find the minimal edge e' in the measure $m' = m - \mu_e$ that is not arrow marked in m,
2. construct $S_{e'}$,
3. add the arrow marks made in m' for $S_{e'}$ to the original measure m,
4. construct $\mu_{e'}$, and
5. form $m'' = m' - \mu_{e'}$.

Repeat this process for the minimal edge $e^{(t)}$ and $S_{e^{(t)}}$ in the measure $m^{(t)}$ until

$$m^{(t)} = m - \sum_i \mu_{e^{(i)}} = 0.$$

We now prove that this decomposition is correct. First, it is easily checked that every measure in sight is a measure in $\mathcal{M}_r$. Second, consider the measure $m' = m - \mu_e \in \mathcal{M}_r$. To construct the next measure m'', we must be able to find an edge e' that has minimal density in m' and is not arrow marked in m. Suppose to the contrary. Then every small edge in m' is arrow marked by S_e in m. Thus, every edge of m' is a descendant of e, and therefore part of S_e. The construction of μ_e and the balance condition, Equation 3.1, ensure that in this case $m' = 0$. Hence, either $m' = 0$ or there is a minimal density edge

e' in m' such that e' is not arrow marked in m. We can repeat this argument in general to show that, either there is a minimal density small edge $e^{(t)}$ in $m^{(t)}$ that is not arrow marked in m, or

$$m^{(t)} = m - \sum_i \mu_{e^{(i)}} = 0.$$

Hence, the process ends with

$$m = \sum_i \mu_{e^{(i)}},$$

and this decomposition is unique up to reordering the sum, as none of the edges $e^{(i)}$ are descendants of any other such edge $e^{(j)}$.

3.2 Hives

We continue our survey with a discussion of hives. Hives are a combinatorial device created by Berenstein and Zelevinsky in [3] that are closely related to honeycombs and puzzles, defined in [13] and [15] respectively. Interestingly, hives did not appear in either of the latter papers. Buch, in [4], gave a proof of the saturation conjecture, the main result of [13], using hives in place of honeycombs. Hives also play a significant role in [10], where a hive-based description of the Horn inequalities was exploited to factor Littlewood-Richardson coefficients.

Our definition of hives follows the edge labelling definition given in [10]. We begin, ostensibly, with a puzzle consisting entirely of white triangle pieces.

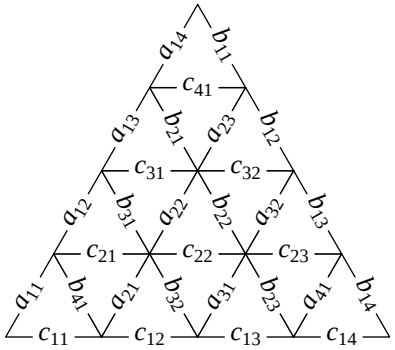


Figure 3.16: A hive diagram with labels a_{ij}, b_{ij}, c_{ij}.

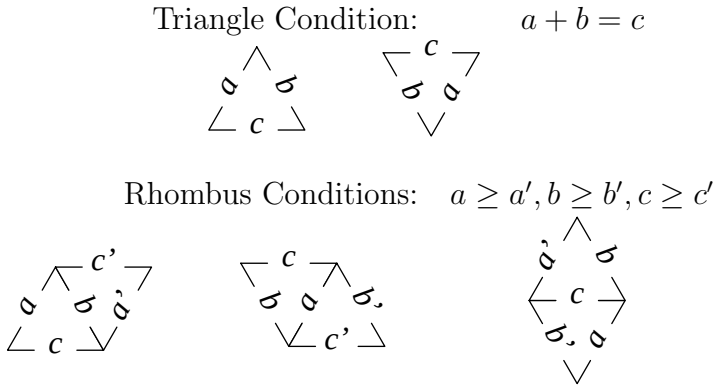


Figure 3.17: The hive conditions.

The edges are labelled with non-negative real numbers, as is shown in Figure 3.16. The result is a **hive** if the labels obey the following two requirements. First, let the labels a, b, c be the labels around a triangle as in the top portion of Figure 3.17. The labels a, b, c must obey the **triangle condition**, that is $a + b = c$. Second, suppose R is a rhombus formed of two triangles, and let the labels around R be as in one of the cases in the bottom portion of Figure 3.17. Each pair of labels on parallel edges around R must obey one of the inequalities $a \geq a', b \geq b', c \geq c'$. For example, the labels a, a', c, c' in the first case of Figure 3.17 must obey $a \geq a', c \geq c'$. If the labels around R obey these inequalities, we say R obeys one of the three **rhombus conditions**. A hive is **integer** if all the labels are non-negative integers.

The conditions on the labels ensure that the boundary of an integer hive is given by the d-part partitions $\nu = (a_{11}, \ldots, a_{1d})$, $\mu = (b_{11}, \ldots, b_{1d})$, and $\lambda = (c_{11}, \ldots, c_{1d})$. For instance, ν is a partition, as the first and third rhombus conditions show that the NW labels obey $a_{1i} \geq a_{1(i+1)}$. We can now state the Littlewood-Richardson rule for hives.

Littlewood-Richardson rule 3. *Let ν, μ, and λ be d-part partitions. Then*

$$c_{\nu\mu}^{\lambda} = \#\ \textit{hives with boundary } \nu, \mu, \lambda.$$

Figure 3.18 shows an integer hive with boundary $\nu = (3, 3, 3, 0, 0)$, $\mu = (4, 4, 2, 0, 0)$, $\lambda = (5, 4, 4, 3, 3)$, so $c_{\nu\mu}^{\lambda} \geq 1$.

3.2.1 A Puzzle Bijection

We now aim to show that hives are in bijection with puzzles defined in Section 1.2. We do this by showing a bijection to the dual measures of Section 3.1,

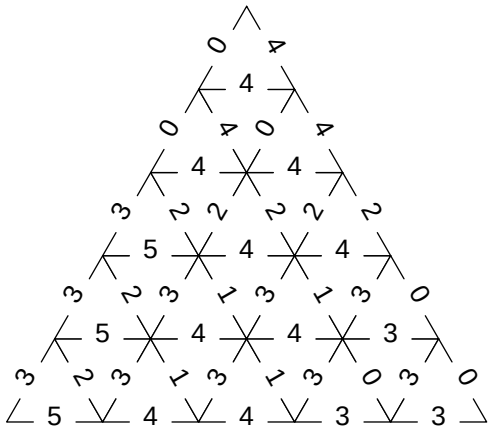


Figure 3.18: An integer hive.

and then apply inflation to get a puzzle with the same boundary, that is the puzzle has boundary I, J, K and the hive has boundary $\lambda(I), \lambda(J), \lambda(K^\vee)$.

Choose a hive h with boundary ν, μ, λ. Let d be the size of h, and choose n so that ν, μ, λ fit in an $n-d$ by d box. Define a measure m^* in $\triangle_d$ as follows. Give the label $n-d$ to every exit edge on the NW side, and 0 to every other exit edge. Place an edge in between the ends of successive exit edges, and label it with the boundary label of h found between the same two exit edges. Figure 3.19 shows how the labels are placed around the region $\triangle_d$. Densities are assigned as follows. The first exit edge on the NW, NE and S sides is assigned density $m(a_0) = a_{1d}$, $m(b_0) = b_{1d}$ and $m(c_0) = c_{11}$ respectively. The final exit edges have densities $m(a_d) = n - d - a_{11}$, $m(b_d) = n - d - b_{11}$, and $m(c_d) = n - d - c_{1d}$ respectively. Every other small edge e in $\triangle_d$ and exit edge is assigned density as in Figure 3.20 and its rotations; that is, if we treat e as the edge through the middle of a rhombus, $m(e)$ is the absolute difference of either the b sides or the a sides.

Figure 3.21 shows an example of this process applied to the hive of Figure 3.18 with $n = 10$ and $n - d = 5$. The darker lines in the hive itself show the relative density of the measure edges; the thickest have density 3, then 2 and finally 1. The thinnest hive edges have density 0. On the right, the measure in $\mathcal{M}_s^*$ is shown as normal, that is at the normal size, without exit edges, and without differentiated thickness of small edges.

We now show that the dual measures created from a hive by the procedure above are indeed in $\mathcal{M}_d^*$ by demonstrating that the balance condition is met at each lattice point. Consider Figure 3.22. We refer to the small edge

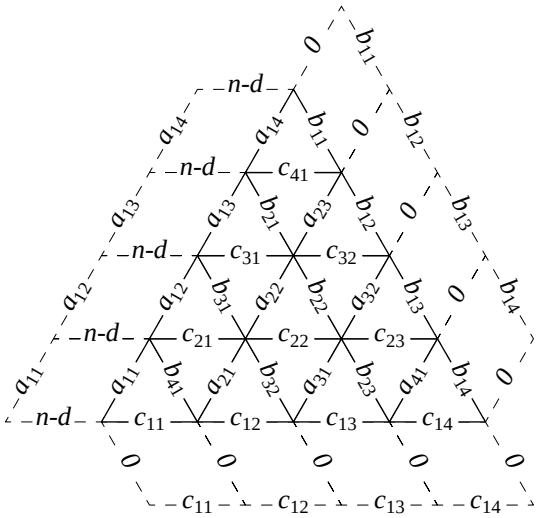


Figure 3.19: The labels placed around $\triangle_d$.

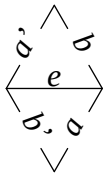


Figure 3.20: $m(e) = a - a' = b - b'$.

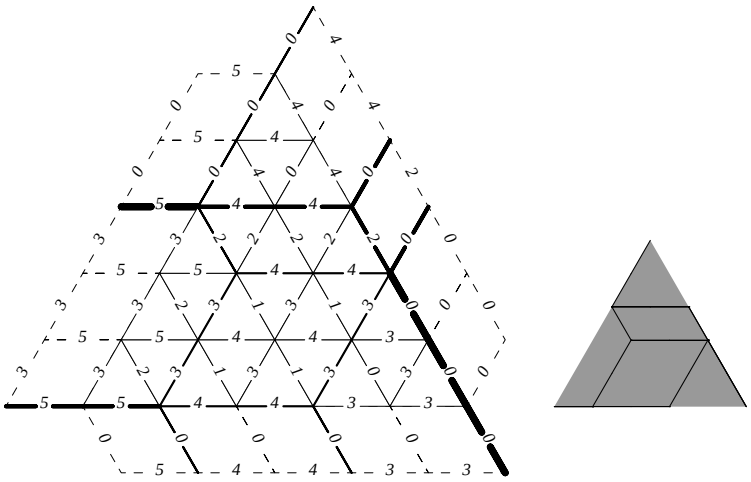


Figure 3.21: The measure obtained from Figure 3.18 both in the hive and as a dual measure on the right.

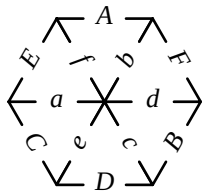


Figure 3.22: Hive labels.

of m by its label from the hive, so the edge labeled x has density $m(x)$. The balance condition is the following condition on the small edge densities meeting at the central point of Figure 3.22:

$$m(a) - m(d) = m(b) - m(e) = m(c) - m(f).$$

The measure density, $m(x)$, for an edge x is given by a difference of hive edge labels, as in Figure 3.20. We have that

$$m(a) - m(d) = (e - E) - (B - b), \tag{3.2}$$
$$m(b) - m(e) = (f - F) - (C - c), \tag{3.3}$$
$$m(c) - m(f) = (D - d) - (a - A). \tag{3.4}$$

The hive labels obey the triangle condition from Figure 3.17, and that condition shows that Equations 3.2, 3.3, and 3.4 do indeed hold with equality. For instance, the triangle condition for the triangle labelled E, f, a is $E + f = a$, and for the triangle labelled C, e, a, the condition is given by $C + e = a$. Hence, $E + f = a = C + e$. Similarly, we have $b + F = d = c + B$. Rearranging and summing these equations gives $(e - E) - (B - b) = (f - F) - (C - c)$, which shows $m(a) - m(d) = m(b) - m(e)$. The equations show that given a dual measure, the edge labels of a hive can be calculated uniquely, so a unique hive can be created from a dual measure. Hence, the procedure above is a bijection.

We then have a bijection between hives and puzzles using the type 0 deflation and inflation operations of Subsection 3.1.1. Further, it is an easy matter to check that if the puzzle P has boundary I, J, K, then the hive obtained by this bijection has boundary $\lambda(I), \lambda(J), \lambda(K^\vee)$, as we would expect.

3.2.2 Horn Inequalities

The Horn inequalities arise when determining conditions on the eigenvalues of $d \times d$ Hermitian matrices A, B, C satisfying $A + B = C$. These inequalities

are closely tied to the Schubert calculus of the Grassmanian in that the sets I, J, K determine a Horn inequality when the Littlewood-Richardson number $c_{IJK} > 0$. The Horn inequalities are related to several other areas of mathematics including symplectic geometry and invariant theory; the surveys [7] and [9] by Fulton examine these connections. In this section, we describe the construction of Horn inequalities presented in [10], where a puzzle with boundary I, J, K is used to build the Horn inequality for I, J, K. We begin with the necessary definitions.

Denote the eigenvalues of the matrix X by $\lambda_1(X) \geq \lambda_2(X) \geq \cdots \geq \lambda_d(X)$. Let (I, J, K) be a triple of sets from $[d]_s$ for some $s \leq d$ such that $c_{IJK} > 0$. Then we say (I, J, K) is a **Horn triple** and

$$\sum_{k \in K^\vee} \lambda_k(C) \leq \sum_{i \in I} \lambda_i(A) + \sum_{j \in J} \lambda_j(B), \tag{3.5}$$

is the **Horn inequality** for (I, J, K). These inequalities characterize the possible eigenvalues of Hermitian $d \times d$ matrices A, B, C for which $A + B = C$.

The Horn inequalities define a cone in $\mathbb{R}^{3d}$, and the facets of this cone are given by the **essential Horn inequalities**. Not every Horn inequality is essential. The essential Horn inequalities have the following characterization due to Belkale [1] (forward) and Knutson, Tao and Woodward [15] (backward):

The Horn inequality for (I, J, K) is essential if and only if $c_{IJK} = 1$.

Schubert calculus and the Littlewood-Richardson numbers are related to the Horn inequalities by the following theorem, known as the **Horn recursion**.

Theorem 3.2.1. *Let ν, μ, λ be partitions of length at most d. Then $c_{\nu\mu}^{\lambda} > 0$ if and only if $|\lambda| = |\nu| + |\mu|$ and the Horn inequalities*

$$\sum_{k \in K^\vee} \lambda_k \leq \sum_{i \in I} \nu_i + \sum_{j \in J} \mu_j,$$

are satisfied for all Horn triples (I, J, K) for $I, J, K \in [d]_s$ for all $s \leq d$.

Theorem 3.2.1 is deducible using the result of [22] along with the main result of [12] and the saturation theorem of [13].

Horn's inequalities can be determined by a construction using hives and puzzles [10], which we describe now. First, take d to be the size of our hive

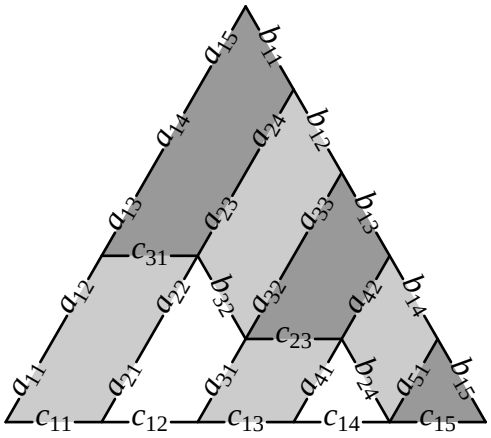


Figure 3.23: A puzzle P with a hive superimposed with $d = 5$ and $s = 3$.

with boundary ν, μ, λ, and choose a Horn triple $(I, J, K) \in [d]_s$ for some $s \leq d$. The triple has at least one puzzle P with boundary given by (I, J, K), as $c_{IJK} > 0$. Superimpose the hive on P and suppress all edge labels within puzzle piece regions, as in Figure 3.23. The Horn inequality can be written using the hive edge labels:

$$\sum_{k \in K^\vee} c_{1k} \leq \sum_{i \in I} a_{1i} + \sum_{j \in J} b_{1j}.$$

This inequality can be obtained from the hive overlay in the following way. For each dark region, find the equation determined by the triangle condition for the region's hive labels. The rhombus conditions are used similarly to determine inequalities for the hive labels around a rhombus region. The Horn inequality for I, J, K is found by using the relations found to relate the sum $\sum_{k \in K^\vee} c_{1k}$ with the sums $\leq \sum_{i \in I} a_{1i}$, and $\sum_{j \in J} b_{1j}$.

As an example of this, we construct the Horn inequality for Figure 3.23. We first apply the triangle condition to each dark region Figure 3.23, and we get the following equations for the hive labels around the three dark regions:

$$\begin{aligned} c_{31} + a_{23} + a_{24} &= b_{11} + a_{13} + a_{14} + a_{15}, \\ c_{23} + a_{42} &= b_{13} + a_{32} + a_{33}, \\ c_{15} &= a_{51} + a_{15}. \end{aligned}$$

We then apply the the three rhombus conditions on the rhombus regions to produce inequalities for the hive labels around each of the gray rhombus

regions; there are four such inequalities:

$$c_1 1 \geq c_{31},\ a_{33} \geq a_{24},\ a_{32} \geq a_{23},\ a_{51} \geq a_{42}.$$

Combining the three equalities and the four inequalities for Figure 3.23, we have the following computation:

$$\begin{aligned} c_{11} + c_{13} + c_{15} &= c_{31} + c_{23} + (a_{51} + b_{15}) \\ &\geq c_{31} + (c_{23} + a_{42}) + b_{15} \\ &= c_{31} + (b_{13} + a_{32} + a_{33}) + b_{15} \\ &\geq (c_{31} + a_{23} + a_{24}) + b_{13} + b_{15} \\ &= (a_{13} + a_{14} + a_{15}) + (b_{11} + b_{13} + b_{15}). \end{aligned}$$

This is the Horn inequality for $I = \{3, 4, 5\}$, $J = \{1, 3, 5\}$ and $K^\vee = \{1, 3, 5\} = K$. In a similar fashion, the white regions can be used to give the Horn inequality for I^*, J^*, K^*:

$$c_{12} + c_{14} \leq (a_{11} + a_{12}) + (b_{12} + b_{14}).$$

Note that the puzzle above is rigid, so $c_{IJK} = 1$. That is the triple I, J, K is essential.

3.3 Tableaux

There are many Littlewood-Richardson rules given by a combinatorial constructions using tableaux. In this section, we describe a classical example using skew tableaux of shape λ/ν with content μ. This rule first appeared in [16] without proof, and it was later proved by Schützenberger in [20]. A concise proof for this version of the Littlewood-Richardson rule can be found in [21]. The interested reader is directed to [6] for additional examples of Littlewood-Richardson rules given in terms of tableaux.

The Littlewood-Richardson rule we present uses semistandard Young tableau. A **semistandard Young tableau** (SSYT) of **shape** λ is a filling of the boxes of λ's diagram with numbers such that the columns are strictly increasing downward, and the columns weakly decreasing to the right. Let t be a SSYT of shape λ. We say t has **content** $\mu \vdash m$ if the number of times i appears in t is equal to μ_i. The shape of a SSYT can also be given by a

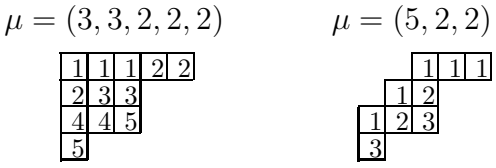


Figure 3.24: Semistandard tableau for a partition and skew partition with content μ.

skew partition λ/ν. An example of a SSYT of shape λ and of shape λ/ν is shown in Figure 3.24.

Suppose t is a SSYT of shape λ/ν with content μ. The **word** of t, $w(t)$, is the word created from the entries of t read right to left, top to bottom. We say t is a **Littlewood-Richardson tableau** if $w(t)$ obeys the following condition denoted $(*)$:

$(*)$: In any initial subword of $w(t), i$ appears at least as often as $i+1$.

The skew tableau of Figure 3.24 has word 111213213, which obeys $(*)$. We state the Littlewood-Richardson rule for these tableaux.

Littlewood-Richardson rule 4. *Let ν, μ, λ be partitions. Then*

$$c_{\nu\mu}^{\lambda} = \#\textit{Littlewood-Richardson tableaux with shape } \lambda/\nu \textit{ and content } \mu.$$

We finish this chapter with a bijection between puzzles and these tableaux. It comes from [10], and it is in the form of a bijection to hives. The bijection in Section 3.2.1 then establishes the bijection to puzzles. Let t be a tableau of shape λ/ν with content μ which fits in an $n-d$ by d box. Fill the positions of ν in the diagram of λ/ν with zeros. Let h a hive of size d with edge labels given by the following:

$$a_{ij} = \#\{ \text{ entries } \leq i-1 \text{ in row } (i+j-1) \text{ of } t\} \tag{3.6}$$

$$b_{ij} = \#\{ \text{ entries } = j \text{ in 1st } (d+1-i) \text{ rows of } t\} \tag{3.7}$$

$$c_{ij} = \begin{pmatrix} \#\{ \text{ entries } = j \text{ in 1st } (i+j-2) \text{ rows of } t\}+ \\ \#\{ \text{ entries } \leq j \text{ in row } (i+j-1) \text{ of } t\}. \end{pmatrix} \tag{3.8}$$

We show that h is indeed a hive, and thus, the bijection of Section 3.2.1 may be applied to h to form a puzzle.

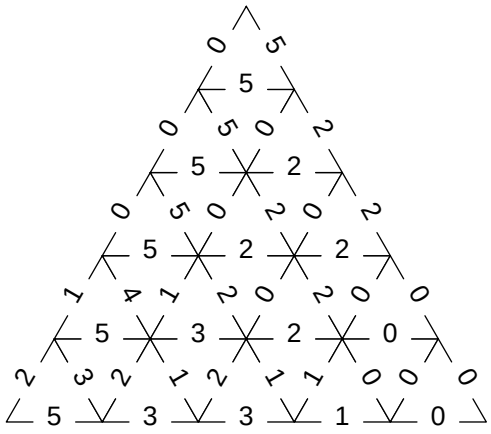


Figure 3.25: The hive for the tableau in Figure 3.24.

We first establish that the hive labels a_{ij} obey the rhombus conditions, which are defined by Figure 3.17. Checking the rhombus conditions for a_{ij} amounts to checking that

$$a_{ij} \geq a_{(i+1)j} \geq a_{i(j+1)}$$

for a tableau obeying $(*)$. The i entries in row $i+j$ of t are positioned weakly to the left of the furthest right $i-1$ entry in row $i+j-1$, so $a_{ij} \geq a_{(i+1)j}$. The definition of a_{ij} is enough to show that $a_{(i+1)j} \geq a_{i(j+1)}$. The rhombus conditions for the labels b_{ij} and c_{ij} are shown by similar observations about t.

The triangle condition for labels around a triangle in h is

$$c_{ij} = a_{ji} + b_{(d+2-i-j)j},$$

if the triangle is rightside up. If the triangle is upside down, the condition on the labels is

$$c_{ij} = a_{(j+1)(i-1)} + b_{(d+3-i-j)j}.$$

Both equations are established by substituting the indices in Equations 3.6, 3.7, and 3.8 followed by some trivial manipulation.

The hive in Figure 3.25 is the hive for the Littlewood-Richardson tableau in Figure 3.24. For instance, the hive label c_{32} in Figure 3.25 is equal to 2, and this agrees with Equation 3.8; c_{32} is the number of 2s in the first three rows plus the number of entries less than 3 in row 4, which is $2+0$. The hive labels can be used to reconstruct a unique tableau using Equations 3.6, 3.7,

and 3.8. By way of example, the number of $i > 0$ that appear in row k of the tableau t is $a_{(i+1)(k-i)} - a_{i(k-i+1)}$, and a_{1k} is the number of zeros in row k.

The triangle condition and the rhombus conditions on the hive labels then guarantee that the tableau constructed is a unique Littlewood-Richardson tableau. Hence, the above describes a bijection between Littlewood-Richardson tableau and hives, so by the bijection in Section 3.2.1, there is a bijection between Littlewood-Richardson tableaux and puzzles.

Chapter 4

Main Results

In this chapter, we describe an algorithm which is run on a rigid puzzle P with boundary $I, J, K \in [n]_d$. The algorithm creates a lattice polynomial describing the single subspace in the triple intersection of Schubert varieties.

$$\Omega_I(F_\bullet) \cap \Omega_J(G_\bullet) \cap \Omega_K(H_\bullet).$$

The chapter is divided up into three sections each describing a major part of the algorithm. The first systematically marks the pieces of P, the second searches for a set of marks obeying some conditions, and the third uses the marked puzzle to build the polynomial recursively.

Now, if P is non-rigid, then $c_{IJK} > 1$, by the puzzle-based Littlewood-Richardson rule. In that case, Theorem 2.1.4 shows that synthesis is impossible. Synthesis is the purpose of our algorithm, so our algorithm must avoid attempting to compute a lattice polynomial using a non-rigid puzzle as input. To this end, the algorithms are defined such that non-rigidity can be detected. We begin with the description of how P is marked.

4.1 Walkers

We show a partially marked puzzle in Figure 4.1, and give a brief preview of how it is marked. A **walker** is a double arrow symbol on a puzzle piece. The double arrow in Figure 4.1 is an example of how a walker can appear in a puzzle. The marking procedure is carried out by having walkers move about a puzzle taking *steps*, and leaving behind a mark on the vacated puzzle pieces. The marks are either a dot, or a single arrow. Figure 4.1 has examples

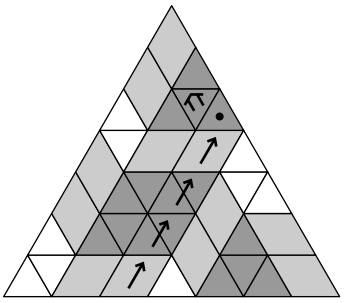


Figure 4.1: A marked puzzle mid-algorithm.

of how these marks can appear in a puzzle. We call the dot a **branch mark** and the single arrow is called a **rhombus mark**.

Subsection 4.1.1 defines the walker steps and some of the necessary terminology. We end that subsection with an example of how a puzzle is marked. Subsection 4.1.2 defines some terms for describing the regions of a puzzle, and introduces a final rule restricting walkers. The remainder of Section 4.1 is devoted to proving that the basic marking procedure terminates and that the marks achieve their aim: to mark a set of pieces in P which deflate to the edges of a descendance set.

4.1.1 Walker Steps

We begin with the rules for placing the initial walker in an unmarked puzzle. Choose a rhombus piece, and place the **initial walker** w so it is parallel to an edge of that rhombus. If the edge it is parallel to is white, we require w to be placed **rightmost**, where rightmost means every rhombus to the right of w has a rhombus mark. If the edge is dark, then w must be placed **leftmost**, which is similar to rightmost. These placement rules will suffice for now, but later, additional rules will be introduced that further restrict the placement of an initial walker.

After the initial placement, w begins to take its steps. If a walker is outside the puzzle after any step, the walker is removed. The basic step is called a **rhombus step**. These steps consist of making a rhombus mark on the piece occupied by the walker. The mark can be made on a pair of unmarked triangles or an unmarked rhombus. The walker is then placed in front of the rhombus mark just made. The rhombus steps are shown in Figure 4.2. The any rhombus marked pair of triangles or rhombi is referred to as marked rhombi.

Figure 4.2: The rhombus steps.

The **branch steps** are taken if a rhombus step is not possible. This occurs when the walker is on an unmarked triangle piece which cannot be matched to another in the direction of the walker. Figure 4.3 shows the cases in which a match cannot be made, and the result after the branch step is taken. The walker taking the branch step, w, is removed, and two new walkers, called the **children** of w, are placed. If w is an initial walker in P, any other walker in P is said to be a **descendant** of w.

Figure 4.3: The branch steps.

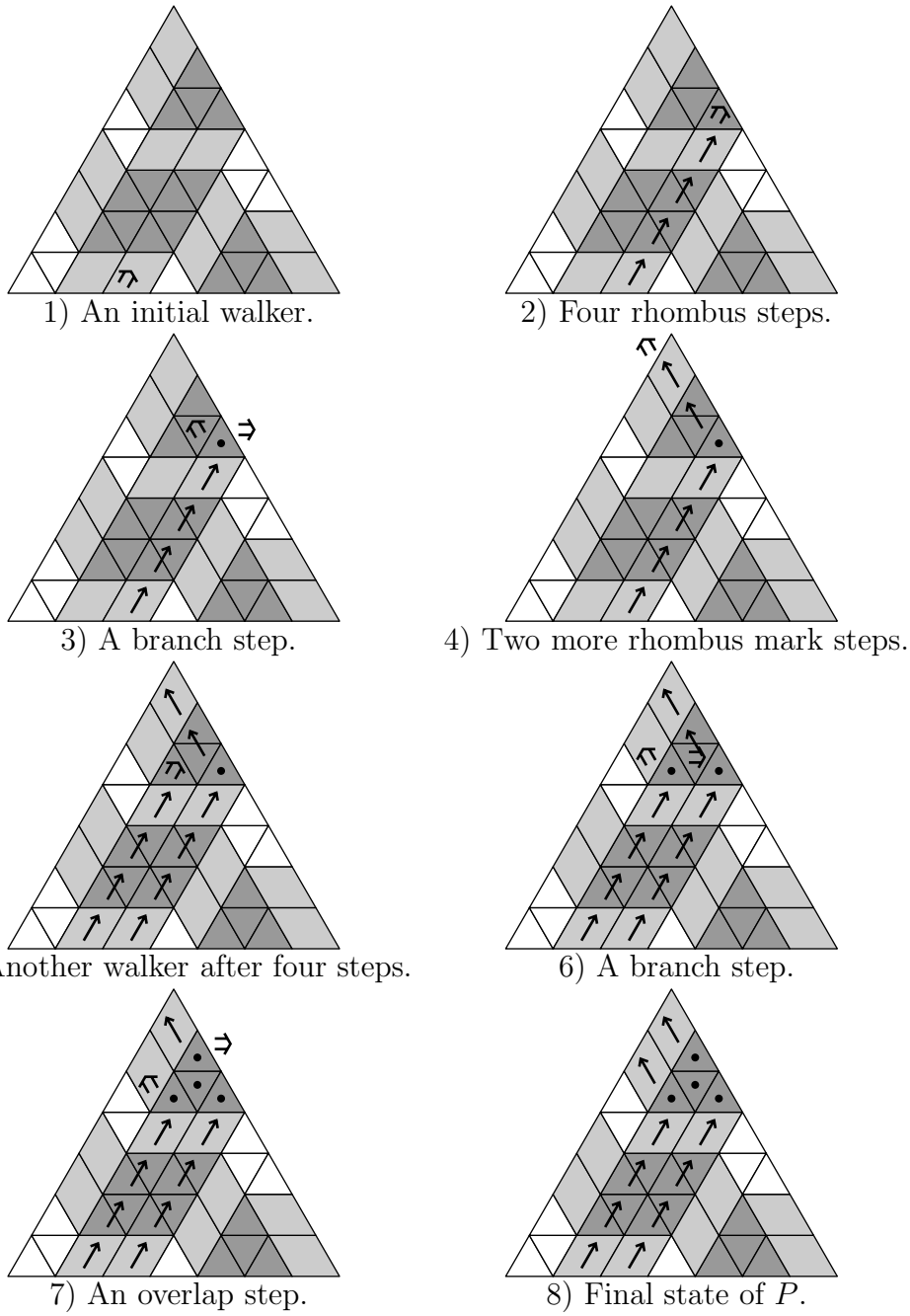
Walkers may step on to previously made marks. An **overlap step** is taken when the overlap is as in Figure 4.4. Specifically, an overlap step is taken when the walker is placed on a rhombus mark it is not parallel to and the walker is parallel to the center line of the marked rhombus. Figure 4.4 shows how the step is resolved.

Figure 4.4: The overlap step.

The last overlaps that can occur are called **non-rigid overlaps**, and as their name suggests, they indicate that the puzzle is non-rigid. We prove this claim in Proposition 4.1.12. If a non-rigid overlap occurs, all the walkers are stopped. Conceivably, there are more way for a walker to overlap a rhombus mark, but Lemma 4.1.2 shows the overlaps in Figures 4.4 and 4.5 are the only that can occur for walkers.

Figure 4.5: The non-rigid overlaps.

Figure 4.6 shows a full example of how a puzzle is marked.



1) An initial walker. 2) Four rhombus steps.

3) A branch step. 4) Two more rhombus mark steps.

5) Another walker after four steps. 6) A branch step.

7) An overlap step. 8) Final state of P.

Figure 4.6: An example of how a puzzle P is marked by a pair of walkers.

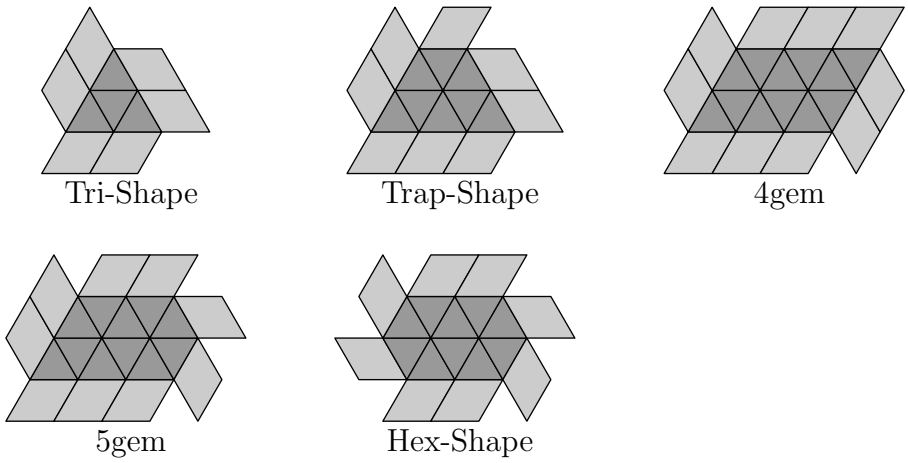


Figure 4.7: Examples of the five tri-regions with adjacent rhombi.

4.1.2 Tri-Regions

We will be examining how walkers proceed through puzzle piece regions, so it is convenient to develop some nomenclature for these regions. To avoid saying 'triangle piece region' throughout, a contiguous region of triangle pieces will be called a **tri-region**. A tri-region is dark or white depending on the triangles composing it.

It is not hard to see that the puzzle conditions ensure that tri-regions are convex, so there are only five types of tri-regions in puzzles. We name them as follows. A triangular shaped tri-region is a **tri-shape**. A trapezoid is a **trap-shape**. A six sided region will be a **hex-shape**. A rhombus shaped tri-region is a **4gem**, as in four sided gem, and a five sided region will be a **5gem**. The five dark tri-regions are shown in Figure 4.7 with their adjacent rhombi. The white versions are obtained by reflection.

A hex-shape in a puzzle indicates that the puzzle is non-rigid. Algorithm 4.2.2 searches the puzzle for hex-shapes, and it returns *EWF* if a hex-shape is found. The symbol EWF stands for 'End With Failure'. For now, we assume there are no hex-shapes in our puzzles.

We have one final rule for how walkers take steps in a puzzle. A **bad side** of a dark tri-region is any one of the sides labelled 'bad' around the tri-regions shown in Figure 4.8. The white tri-region bad sides are obtained by reflecting each case of Figure 4.8. All the other sides are called **good sides**. The rule for bad sides the following.

Bad Side Stop: If a walker takes a step over a bad side and finishes on a triangle piece, then all walkers are stopped from taking further steps. Figure 4.9 shows an example of a walker causing a bad side stop.

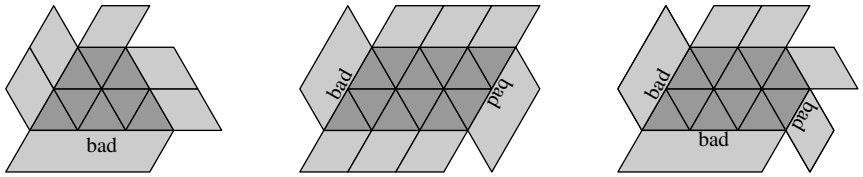


Figure 4.8: The bad sides of a trap-shape, 4gem and 5gem.

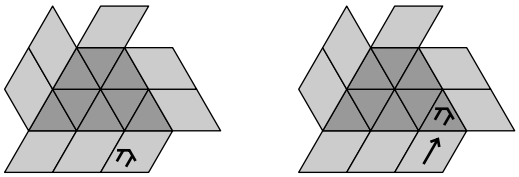


Figure 4.9: A walker crossing a bad side.

4.1.3 Skeleton Correspondence

Here, we prove some of the basic facts about walkers. These facts are needed to show that walkers fulfill their purpose, which is marking pieces in a puzzle which deflate to the small edges of a skeleton. The type of a walker is **dark** or **white** according to the type of triangle pieces they mark. This is well defined for a single walker as the rhombus and overlap steps taken by a single walker can only be made on a single type of triangle. The following lemma shows that all the descendants of a dark (respectively white) walker are dark (respectively white) walkers.

Lemma 4.1.1. *Let w be a walker in a puzzle P. All the descendants of a dark (respectively white) walker are dark (respectively white) walkers.*

Proof. Suppose w is an initial walker. We can assume without loss of generality that w is a dark walker, as the white walker case is similar. Then w was placed on a rhombus parallel to the white edges. First, we show that any all descendant walkers of w are also dark walker. The walker w can only take rhombus steps in one direction, so w is always parallel to the white edges

of a rhombus w started on. This is the case even if w has made taken an overlap step. Hence, w can not change to a white walker.

Suppose w takes a branch step in a tri-region T. Let v be a child of w. The walker v will take rhombus steps or overlap steps until it either takes a branch step, or is placed on rhombus piece outside of T. In the latter case, v is on a rhombus piece parallel to the white edge of that rhombus. Hence, v is also a dark walker. If v takes a branch step, we can repeat the above argument to show that any descendant of v which exits T will be a dark walker. The tri-region T is finite, so eventually, there can be no more descendants of w in T.

The argument above can be repeated when a descendant of w, v, enters another tri-region T'. This shows that all the descendants of v which exit T' will also be dark, as v must be dark. □

We get the following corollary recording a fact about the ways a walker can overlap a rhombus mark in P. We use Lemma 4.1.1 in Corollary 4.1.2 to show that the overlaps of Figures 4.4 and 4.5 are the only overlaps which can occur during the marking of a puzzle.

Corollary 4.1.2. *Let w be a walker in a puzzle P. The walker w can only overlap a rhombus mark as in Figures 4.4 or 4.5.*

Proof. Suppose w is an initial walker, and that w is a dark walker, the white walker case being similar. All the walkers that mark P are descendants of the dark walker w, so by Lemma 4.1.1, all the walkers that mark P are dark.

There are three types of overlaps not handled in the walker steps. The first type overlaps are pictured in the following diagram.

Both overlaps are impossible in P, as the walker in the diagrams is white.

The second type, pictured below, do not occur. The walkers are not parallel to either side of the rhombus piece, and so are neither dark or white walkers.

For the last type overlap, the walker v in both diagrams would have to have taken rhombus steps to the current position. There are a series of rhombus marks behind v with the same direction, so v must have entered this dark tri-region by a side parallel to the rhombus mark. However, the

walker v could not have been placed on any of rhombus pieces adjacent to that side and have the direction shown in the diagrams. This shows that the overlaps in the final diagram are not possible in P.

□

Now, if P has been marked by dark walkers, then the rhombus marks in P can survive a type 1 deflation. By 'survive', we mean that the rhombus mark is parallel to the rhombus edges that are identified in the shrinking process. The rhombus mark on a shrunk rhombus can simply be placed on the edge the rhombus its mark is shrunk to in $\mathcal{D}_1(P)$. An example of this is shown in Figure 4.10. Rhombus marks made by white walkers survive a type 0 deflation. In this way, the rhombus marks in a puzzle can be turned into marks on small edges in a measure.

Suppose P is a puzzle marked by walkers, and no bad side stop or a non-rigid overlaps occurred. We claim that the rhombus marks around each dark (respectively white) tri-region T in a puzzle P correspond to descendance marks around the lattice point of T branch point in P's type 1 (respectively 0) deflation. Proving this claim is the focus of this and the next subsections. We begin with a lemma characterizing when a walker will exit a tri-region without making a branch mark.

Lemma 4.1.3. *Let T be an unmarked tri-region in a puzzle P, and let w be a walker in T. Then no branch step was taken in T if and only if the side by which w entered T is parallel to the side by which w tried to leave T.*

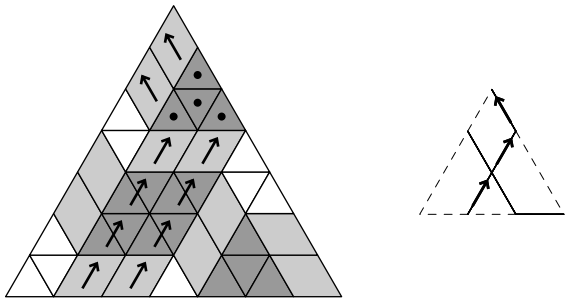


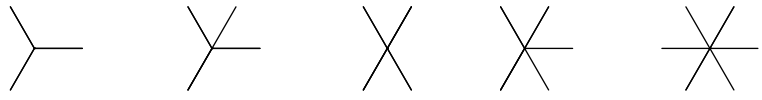
Figure 4.10: Rhombus marks made by dark walkers survive type 1 deflations.

We clarify the usage of 'tried to leave T.' Suppose the walker w took a rhombus step, regardless of whether it was possible, and landed outside of T. The side of T that w crossed over is the side by which w tried to leave T. Of course, w may be forced to take a branch step instead of a rhombus step. In that case, w would be removed without being placed outside of T.

Proof. If a walker, w, left T without leaving a branch mark, then w only placed rhombus marks in T. Those marks form a parallelogram whose short sides form part of the sides by which w entered and left T, so those sides are parallel.

On the other hand, let s be the side by which w entered T, and u be a side parallel to s. Suppose further that w tried to leave by the side u. Since s and u are parallel, the triangles forming s have the opposite orientation, pointing up or down, to the triangles forming u. All the triangle pieces in T on which w is placed have the same orientation as those forming s. Hence, the final triangle piece t_1 in T on which w lands on cannot be part of u. To leave by the side u, w must step over a triangle piece of the side u, say t_2. The triangle t_2 is adjacent to t_1. A rhombus mark can be placed over t_1 and t_2, so w can take a rhombus step and land outside of T, completing the proof. □

We note that we can define the deflation operation for incomplete puzzles, that is puzzle piece arrangement that obey the rules for adjacent pieces but do not form a triangle. If we take T to be a dark tri-region with its adjacent rhombi, the deflation $\mathcal{D}_1(T)$ is a lattice point with its surrounding small edges. Examples of the tri-region's deflations are shown in Figure 4.11. A **branch point** is the lattice points in $\mathcal{D}_1(T)$ or $\mathcal{D}_0(T)$ which is a deflation of a tri-region. The next result shows that when a walker enters an unmarked dark tri-region T, the lattice point $\mathcal{D}_1(T)$ is descendance marked. Our discussion is eased by the following notation: if T is a tri-region in a puzzle fragment, T_R is T plus the adjacent rhombi.



Tri-Shape Trap-Shape 4gem 5gem Hex-Shape

Figure 4.11: $\mathcal{D}_1(T)$ for the tri-regions shown in Figure 4.7.

Proposition 4.1.4. *Let T be an unmarked dark tri-region. Let w be a walker about to enter T by a good side. Then after all walkers have exited T_R, the branch point $\mathcal{D}_1(T_R)$ is descendance marked.*

Proof. We show for the four allowed tri-regions that the rhombus marks made around a tri-region deflate to descendance marks. In each case, T is a tri-region, and we examine the arrows around $\mathcal{D}_1(T_R)$.

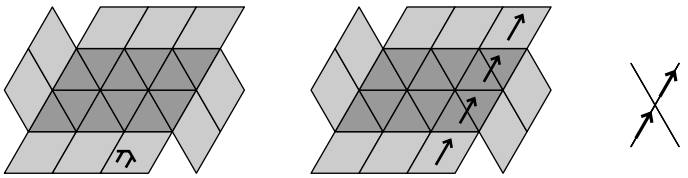


Figure 4.12: The marking of a 4gem, and the deflation.

4gem: The walker w enters T by a good side, and it is clear that w tries to leave T by the other good side. The good sides are parallel, so by Lemma 4.1.3, w will exit T without making a branch mark. The deflation $\mathcal{D}_1(T_R)$, shown in Figure 4.12, is descendance marked.

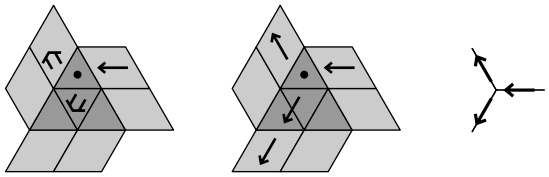


Figure 4.13: A walker marked tri-shape, and its deflation.

Tri-shape: No side of a tri-shape is parallel to any other, so w will make a branch mark on a triangle piece b in T. The branch mark is on the boundary of T, so it has the same orientation as T. The children of w are placed next to the branch mark so one of the children is placed on a rhombus outside of T. The other child, v, can be viewed as entering T by crossing the bottom line of b. Hence, v leaves T without making a branch mark, as the bottom edges of b and T are parallel. The lattice point $\mathcal{D}_1(T_R)$ is descendance marked as the children of w leave T at angles $\pm 60^\circ$ to the direction of w.

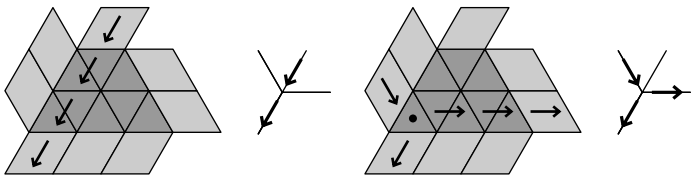


Figure 4.14: Both cases of a walker marked trap-shape, and their deflations.

Trap-shape: We first deal with w entering a trap-shape in T by a side parallel to the bad side. The deflation is descendance marked by the same argument as for the 4gem.

If w enters by either of the remaining good sides, we can argue as we did in the tri-shape case that w will make a branch mark, both children will leave T, and that $\mathcal{D}_1(T_R)$ is descendance marked. This is because additional triangle pieces can be added to the top side of T to make it a tri-shape, and none of the walkers take steps on those added triangles.

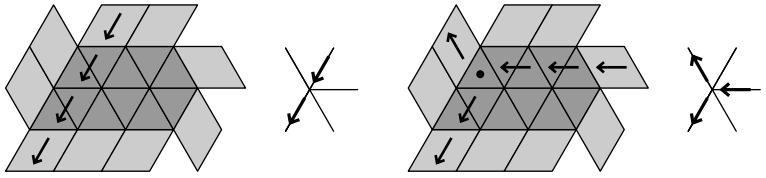


Figure 4.15: Both cases for a 5gem.

5gem: The argument for trap-shapes can be used for 5gems. First, add triangle pieces to the lower right corner of the 5gem, making T a trap-shape. Then if w enters by a good side of the 5gem, then w or its children do not step on those added pieces. Hence, $\mathcal{D}_1(T_R)$ is descendance marked by the argument for trap-shapes. □

4.1.4 Rhombus Mark Deflation

In this subsection, we aim to show that the deflations of marked tri-regions are descendance marked regardless of the number of walkers which passed through them. To help show this, we define a special deflation operation.

Rhombus Mark Deflation: Let U be a marked puzzle fragment. Say further that no bad side removals occurred during the marking of U. The rhombus mark deflation of U, denoted $\mathcal{D}_{\rightarrow}(U)$, is obtained by removing all the branch marked triangles, and identifying the sides parallel to a rhombus mark of any marked rhombus. The rhombus marks survive this deflation, and any arrows that are not on a piece edge are dropped.

Figure 4.16 shows an example of rhombus mark deflation. The furthest right horizontal rhombus mark is dropped in the deflation. Note that several rhombus marks in U may be deflated to a single arrow in $\mathcal{D}_{\rightarrow}(U)$.

It is a matter of checking simple cases to see that a tri-region marked by a

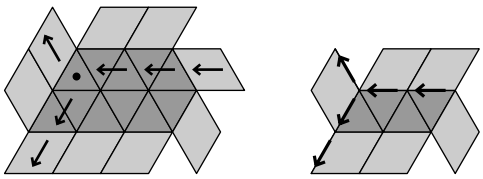


Figure 4.16: The rhombus mark deflation on a puzzle piece fragment.

single walker deflates to another tri-region. Further, any unmarked rhombus in T_R which is adjacent to a good side of T is adjacent to a good side in the deflation $\mathcal{D}_{\rightarrow}(T_R)$. The following lemma shows that if the arrows in $\mathcal{D}_{\rightarrow}(T)$ are on the boundary of the tri-region $\mathcal{D}_{\rightarrow}(T)$, then the behaviour of another walker entering the marked tri-region T can be understood by considering that walker's behaviour in the tri-region $\mathcal{D}_{\rightarrow}(T)$. The results of this section are all proved for dark walkers. The white walker case can obtained by reflecting the arguments and replacing dark and rightmost everywhere with white and leftmost.

Lemma 4.1.5. *Let T be a marked, dark tri-region, and let T_R be T with adjacent rhombi. Let the arrow marks in $\mathcal{D}_{\rightarrow}(T)$ be on the boundary. If a walker w enters T by a good side, then the new pieces marked in T_R by w and its children are the corresponding pieces marked by w and its children in the deflation $\mathcal{D}_{\rightarrow}(T_R)$.*

Proof. We consider any puzzle piece in T_R which is not removed to create the rhombus mark deflation $\mathcal{D}_{\rightarrow}(T_R)$ to be the corresponding puzzle piece in the deflation $\mathcal{D}_{\rightarrow}(T_R)$. Thus, if r is an unmarked rhombus in T_R, then r is a rhombus in $\mathcal{D}_{\rightarrow}(T_R)$.

Now, the walker w entered T by an unmarked rhombus; otherwise, a non-rigid overlap would have occurred. Let r be that unmarked the rhombus piece. As r is unmarked, r is a rhombus in the deflation $\mathcal{D}_{\rightarrow}(T_R)$, so r is the rhombus by which w enters the tri-region $\mathcal{D}_{\rightarrow}(T)$. The rhombus r is adjacent to a good side of $\mathcal{D}_{\rightarrow}(T)$, as r was adjacent to a good side of T.

The arrow marks left in $\mathcal{D}_{\rightarrow}(T)$ do not affect the types of steps a walker will take in the tri-region $\mathcal{D}_{\rightarrow}(T)$, so rhombus steps and branch steps are the only type of steps w takes in $\mathcal{D}_{\rightarrow}(T)$. We consider two cases; the walker w either does not take a branch step in $\mathcal{D}_{\rightarrow}(T)$ or it does.

In the first case, w only takes rhombus steps. If w does not take a rhombus step over an arrow mark in $\mathcal{D}_{\rightarrow}(T)$, then w will not overlap any rhombus marks in T. Thus, it is clear that the new pieces marked by w in

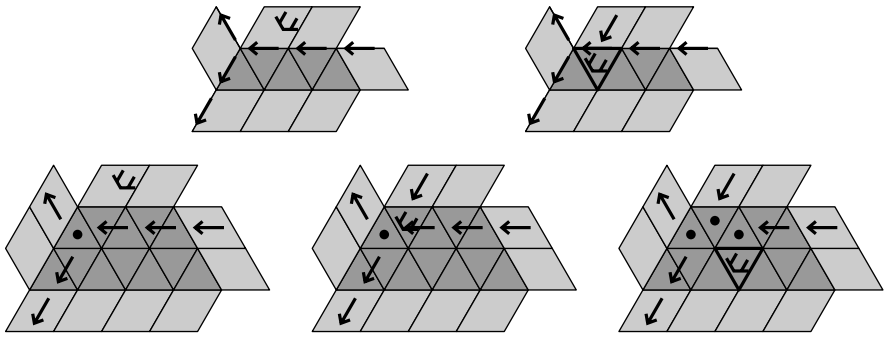


Figure 4.17: A rhombus step over an arrow in $\mathcal{D}_{\rightarrow}(T)$, and the result in T.

$\mathcal{D}_{\rightarrow}(T_R)$ are the same as w marks in T_R.

Suppose w does step over an arrow mark in $\mathcal{D}_{\rightarrow}(T)$. Figure 4.17 shows an example of this case. The top row shows the rhombus mark step in $\mathcal{D}_{\rightarrow}(T)$. The rhombus r in $\mathcal{D}_{\rightarrow}(T)$ is the rhombus the walker is on in the left diagram. Let t be the puzzle piece in $\mathcal{D}_{\rightarrow}(T)$ that w lands on after that step. The piece t in $\mathcal{D}_{\rightarrow}(T)$ is the bold piece in the top right diagram of Figure 4.17.

The arrow w steps over in $\mathcal{D}_{\rightarrow}(T_R)$ corresponds to one or more rhombus marked triangle pairs in T. Let a be the arrow in $\mathcal{D}_{\rightarrow}(T)$ which w steps over. Since a is on the boundary of $\mathcal{D}_{\rightarrow}(T)$, the corresponding rhombus marks are parallel to the side by which w enters or exits T. By lemma 4.1.3, we have that w will land on a rhombus mark as in one of the cases of Figure 4.4, so the next step w takes in T is an overlap step. The walker w will continue to take overlap steps until all the rhombus marks corresponding to the arrow a have been stepped over. The bottom row of Figure 4.17 shows how these overlap steps can occur in T_R.

The rhombus mark deflation ensures that the piece in T_R which w lands on after these overlap steps is the piece t on which w lands on in $\mathcal{D}_{\rightarrow}(T_R)$. In Figure 4.17, the piece t in T_R is the bold piece in the lower right diagram. Since t is the same piece in both $\mathcal{D}_{\rightarrow}(T_R)$ and T_R, the new pieces marked in $\mathcal{D}_{\rightarrow}(T_R)$ by w are the same as the new pieces marked in T_R, completing the first case.

In the second case, w takes a branch step in $\mathcal{D}_{\rightarrow}(T_R)$. The branch step must be the first type of branch step in Figure 4.3. Figure 4.18 shows an example of the steps taken by w and the children of w in $\mathcal{D}_{\rightarrow}(T_R)$ and T_R. If the branch step occurred in $\mathcal{D}_{\rightarrow}(T)$ such that there is an arrow, a, in front of w, then at least one of the children of w is placed on the other side of a in $\mathcal{D}_{\rightarrow}(T_R)$. Let v be the child placed on the other side of a, and let t be

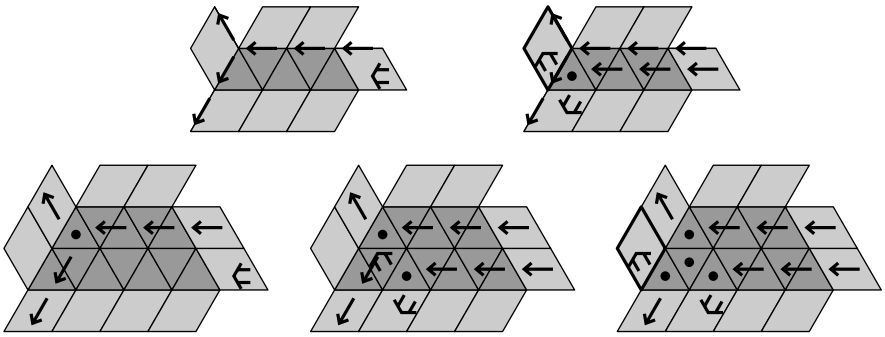


Figure 4.18: A branch mark step that encountered a boundary arrow in $\mathcal{D}_{\rightarrow}(T)$, and the result in T.

the piece v is on. In Figure 4.18, t is again the bold piece in the top right diagram.

In T_R, the branch mark step must be the second type of branch step in Figure 4.3, as the arrow a corresponds to rhombus marks in T. The child v is not immediately placed on the piece t in T_R; v will instead be placed on a rhombus mark, as in the bottom middle diagram of Figure 4.18. It is clear from the argument for the first case above that v will take overlap steps until it is placed on the piece t. Figure 4.18 shows v in T_R on the bold piece t in the bottom right case. Hence, it is clear that the new pieces marked in $\mathcal{D}_{\rightarrow}(T)$ and T_R are the same, completing the second case and the proof. □

Lemma 4.1.6. *Let T be a dark tri-region, and assume that any marks in T were made by walkers which entered T_R rightmost. Further, let w be a rightmost walker that enters T by a good side. Then w or its children leave T_R rightmost, and the arrow marks in $\mathcal{D}_{\rightarrow}(T)$ are on the boundary.*

Proof. We begin by showing the result in the case when w is the first rightmost walker to enter T. We claim that all of rhombus steps taken by w prior to a branch step are along a boundary of T. To see this, consider Figure 4.19. The figures shows five dotted lines, one for each of the other possible boundaries of T meeting at the triangle piece w first lands on. The fourth and fifth lines cannot be boundaries of T due to the convexity of tri-regions. The third cannot occur because w is a rightmost walker. In the first case, w makes a branch mark immediately. In the second, all of the rhombus steps taken by w in T are taken along that boundary of T, as claimed.

If w does not make a branch mark in T, then there are no triangle pieces to the right of the rhombus marks left in T, as all its rhombus steps in T were

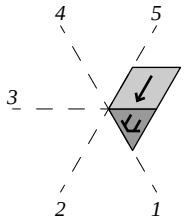


Figure 4.19: The five possible boundaries of T meeting at the triangle by which w.

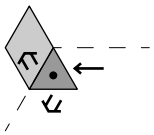


Figure 4.20: w children in relation to T's boundary at a branch mark triangle.

along a boundary. Hence, there are no rhombi to the right of the rhombus w lands on in T_R, so w is rightmost. Clearly, the rhombus marks it left deflate to the boundary of $\mathcal{D}_{\rightarrow}(T)$.

If w does leave a branch mark, then because w stepped along a boundary of T prior to the branch step, the children of w children are placed as in Figure 4.20. In particular, one of the children is placed rightmost on a rhombus adjacent to the branch marked triangle. The other child takes rhombus steps along a boundary of T until it exits T, by Proposition 4.1.4, so it leaves rightmost, similarly to w leaving rightmost above. It is also clear that all the rhombus marks left by w and its children deflate to arrows on the boundary of $\mathcal{D}_{\rightarrow}(T)$.

We now take T to be a marked tri-region. We proceed by induction on k, the number of rightmost walkers to enter T. If $k = 1$, then the paragraphs above serve to show that when a single rightmost walker w enters T, all walkers leaving T_R are rightmost and the arrows in $\mathcal{D}_{\rightarrow}(T)$ are on the boundary.

Suppose that k rightmost walkers have marked T, that the arrows in $\mathcal{D}_{\rightarrow}(T)$ are on the boundary, and all walkers exited T_R rightmost. w is the $(k+1)$st rightmost walker entering T. To distinguish the state of T before and after w has marked it, denote by T^* the tri-region T after w and its children have marked it. The notation T_R^* and $(\mathcal{D}_{\rightarrow}(T_R))^*$ is used similarly.

The tri-region $\mathcal{D}_{\rightarrow}(T_R)$ is not rhombus marked, as all the rhombus marks in T_R have been deflated, so we can apply the $k = 1$ case to show that

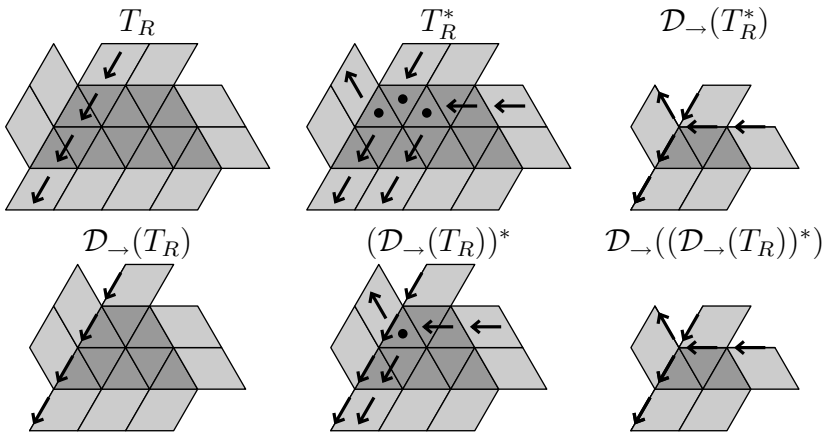


Figure 4.21: The various stages of marking T_R and their rhombus mark deflations.

all walkers leaving $\mathcal{D}_\rightarrow(T_R)$ are rightmost and the arrows in $\mathcal{D}_\rightarrow((\mathcal{D}_\rightarrow(T))^*)$ are on the boundary. We claim that the arrows in $\mathcal{D}_\rightarrow(T^*)$ are also on the boundary. Since the original arrows in $\mathcal{D}_\rightarrow(T)$ are on the boundary, we can apply Lemma 4.1.5 to get that the newly marked pieces in T_R^* are the same pieces marked in $(\mathcal{D}_\rightarrow(T_R))^*$ by w and its children. It is then clear that $\mathcal{D}_\rightarrow(T_R^*) = \mathcal{D}_\rightarrow((\mathcal{D}_\rightarrow(T_R))^*)$, proving the claim. Figure 4.21 shows an example for each stage of deflation.

We show that a walker v which exited $\mathcal{D}_\rightarrow(T)$ during the marking of $(\mathcal{D}_\rightarrow(T_R))^*$ is rightmost in T_R^*. Say v exited $\mathcal{D}_\rightarrow(T)$ by crossing a side s. The $k = 1$ case gives that v is rightmost in $\mathcal{D}_\rightarrow(T_R)$. The arrow marks in $\mathcal{D}_\rightarrow(T_R)$ which point away from the side s must be to the right of v. Otherwise, a previous walker exited T by a rhombus piece to the left of an unmarked rhombus, contradicting the induction hypothesis that all previous exiting walkers were rightmost. Hence, v must be rightmost in T_R, as any rhombi to the right of v are rhombus marked. □

Corollary 4.1.7. *All walkers marking a puzzle P are rightmost.*

Proof. As the initial walkers are rightmost, this corollary is proved by applying Lemma 4.1.6 to give a simple induction on the number of times a walker enters a tri-region in P. □

We now show that for any dark tri-region T, marked or not, the marks around T_R deflate so that $\mathcal{D}_1(T)$ is descendance marked. Again, we note that

we are have assuming that the walkers are dark and rightmost, as the white and leftmost case is easily obtained from the dark case.

Proposition 4.1.8. *Let T be a dark tri-region, and assume that all the marks in T were made by rightmost walkers and their children. Then $\mathcal{D}_1(T_R)$ is descendance marked.*

Proposition 4.1.8 proves a stronger results than does Proposition 4.1.4. In particular, Proposition 4.1.8 proves that $\mathcal{D}_1(T_R)$ is descendance marked when the tri-region T has been previously marked, whereas Proposition 4.1.4 assumes T is unmarked.

Proof. We proceed by induction on the number of walkers to enter T. Our base case is provided by Proposition 4.1.4. Now say that k walkers have previously entered T, and that $\mathcal{D}_1(T_R)$ is descendance marked. Let w be the $(k+1)$st walker to enter T, and again use the notation T^* to denote the state of T after w and its children have marked it.

By assumption, the previous walkers that entered T were rightmost, so by Lemma 4.1.6, the arrows in $\mathcal{D}_{\rightarrow}(T)$ are on the boundary. Applying Lemma 4.1.5, we have that the new marks in T_R^* are the marks made in $(\mathcal{D}_{\rightarrow}(T))^*$. For the moment, we erase the arrow marks in $(\mathcal{D}_{\rightarrow}(T))^*$. We have that the deflation $\mathcal{D}_1((\mathcal{D}_{\rightarrow}(T))^*)$ is descendance marked by Proposition 4.1.4, that is the arrow pointing into the branch point has all of its descendants marked by out arrows.

To complete the induction, we need to show that $\mathcal{D}_1(T_R^*)$ is still descendance marked. The deflation $\mathcal{D}_1(T_R)$ is descendance marked by induction, so we need only show that the new arrows in $\mathcal{D}_1(T_R^*)$ are descendance marks. But this is true since the small edges around the branch point in

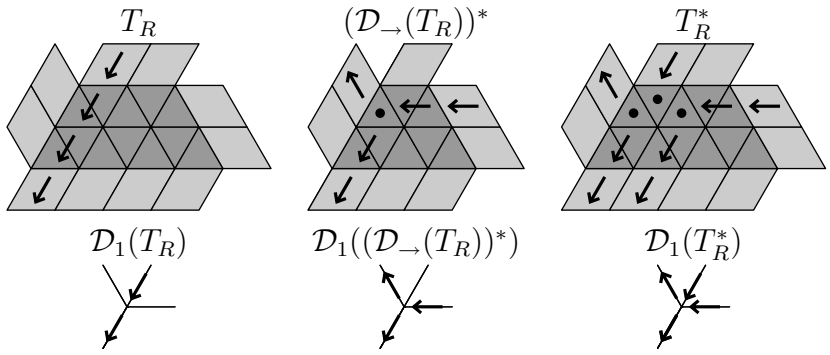


Figure 4.22: The type 1 deflation of each marking of T_R.

$\mathcal{D}_1((\mathcal{D}_{\rightarrow}(T))^*)$ are present around $\mathcal{D}_1(T_R^*)$, and we showed in the previous paragraph that those new arrows are descendance marks. □

We get the following corollary.

Corollary 4.1.9. *Let P be a marked puzzle. Then all the lattice points of in a deflation of P deflation are descendance marked or have no incident arrows.*

Proof. Proposition 4.1.8 shows that if a branch point has arrows around it, then it is descendance marked. The remainder of the lattice points are on line segments that inflate to rhombus regions. If a walker proceeded through a rhombus region R, then there are arrows with the same direction on every small edge of R's deflation. The arrows along e_R point from ancestor to descendant, so each lattice point along e_R is descendance marked. □

From Corollary 4.1.9, it is clear that following a series of rhombus marks describes a descendance path in a deflation of P.

4.1.5 Non-Rigid Overlaps

In this section, we deal with the claim made in Subsection 4.1.1 that non-rigid overlaps indicate that P is non-rigid. Figure 4.23 shows a simple example of how a non-rigid overlap can occur in a puzzle fragment. The left diagram of Figure 4.23 shows an initial, white walker placed on a rhombus piece, and the right diagram shows a non-rigid overlap that occurs when a descendant of the initial walker steps onto a rhombus mark. We prove a lemma and some corollaries which will be used in the proof of Proposition 4.1.12 below.

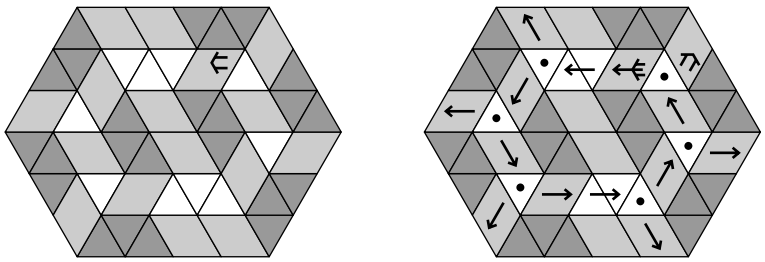


Figure 4.23: A non-rigid overlap in a puzzle fragment.

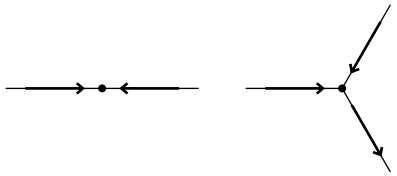


Figure 4.24: These rhombus marks indicate P is not rigid.

Lemma 4.1.10. *Let P be a marked puzzle. If a pair of rhombus marks is made in P such that in P's deflation the arrows are as in figure 4.24, then P is not rigid.*

We clarify how two rhombus marks might end up as in Figure 4.24 with the two examples in Figure 4.25. In each, it is clear that the type 1 deflation will have a pair of arrows arranged as one of the cases in Figure 4.24. The walkers that made the marks in both cases are rightmost. We now prove the lemma.

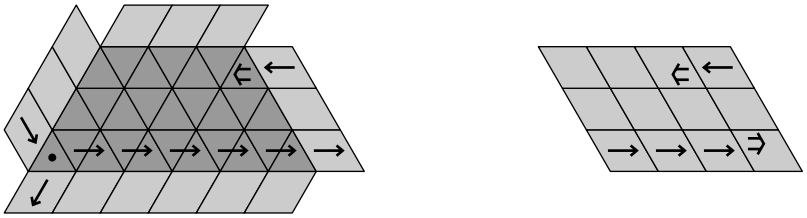


Figure 4.25: Two examples of rhombus marks which deflate to arrows as in Figure 4.24.

Proof. We prove the result for dark tri-regions, the white case being similar. Now, there is a series of rhombus and branch steps that can be followed backward from the two rhombus marks in each case of Figure 4.24. Since every walker is descended from the initial walker, these two series of steps must have some mark in common, either a branch mark or an initial rhombus mark. This forms a cycle of rhombus and branch marks.

We show that this cycle is an evil loop in $\mathcal{D}_1(P)$. This requires that each turn be evil. Corollary 4.1.9 says that the rhombus marks deflate to descendance paths. From Section 3.1.2, we have that each turn in a descendance path is evil. Thus, if the turn where the rhombus marks meet is evil, then there is an evil loop in P, and in each case of Figure 4.24, the turn is clearly evil. □

The following corollary gives another way to show that P is non-rigid.

Corollary 4.1.11. *Let T be a tri-region in a puzzle P. If a walker w tries to enter T by a side through which another walker previously exited T, then P is non-rigid.*

Proof. If this occurred, the deflation of T would deflate to one of the two cases of Figure 4.24. We can apply Lemma 4.1.10 to show P is non-rigid. □

We now show that non-rigid overlaps are correctly named, in that they indicate that P is not rigid.

Proposition 4.1.12. *If a non-rigid overlap occurs in a puzzle P, then P is not rigid.*

Proof. Let w be a walker overlapping another mark. All walkers were halted when the non-rigid overlap occurred, so this overlap is the first to have occurred in P. Let v be the walker that originally took the step that created the mark w overlaps. Suppose v was the initial walker. Then there is a cycle of rhombus marks starting and ending at the rhombus the walker v was initially placed on. The second paragraph of the proof of Lemma 4.1.10 can be used to establish that there is an evil loop in $\mathcal{D}_1(P)$, so P is not rigid.

We now assume that v was not the initial walker. It is the case that the walkers w and v entered the piece where the overlap occurred from different directions, as otherwise the overlap would have occurred earlier. We deal with three cases for how the overlap occurred.

If the overlap occurred in a tri-region T, then w and v entered T by different sides, so the direction of w is 120° or 180° from the direction of v. Hence, the deflation of T looks like one of the cases in Figure 4.25, so P is non-rigid by Lemma 4.1.10. If the overlap occurred on a rhombus adjacent to a tri-region, a similar argument again shows that P is non-rigid. Finally, if the overlap occurred on any other rhombus, then the deflation of P has the first case in Figure 4.24 at a lattice point adjacent to the overlap, and again, we apply Lemma 4.1.10 to show P is not rigid. □

4.2 Puzzle Skeletons

In this section, we define two algorithms. The first algorithm is called a rhombus walk. Rhombus walks employ the walkers of Section 4.1 to mark a puzzle P. The second algorithm, the Puzzle Skeleton Algorithm, searches for a rhombus walk in which no bad side stops or non-rigid overlaps occur. In

Subsection 4.2.4, we introduce two puzzles created from the marks made in a puzzle by the Puzzle Skeleton Algorithm. The statements of the algorithms are followed by some example computations in Subsection 4.2.2, and the proofs of termination and correctness can be found in Subsection 4.2.3. We turn to stating the algorithms of this section.

4.2.1 Puzzle Skeletons

The first algorithm we state in this section is called the rhombus walk algorithm. As mentioned in Subsection 4.1.1, the conditions on initial walker placement have not been fully stated. We begin this subsection by stating the initial placement rules.

Initial walkers are placed in a puzzle P in pairs. Let w and v be the initial walkers placed in P. The position of v is determined from the position of w. To place w, choose a tri-region T and a side s of T. The walker w is placed on a rhombus adjacent to the side s. If T is dark, then w is dark and is placed rightmost. If T is white, then w is white and is placed leftmost. The tri-region T is required to be a trap-shape, 4gem or 5gem, and the side s is determined by the type of tri-region chosen. If T is a 4gem, s is either one of the good sides. For a trap-shape and 5gem, the side s is the good side of T such that if a walker entered T by s, then that walker will not take a branch step in T.

The walker v is placed as follows. Place v on the rhombus w is on but with the opposite direction to w. If w is a dark walker, then shift v right till it is on the rightmost rhombus from the perspective of v. If w is white, then replace right by left in the preceding statement. Finally, advance v one rhombus step without making a rhombus mark. Figure 4.26 shows an example of this. We call the tri-region T a **starting tri-region**, and w and v are T's **initial walkers**.

Once initial walkers are placed, the oldest walker in the puzzle is selected to take a step. **Walker age** is uniquely determined as the time since the

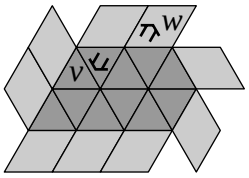


Figure 4.26: The initial walkers of T.

walker was placed in the puzzle and two other conditions for the walkers which are placed simultaneously. First, w is the oldest of a pair of initial walkers. Second, the child created clockwise of the other at a branch step is older. The oldest walker will take steps until it is removed or some other terminating condition occurs.

The terminating conditions for a rhombus walk are a bad side stop, a non-rigid overlap or all walkers have been removed from the puzzle. In the case of a bad side stop, the rhombus walk terminates and returns the tri-region B where the bad side stop occurred. If a non-rigid overlap occurs, the rhombus walk terminates and returns the symbol **EWF**, which is an 'End With Failure'. If there are no walkers remaining in P, then the marked puzzle is returned. We now state the algorithm.

Algorithm 4.2.1 (Rhombus Walk)**.** *The input to the algorithm is a puzzle P without hex-shapes, and a pair of walkers in P.*

1. *Place T's initial walkers.*
2. *While walkers are in P,*
   - *i) Let the oldest walker w take a step.*
   - *ii) If a non-rigid overlap occurs, then terminate and return EWF.*
   - *iii) If a bad side stop occurs, then terminate and return the tri-region B.*
3. *Return the marked puzzle P.*

We describe the Puzzle Skeleton Algorithm next. It functions by starting a new rhombus walk using the tri-region B at which a bad side stop occurred in the preceding rhombus walk. It is possible to have a cycle of starting tri-regions. As we see in Proposition 4.2.4, this indicates that the puzzle is not rigid. To ensure the algorithm terminates, a list of starting tri-regions is kept, and the algorithm returns EWF if a repeat occurs. The Puzzle Skeleton Algorithm also returns EWF if P contains a hex-shape or a rhombus walk returns EWF.

The Puzzle Skeleton Algorithm uses rhombus walks in a second way, for which we extend the rules for placing pairs of **initial walkers** in a marked puzzle P. The new walkers are placed as follows.

Let W be a rhombus walk which returns a puzzle skeleton. Let T be the starting tri-region of W, and let s be the side which was used to place the initial walkers w and v. If all the pieces along the side s are marked, then no more walkers may be placed in P. Let s have an unmarked piece adjacent to it, and let w' and v' be the additional pair of walkers we wish to place in P. The walkers w' and v' are placed obeying the following three conditions if T is a dark tri-region.

1. The walker w' is placed on an unmarked rhombus and v' is placed on an unmarked triangle both adjacent to the side s.

2. The walker w' is dark and points away from the side s, and v' has the opposite direction.

3. The walker w' is placed on the rightmost unmarked rhombus adjacent to s, and v' is similarly placed on the rightmost unmarked triangle adjacent to s.

If T is white, w' is white, and both walkers are placed leftmost. The walkers w' and v' are called T's initial walkers in the marked puzzle. The puzzle algorithm starts rhombus walks from T until initial walkers can no longer be placed in along the side s.

We call the set of marks returned by the Puzzle Skeleton Algorithm a **puzzle skeleton**, and a puzzle skeleton is dark or white depending on the type of the initial walkers. We now state the algorithm.

Algorithm 4.2.2 (Puzzle Skeleton Algorithm)**.** *The input to the algorithm is a puzzle P with a trap-shape. It maintains a list L of starting tri-regions.*

1. *If T is a hex-shape, terminate, and return EWF.*

2. *Find either a 5gem, 4gem or trap-shape T in P, in that order of precedence.*

3. *Run a rhombus walk W starting at T, and add T to L.*

4. *If W returns a tri-region B, loop until W returns a marked puzzle or EWF.*

   i) Erase the marks in P.

   ii) If B is already on the list L, terminate and return EWF.

iii) Run a rhombus walk W on P with starting tri-region B.

5. *If W returns EWF, terminate and return EWF.*
6. *While initial walkers can be placed for P,*

   i) Run a rhombus walk W starting at P.

7. *Return the marked puzzle P.*

Note that if the Puzzle Skeleton Algorithm does not return EWF, then it returns a marked puzzle P. The marks in P are the result of a rhombus walk which terminated without a bad side stop or non-rigid overlap, so the marks form a puzzle skeleton.

4.2.2 Example Computations

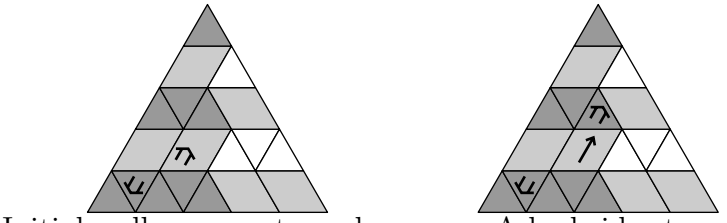
In this subsection, we demonstrate the Puzzle Skeleton Algorithm on some examples. A simple example can be seen at the end of Subsection 4.1.1, though the initial walkers are not placed on the 4gem. The first example here is shown in Figure 4.27. It shows the initial walkers, a bad side stop, a new rhombus walk, and the final marked puzzle.

The next example, Figure 4.28, shows the Puzzle Skeleton Algorithm executing on a puzzle P where a starting tri-region is placed on the list L a second time. The initial walkers are placed on the 5gem, and one of the walkers quickly comes to a bad side stop. The next two rhombus walks both end with bad side stops, and in particular, the second has a bad side stop on the original 5gem. Hence, the symbols EWF is returned, indicating P is not rigid. In fact, the cycle itself can be used to find a gentle loop in P, and the gentle in this example is indicated by the bold polygon in the final step.

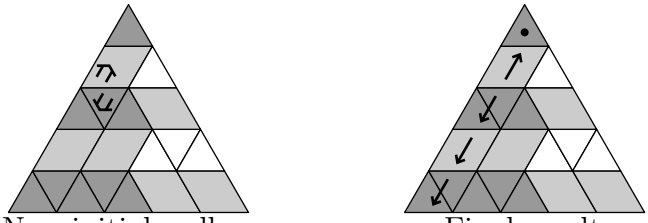
4.2.3 Termination and Correctness

In this subsection, we prove three facts about the Puzzle Skeleton Algorithm. The first is that the algorithm terminates. The second is that the algorithm only ends with failure if the puzzle is non-rigid. Finally, we prove that a puzzle skeleton marked in a rigid puzzle deflates to a skeleton in the deflation of the puzzle.

In Subsection 4.1.1, the marking procedure was started with a single initial walker. We now start with two initial walkers. To use the results

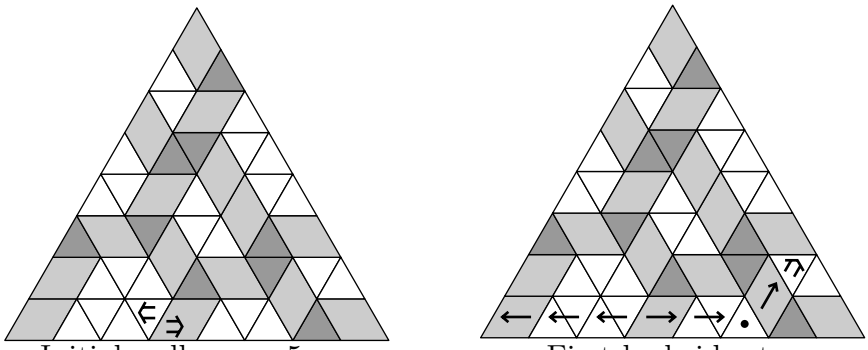


Initial walkers on a trap-shape. A bad side stop.

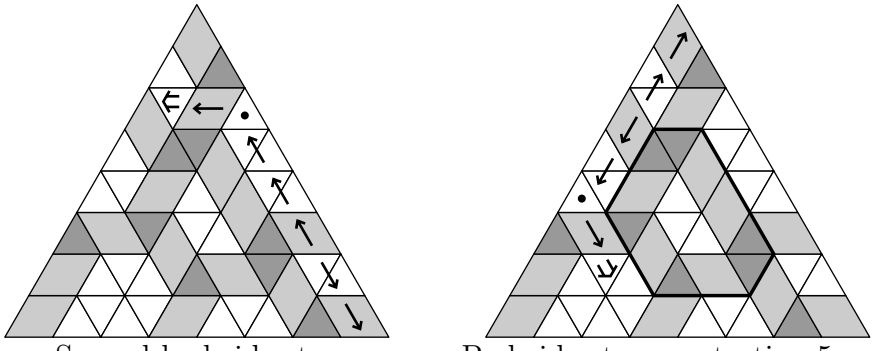


New initial walkers. Final result.

Figure 4.27: A simple example with a bad side stop.



Initial walkers on 5gem. First bad side stop.



Second bad side stop. Bad side stop on starting 5gem.

Figure 4.28: A cycle of starting tri-regions, and the gentle loop in P found using the cycle.

from Section 4.1, we must ensure that the proofs of Section 4.1 still hold. The proofs of Subsections 4.1.3 and 4.1.4 were all made in terms of puzzle fragments, so they did not depend on the number of initial walkers. For Subsection 4.1.5, the placement of the initial walkers w and v ensures that the first rhombus marks they each make on a rhombus piece both point away from the same branch point in the deflation of P. Hence, the arguments for Lemma 4.1.10 and Proposition 4.1.12 are easily extended to the placement of two initial walkers. We turn to proving that the Puzzle Skeleton Algorithm terminates.

Lemma 4.2.3. *The Puzzle Skeleton Algorithm terminates for any puzzle P.*

Proof. The number of walkers created in a puzzle is finite, as walkers are only created during a branch step. There are finitely many triangle pieces in P, and each triangle piece can only be marked once by a branch mark, or a non-rigid overlap occurs. Hence, every rhombus walk started by the Puzzle Skeleton Algorithm terminates.

There are finite number of tri-regions in P. Each tri-region in P can only be used to start a single rhombus walk. Otherwise, the Puzzle Skeleton Algorithm terminates and returned EWF. □

The next proposition proves that if the Puzzle Skeleton Algorithm returns EWF, then P is indeed non-rigid.

Proposition 4.2.4. *Let P be a puzzle without hex-shapes. If the Puzzle Skeleton Algorithm returns EWF, then P is not rigid.*

Proof. The Puzzle Skeleton Algorithm returns EWF in three situations. We deal with each separately.

First, suppose EWF is returned because P contains a hex-shape. There is a gentle loop running along the boundary of the hex-shape, so P is not rigid.

Second, the rhombus walk returned EWF. In this case, a non-rigid overlap occur during the rhombus walk, and Proposition 4.1.12 shows that P is not rigid.

The third condition is that a tri-region T is placed on the list L of initial tri-regions more than once. Let k be the length of the list L. Let $T_0 = T_k = T$, and For $1 \leq i \leq k-1$, let T_i be the ith tri-region on L. We can assume without loss of generality that the T_i are all dark tri-regions. Let $b_i = \mathcal{D}_1(T_i)$

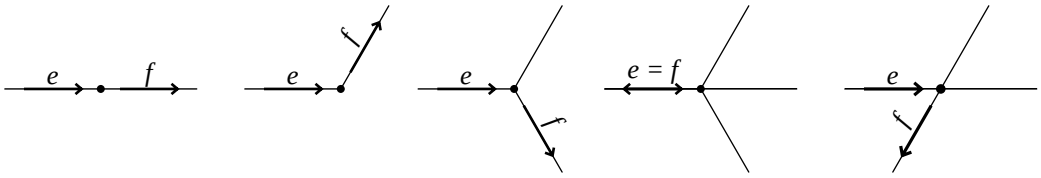


Figure 4.29: The five evil turns from e to f.

be the branch mark T_i deflates to, and let w_i and v_i be the initial walkers started at T_i. Our indexing will be carried out mod k for simplicity.

In every case, a walker descended from w_i or v_i tried to enter a bad side of T_{i+1}. There is a directed path along rhombus marks from a side of T_i to a bad side of T_{i+1}. Let Q_i be the rhombus mark path from T_i to T_{i+1}, and let R_i be the rhombus mark path where each rhombus mark of Q_i is reversed. Place all the rhombus marks from the paths R_i in P.

Corollary 4.1.9 shows that each R_i is a descendance path from b_{i+1} to b_i. Every turn in a descendance path is evil in either direction. Hence, if we can show that there is an evil turn at every b_i that takes you from R_i to R_{i-1}, then the cycle formed by the paths is an evil loop in $\mathcal{D}_1(P)$. In Figure 4.29, we reproduce the figure showing the five evil turns for measures in $\mathcal{M}_r$.

If T_i is a trap-shape, then there are two cases for b_i depending on the side by which R_i enters T_i. Note that R_{i-1} points away from the bad side of the trap-shape T_i. If the path Q_i was started by the walker w_i, then Q_i left T_i from a good side. When rhombus marks of the paths are reversed, there is an evil turn at b_i from the first arrow of R_{i-1} to last arrow of R_i, as the marks are parallel. If Q_i left T_i by the bad side, then the turn from R_i to R_{i-1} at b_i is evil, as it is the fourth evil turn shown in Figure 4.29.

Suppose T_i is a 4gem. The first arrow of Q_i points away from a good side of T_i, so the last arrow of R_i points towards T_i. The last arrow of R_{i-1} points away from a bad side of T_i. In the deflation, there is an arrow pointing toward b_i that is 60° counter clockwise or 120° clockwise from an arrow pointing away from b_i. We see that the turn from R_i to R_{i-1} at b_i is evil, as it is either the second or fifth turn in Figure 4.29.

Let T_i be a 5gem. Note that 4gems and 5gems have the same deflation save one additional edge for the 5gem. The additional edge is the deflation of a good side that is not used to start a rhombus walk, so there is no arrow on it. Hence, the argument for 4gems shows that the turn at b_i is evil, unless the arrows in and out of b_i use the same edge. In that case, the turn at b_i is the fourth case of Figure 4.29.

These three cases show that the arrows at each b_i form an evil turn from the path R_i to the path R_{i+1} for each i, giving an evil loop in $\mathcal{D}_1(P)$. Hence, P is not rigid. □

We now prove that the marks made by the Puzzle Skeleton Algorithm do indeed deflate to a skeleton in the deflation of the puzzle.

Proposition 4.2.5. *Let P be a rigid puzzle in which a puzzle skeleton is marked. Then the arrows in the deflation of P mark a skeleton, and all the pieces in P which deflate to that skeleton are marked.*

Proof. Let the puzzle skeleton in P be dark, the white case being similar, and let W be the rhombus walk that marked the puzzle skeleton. Let T be the starting tri-region of W, and let R be the rhombus region that one of T's initial walkers was placed in. Denote the deflation of R in the measure $\mathcal{D}_1(P)$ by e_R.

The initial walkers were rightmost, so all the walkers marking P were rightmost, by Lemma 4.1.6. This allows us to use Proposition 4.1.8 and Corollary 4.1.9 to show that there is a descendance path following arrows from e_R to every arrow marked edge in $\mathcal{D}_1(P)$.

We also have that every descendant of e_R is marked. To see this, note that no bad side stops occurred during W, so every branch point with an arrow pointing in has arrows pointing out, unless the point is on the boundary. Proposition 4.1.8 shows that each branch point is descendance marked, so the arrows pointing out of the branch point are all the descendants of an arrow pointing in. Hence, by the argument made in Subsection 3.1.3, the arrows in $\mathcal{D}_1(P)$ mark the small edges of a skeleton S_{e_R}, as the marks in $\mathcal{D}_1(P)$ form a descendance set without any lattice points like those in Figure 3.14.

Recall the measure μ_{e_R} from Subsection 3.1.3 that is created from the skeleton S_{e_R}. To show we have marked all the pieces which deflate to μ_{e_R}, we must show that the number of rhombus marks which deflate to a particular arrow on a small edge a in $m = \mathcal{D}_1(P)$ is equal to the density of the measure μ_{e_R} on the small edge a. The number of rhombus walks started by the second while loop of the Puzzle Skeleton Algorithm is equal to the number of rhombi that deflate to e_R in m, which $m(e_R) = \mu_{e_R}(e_R)$. The number of rhombus marks that deflate to the arrow on a is equal to $m(e_R)$ times the number of descendance paths from e_R to a. The density $\mu_{e_R}(a)$ of the small edge a is constructed in the same way, so the number of arrow marks on a is equal to

the density of the measure μ_{e_R} on a. Hence, every puzzle piece which deflates to μ_{e_R} is marked in P. □

4.2.4 The Two Puzzles Created From a Puzzle Skeleton

We finish this section with the descriptions of two puzzles constructed from the marks of a puzzle skeleton which will be used later. Our main goal is simply to confirm that they are both puzzles, and that when P is rigid, they are rigid as well. We also show the construction of a third puzzle that is related to the Horn inequalities of Subsection 3.2.2. However, the third puzzle is not used later in this thesis.

The first puzzle, called the **rhombus mark deflation** of P is constructed using the rhombus mark deflation, $\mathcal{D}_{\rightarrow}(P)$, and removing the arrow marks. We denote this puzzle $\mathcal{D}^{\circ}_{\rightarrow}(P)$ the puzzle created from $\mathcal{D}_{\rightarrow}(P)$ without the arrows. Figure 4.30 shows an example of a puzzle skeleton in a puzzle, and the construction of $\mathcal{D}^{\circ}_{\rightarrow}(P)$. We show that $\mathcal{D}^{\circ}_{\rightarrow}(P)$ is a puzzle.

Lemma 4.2.6. *Let P be rigid and have a puzzle skeleton marked in it. Then $\mathcal{D}^{\circ}_{\rightarrow}(P)$ is a rigid puzzle.*

Proof. Branch marked pieces are only adjacent to other marked pieces or the boundary of P, so their deletion will not leave holes in $\mathcal{D}^{\circ}_{\rightarrow}(P)$. The marks in P are a puzzle skeleton, so every rhombus mark points towards another rhombus mark, a branch mark or the boundary of P, as no bad side removal occurred. This ensures that any edge shrunk in the deflation is adjacent to another shrinking edge, a deleted piece or the puzzle boundary.

The rhombus mark deflation also identifies the two edges of a marked rhombus which are parallel to the rhombus mark itself. Those two sides are

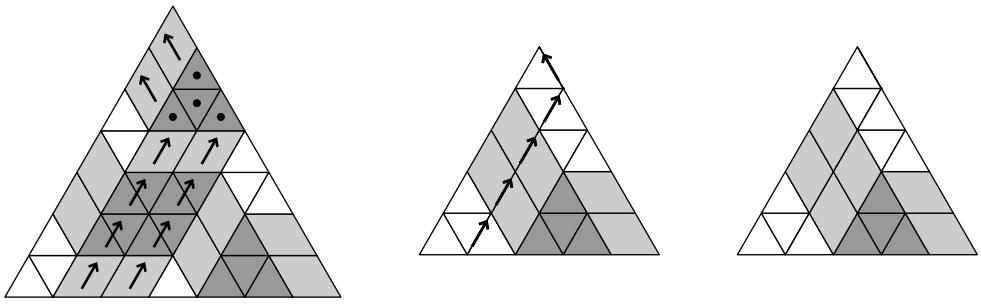


Figure 4.30: A puzzle skeleton in P, and the puzzle $\mathcal{D}^{\circ}_{\rightarrow}(P)$.

the same type, dark or white, regardless of whether the marked rhombus is a rhombus piece or a pair of triangles. Hence, the pieces made adjacent by the deflation meet along edges of the same type.

The boundary of $\mathcal{D}^{\circ}_{\rightarrow}(P)$ is triangular. To see this, note that the puzzle skeleton deflates to a skeleton in $\mathcal{D}_1(P)$, and skeletons obey the balance conditions at every lattice point in $\triangle_{n-d}$. Hence, the densities on the exit edges along each side of the boundary sum to the same integer, so the same number of edges along on the boundary of P are shrunk to form $\mathcal{D}^{\circ}_{\rightarrow}(P)$.

Finally, suppose that $\mathcal{D}^{\circ}_{\rightarrow}(P)$ is not rigid. Then there is a gentle loop in $\mathcal{D}^{\circ}_{\rightarrow}(P)$. Consider $\mathcal{D}_{\rightarrow}(P)$ where we have kept the arrow marks. If this gentle loop crosses arrow marks in $\mathcal{D}_{\rightarrow}(P)$, a simple case analysis shows that the gentle loop can be extended across the deflated pieces to form a gentle loop in P. If the gentle loop does not cross the arrows, the loop is a gentle loop in P. Both cases contradict the rigidity of P. □

The second puzzle we constructed is called the **stretched puzzle**. The stretched puzzle is constructed from the marks of a puzzle skeleton. Suppose a dark (respectively white) puzzle skeleton is marked in P. The stretched puzzle is denoted by $\mathcal{S}(P)$, and it is constructed as follows.

1. Set all unmarked puzzle positions to be white (respectively dark) triangles,

2. change rhombus marked pairs of triangles into rhombus pieces, and

3. clear the branch and rhombus marks.

Note that the branch mark triangles remain dark (respectively white) triangles. Figure 4.31 shows a puzzle skeleton in a puzzle, and the stretched puzzle constructed from it. We now prove $\mathcal{S}(P)$ is a puzzle.

Lemma 4.2.7. *If P is rigid, then $\mathcal{S}(P)$ is a rigid puzzle.*

Proof. Say that the puzzle skeleton in P is dark, the white case being similar. We first show that $\mathcal{S}(P)$ is a puzzle. It is easily checked that if a rhombus marked pair of triangles was changed into a rhombus piece, the rhombus mark would run parallel to the white edges of that rhombus. The rhombus marks on rhombus pieces also run parallel to white edges.

Let the rhombus marks persist in $\mathcal{S}(P)$. It is then clear from the above that every rhombus mark in $\mathcal{S}(P)$ points from a dark edge to a dark edge.

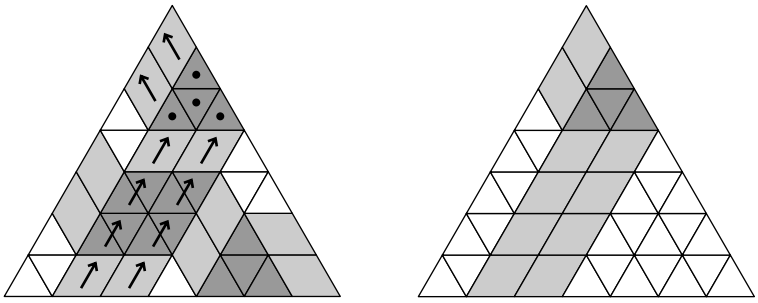


Figure 4.31: A puzzle skeleton in P, and the puzzle $\mathcal{S}(P)$.

Hence, the only dark edges in $\mathcal{S}(P)$ are between two rhombi, a dark triangle and rhombus, two dark triangles, or along the boundary. Hence, $\mathcal{S}(P)$ obeys the conditions for puzzle pieces to be adjacent.

We now show that $\mathcal{S}(P)$ is rigid. Suppose to the contrary. Then there is an evil loop in the type 1 deflation of $\mathcal{S}(P)$. Hence, there is an evil loop in the type 1 deflation of P, as $\mathcal{D}_1(\mathcal{S}(P))$ is part of the decomposition into skeletons of $\mathcal{D}_1(P)$. But this contradicts the rigidity of P. □

We note that if a dark puzzle skeleton is marked in P, then the type 1 deflation of $\mathcal{S}(P)$ is an extremal measure. This follows from Proposition 4.2.5 which says that the marked pieces in P deflate to a skeleton. It is an easy matter to then check that $\mathcal{S}(P)$ does not have any dark 4gems or dark 5gems. The white case is similar.

We construct the third puzzle as follows. The white edges in the large puzzle P are shrunk to zero length. Then the marked pieces are changed into different puzzles pieces in the same they are changed to create $\mathcal{S}(P)$. Figure 4.32 shows how this puzzle is created.

This puzzle relates to Horn inequalities. Suppose the boundary of the large puzzle P is ν, μ, λ under the bijection between $[n]_d$ and $\mathrm{Part}_{n,d}$ from Subsection 2.2.2. Further, suppose that the boundary of the smaller puzzle created as above is I, J, K. Then the partitions λ, μ, ν satisfy the Horn inequality for I, J, K with equality. That is

$$\sum_{k \in K^\vee} \lambda_k = \sum_{i \in I} \nu_i + \sum_{j \in J} \mu_j.$$

We do not prove this here. It can be deduced from the results in [2].

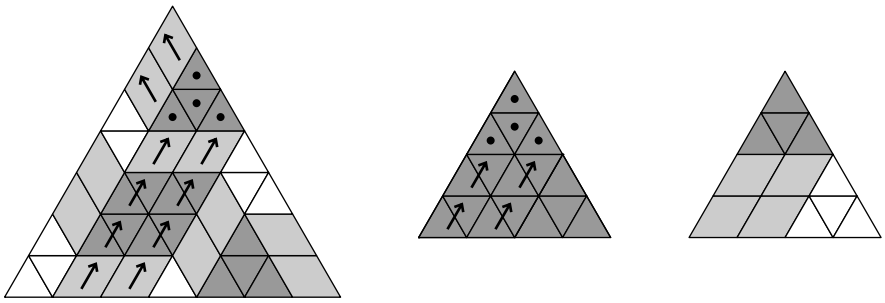


Figure 4.32: The third puzzle created from a puzzle skeleton.

4.3 Lattice Polynomial Construction

In this section, we at last describe the algorithm which constructs the lattice polynomial for a rigid puzzle. The Lattice Polynomial Algorithm is recursive, and uses the Puzzle Skeleton Algorithm as a sub-procedure. The algorithm is stated in Subsection 4.3.1. We calculate some examples in Subsection 4.3.2. In the final subsection, we prove our main result: the Lattice Polynomial Algorithm synthesizes the unique point of a triple intersection of Schubert varieties for a rigid puzzle. We now state the algorithm.

4.3.1 Lattice Polynomial Algorithm

To start, we introduce **flag labelled** puzzles. Given a puzzle P and three generic, complete flags, we write the subspaces of the flags clockwise along the boundary of P as shown in Figure 4.33.

The flag labels for the puzzles $\mathcal{S}(P)$ and $\mathcal{D}^{\circ}_{\rightarrow}(P)$ are obtained from the

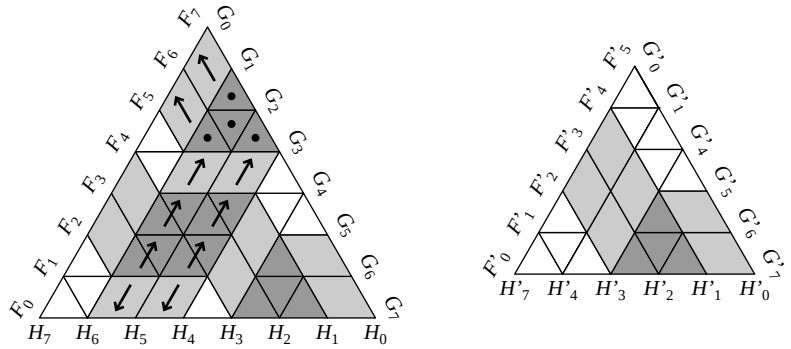


Figure 4.33: A flag labelled and rhombus marked puzzle, and $\mathcal{D}^{\circ}_{\rightarrow}(P)$ with adjusted flag labels.

labels of P in the following ways. The labels for $\mathcal{S}(P)$ are simply copied from P. For the flag labels around $\mathcal{D}^{\circ}_{\rightarrow}(P)$, we require the lattice polynomial computed by the Lattice Polynomial Algorithm for the puzzle $\mathcal{S}(P)$.

Let p be the lattice polynomial returned by the Lattice Polynomial Algorithm, and let V be the evaluation of p at the three flags, $V = p(F_\bullet, G_\bullet, H_\bullet)$. To create the flag labels for $\mathcal{D}^{\circ}_{\rightarrow}(P)$, we first change the labels of P as follows. If the puzzle skeleton in P is dark, we relabel the NW side of P with the labels $F_\bullet \vee V = (F_0 \vee V, F_1 \vee V, \ldots, F_n \vee V)$. The NE and S sides are relabelled similarly with the labels $G_\bullet \vee V$ and $H_\bullet \vee V$. If the puzzle skeleton is white, the new labels are $F_\bullet \wedge V$, $G_\bullet \wedge V$ and $H_\bullet \wedge V$. After relabelling P, create the rhombus mark deflation as normal, and at any point where multiple labels are brought together, the label with least index is kept. Figure 4.33 shows the induced flag labels after a rhombus deflation, where $F'_\bullet = F_\bullet \vee V$.

Now, the list of subspaces $F_\bullet \vee V$ is not truly a flag, as it contains repeated subspaces. However, after a rhombus mark deflation, the repeated subspaces are removed. This is proved in the proof of correctness in Section 4.3.3. Even after the deflation, the lists of subspaces around $\mathcal{D}^{\circ}_{\rightarrow}(P)$ are not honest flags in $\mathbb{C}^n$, as either the first step is not $\{0\}$ or last step is not $\mathbb{C}^n$. However, $F_\bullet \wedge V$ is a complete flag in V, and $(F_\bullet \vee V)/V$ is a complete flag in $\mathbb{C}^n/V$, and they are commonly referred to as induced flags. We prefer to work with $F_\bullet \wedge V$ and $F_\bullet \vee V$ as dishonest flags of $\mathbb{C}^n$, and we will refer to them as **induced flags**.

Next, we describe the basic puzzles where the recursion of the Lattice Polynomial Algorithm stops. A **basic puzzle** is a flag labelled puzzle where there are two tri-regions: a dark tri-shape and a white tri-shape. Figure 4.34 shows a basic puzzle P. The lattice polynomial returned for P is the flag label between the two tri-shape corners on the boundary of P, which is F_3 in Figure 4.34.

A basic puzzle has boundary $I^{\min}, I^{\max}, I^{\max}$, in some order. Suppose the

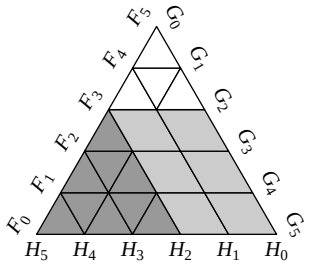


Figure 4.34: A basic puzzle, with $V = F_3$.

NW side is given by $I^{\min}$, and the NE and S sides are given by $I^{\max}$. The discussion at the start of Section 2.1 shows that the triple intersection in this case is

$$\Omega_{I^{\min}}(F_\bullet) \cap \Omega_{I^{\max}}(G_\bullet) \cap \Omega_{I^{\max}}(H_\bullet) = \{F_d\} \cap \mathrm{Gr}_d(\mathbb{C}^n) \cap \mathrm{Gr}_d(\mathbb{C}^n) = \{F_d\}.$$

The flag labels on the NW side are given by the flag $F_\bullet$, so the subspace F_d is the flag label at the point on the NW side where the white and dark tri-shapes meet, as the dark tri-shape has size d. Hence, the lattice polynomial returned by the Lattice Polynomial Algorithm is F_d, which is the unique point in the associated triple intersection of Schubert varieties. The cases when the NE or S side is given by $I^m in$ are the same, save that G_d or respectively H_d is the unique point in the intersection. The flag label returned for a basic puzzle P is called the **basic lattice polynomial** for P.

We can now state the Lattice Polynomial Algorithm.

Algorithm 4.3.1 (Lattice Polynomial Algorithm)**.** *The algorithm starts with a flag labelled puzzle P. The lattice polynomials p, q are given in terms of the flags labels $F_\bullet, G_\bullet, H_\bullet$ around P.*

1. *If P is basic, let q be the basic lattice polynomial for P.*
2. *ELSE*
   - *i) Run Puzzle Skeleton Algorithm on P.*
   - *ii) IF Puzzle Skeleton Algorithm returns EWF, terminate and return EWF.*
   - *iii) Run Lattice Polynomial Algorithm on $\mathcal{S}(P)$ with P's flag labels to get p.*
   - *iv) IF Lattice Polynomial Algorithm returns EWF, terminate and return EWF.*
   - *v) Let $V = p(F_\bullet, G_\bullet, H_\bullet)$, and let $F'_\bullet, G'_\bullet, H'_\bullet$ be the induced flag labels for $\mathcal{D}^\circ_\rightarrow(P)$.*
   - *vi) Run Lattice Polynomial Algorithm on $\mathcal{D}^\circ_\rightarrow(P)$ with the induced flag labels for $V = p$ to get q.*
   - *vii) IF Lattice Polynomial Algorithm returns EWF, terminate and return EWF.*
3. *Return q.*

4.3.2 Example Computations

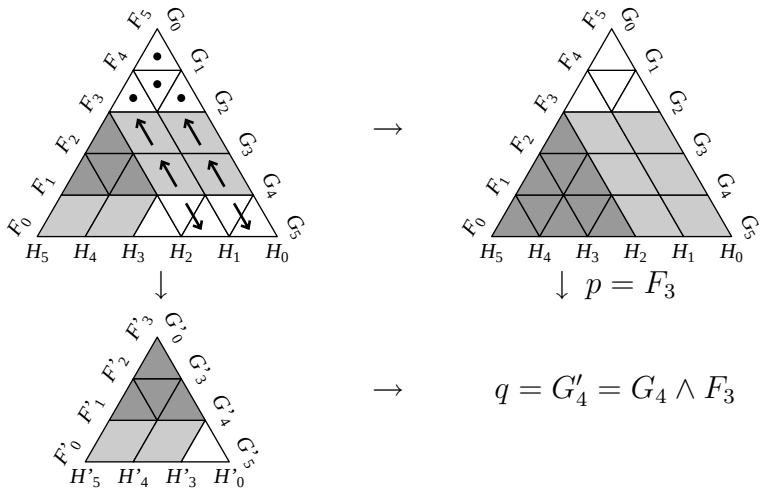


Figure 4.35: The Lattice Polynomial Algorithm on a simple example.

We now do some example computations using the Lattice Polynomial Algorithm. Figure 4.35 shows how the algorithm runs on a simple puzzle. It is fairly easy to see that the marks made in the puzzle are those that would be made by the puzzle skeleton sub-procedure. The puzzle $\mathcal{S}(P)$ is a basic puzzle with lattice polynomial $p = F_3$, so the induced flag on the NE boundary is $G'_\bullet = G_\bullet \wedge F_3$, as the puzzle skeleton is white. The puzzle $\mathcal{D}^\circ_\rightarrow(P)$ is also basic. Therefore, the Lattice Polynomial Algorithm returns $q(F_\bullet, G_\bullet, H_\bullet) = G'_4 = G_4 \wedge F_3$.

The next example builds the polynomial for the running example of this chapter. The Puzzle Skeleton Algorithm must choose a 4gem over a trap-shape as starting tri-region, so the puzzle skeleton marked in Figure 4.36 is the result of the first and only rhombus walk in the puzzle. The puzzles $\mathcal{S}(P)$ and $\mathcal{D}^\circ_\rightarrow(P)$ have polynomials $p = G_3 \wedge H_6$ and $q = F'_4 \wedge H'_3$ by a similar argument to that made for the above example. Since the puzzle skeleton is dark, $F'_\bullet = F_\bullet \vee p$, and similarly for $H'_\bullet$. Therefore, the Lattice Polynomial Algorithm returns the lattice polynomial q, for which we have

$$\begin{aligned} q(F_\bullet, G_\bullet, H_\bullet) &= F'_3 \wedge H'_4 \\ &= (F_3 \vee (G_4 \wedge H_1)) \wedge (H_4 \vee (G_4 \wedge H_1)). \end{aligned}$$

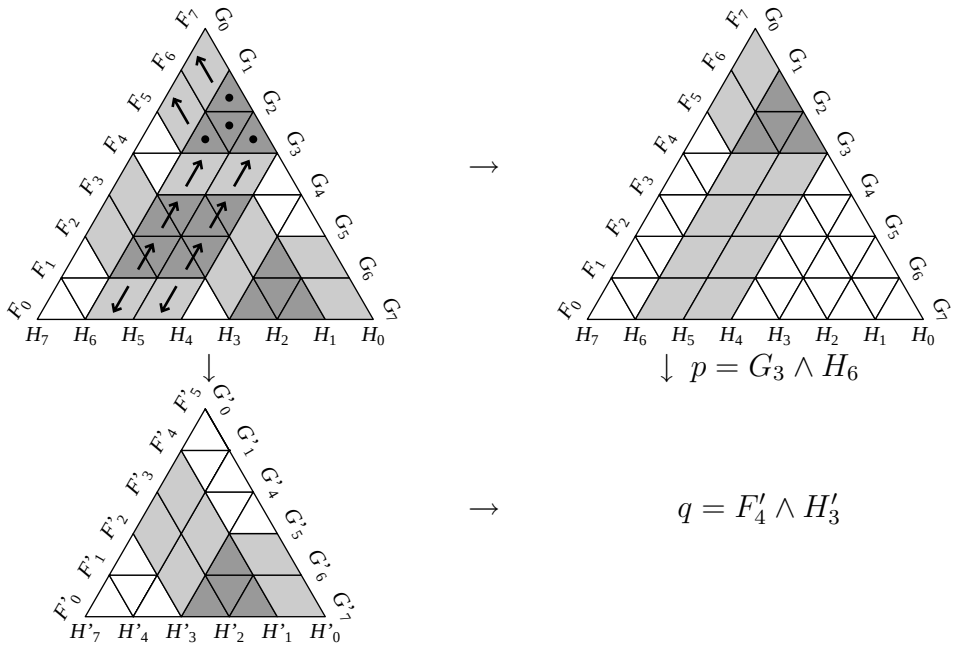


Figure 4.36: Another example of Lattice Polynomial Algorithm.

4.3.3 Termination and Correctness

In this subsection, we show that the Lattice Polynomial Algorithm terminates with a lattice polynomial satisfying the triple intersection, or it returns the symbol EWF. To show that it terminates in general, we first bound the number of recursive steps taken by the Lattice Polynomial Algorithm to show that it terminates for a rigid puzzle. The number of rhombus mark deflations is bounded by the size of P. We also need to bound the number of times the Lattice Polynomial Algorithm is run on a puzzle the same size as P. That is we need to show that for some k, $\mathcal{S}^k(P)$ is basic. We begin the proof of the first point with a lemma. The lemma is proved for a dark puzzle skeleton in P; the white case can be obtained from this argument by reflection.

Lemma 4.3.2. *Let P have a dark puzzle skeleton marked in it. If $\mathcal{S}(P)$ has a dark trap-shape, then it has a white 5gem.*

Proof. Let $Q = \mathcal{S}(P)$. The note after Lemma 4.2.7 shows that Q has no dark 4gems or 5gems, so $\mathcal{D}_1(Q)$ has no lattice points as in the diagrams:

The puzzle Q has a trap-shape, so the measure $\mathcal{D}_1(Q)$ does have a lattice point as in the following diagram.

For this proof, we call such a lattice point a *fork*, and the three edges opposite the one edge are called *tines*. The fork in $\mathcal{D}_1(Q)$ will be denoted f. The measure $\mathcal{D}_1(Q)$ is a skeleton by Proposition 4.2.5. There is some small edge e in $\mathcal{D}_1(Q)$ such that the descendance set S_e equals the support of $\mathcal{D}_1(Q)$, and there must be a descendance path from e to the center tine of f and another path to one of the outer tines of f.

The small edges of the two descendance paths form an enclosed white region, R. As the descendance paths enclose R, we see that every corner has internal angle 120° or 240°, save the corner at the fork f. Let R have minimal area among all such enclosed regions associated to forks. We claim that R has no 240° degree corners.

Suppose otherwise, and let c be such a 240° corner. An example of this situation is shown in Figure 4.37. As $\mathcal{D}_1(Q)$ is a skeleton, the small edge e in Figure 4.37 must have non-zero density by the balance conditions. The descendants of e which are not shown must 'find a way out' of R to ensure the balance conditions are met. These descendants cannot cross the borders of R by the above paragraph, so the descendants must leave by at least one of the other corners of R. But this shows there is another fork in $\mathcal{D}_1(Q)$, with an associated region that is strictly contained in R. This idea is demonstrated by extending the edge e in Figure 4.37 straight out until it leaves R. This contradicts the minimality of R.

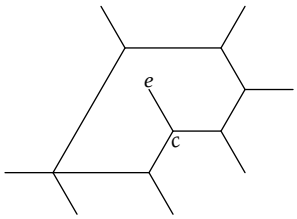


Figure 4.37: The region R with a 240° corner c, and the small edge e.

Hence, each corner of the region R has 120° internal angle, save the corner at the fork. It is an easy matter to show that R is shaped like a 5gem. Inflating $\mathcal{D}_1(Q)$, we see that Q must have a white 5gem. □

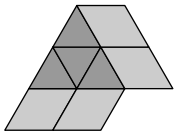
Proposition 4.3.3. *Let P have a puzzle skeleton marked in it and let $\mathcal{S}(P)$ be non-basic. Then the Puzzle Skeleton Algorithm run on $\mathcal{S}(P)$ ends with a puzzle skeleton of the opposite type to the puzzle skeleton found in P. Further, the Puzzle Skeleton Algorithm does not mark every rhombus region in $\mathcal{S}(P)$.*

Proof. We again prove the result, without loss of generality, for the case that the puzzle skeleton is dark. Let $\mathcal{S}(P)$ be the puzzle created from the marked puzzle P.

If the initial rhombus walk started in $\mathcal{S}(P)$ was started by a white walker, then all subsequent rhombus walks will be started by white walkers. In this case, the puzzle skeleton marked in $\mathcal{S}(P)$ will be white. Thus, it suffices to show that the Puzzle Skeleton Algorithm must select a white tri-region in $\mathcal{S}(P)$ to start with. Recall that the Puzzle Skeleton Algorithm first tries to find a 5gem, then a 4gem and finally a trap-shape to use as the first starting tri-region.

The note after Lemma 4.2.7 shows there are no dark 4gems or 5gems in $\mathcal{S}(P)$. By Lemma 4.3.2, if there is a dark trap-shape, then there is a white 5gem, so the algorithm will select the white 5gem over the dark trap-shape to start the Puzzle Skeleton Algorithm.

We now assume all the dark tri-regions are tri-shapes. Then there is a dark tri-region in $\mathcal{S}(P)$ that has rhombi adjacent on at least two sides as in the diagram:



The white tri-region between the rhombi has a corner with an internal angle of 120°, so the white tri-region cannot be a tri-shape. Hence, the Puzzle Skeleton Algorithm will select a white tri-region to start the first rhombus walk, so the puzzle skeleton marked in $\mathcal{S}(P)$ is white.

Now, we show that there is a rhombus region that is left unmarked by a white puzzle skeleton marked in $\mathcal{S}(P)$. Let T be the starting white tri-region of a puzzle skeleton in $\mathcal{S}(P)$. If T is a 4gem or a 5gem, then T has a bad

side b that is never stepped over by a walker leaving T. Hence, the puzzle skeleton in $\mathcal{S}(P)$ started at T never marks the rhombus region adjacent b.

If the starting tri-region T is a trap-shape, then by Lemma 4.3.2, all the dark tri-regions must be tri-shapes. We show that in this case, T cannot be completely marked by the white puzzle skeleton. To completely mark T, a series of walkers must exit one good side of T and enter by another good side. But all the dark tri-regions are tri-shapes in this case, so such a cycle would have to make a 120° turn around a dark tri-region to return to the starting trap-shape. The directions of the children of a walker w are only ever 60° from the direction of w. Therefore, the starting trap-shape cannot be completely marked by a puzzle skeleton. There must be a rhombus region adjacent to the deflated tri-region T in $\mathcal{D}^\circ_\rightarrow(P)$, and so there is an unmarked rhombus region in $\mathcal{S}(P)$, completing the proof. □

Corollary 4.3.4. *Let P be a non-basic puzzle. Then $\mathcal{S}(\mathcal{S}(P))$ has fewer rhombus regions than $\mathcal{S}(P)$.*

Proof. This follows from the fact that any puzzle skeleton in $\mathcal{S}(P)$ leaves at least one rhombus region R unmarked. The creation of $\mathcal{S}(\mathcal{S}(P))$ turns R into triangle pieces. □

Corollary 4.3.4 shows that the number of puzzles of size n that the Lattice Polynomial Algorithm is run on is bounded by the number of rhombus regions in the original puzzle P. Hence, the Lattice Polynomial Algorithm terminates for any puzzle P.

Our final theorem establishes that the algorithm does indeed produce a lattice polynomial whose evaluation at the flags $F_\bullet, G_\bullet, H_\bullet$ is the unique subspace in a triple intersection of Schubert varieties. It is followed by a corollary recording the fact that the symbol EWF is returned if and only if P is not rigid.

Theorem 4.3.5. *Let P be a puzzle with boundary I, J, K, and let q be the lattice polynomial returned by the Lattice Polynomial Algorithm. If $W = q(F_\bullet, G_\bullet, H_\bullet)$ for the generic complete flags $F_\bullet, G_\bullet$, and $H_\bullet$, then*

$$\{W\} = \Omega_I(F_\bullet) \cap \Omega_J(G_\bullet) \cap \Omega_K(H_\bullet).$$

Note that the requirement that the flags be generic ensures that the triple intersection is transverse.

Proof. We prove the stronger statement that

$$\{W\} = \Omega^\circ_I(F_\bullet) \cap \Omega^\circ_J(G_\bullet) \cap \Omega^\circ_K(H_\bullet),$$

and we proceed by induction on the number of recursive steps taken. The base case, when P is a basic puzzle, is proved in the comment following the definition of basic puzzles in Subsection 4.3.1.

Now, suppose that the Lattice Polynomial Algorithm is run on a non-basic puzzle P with boundary I, J, K. We start with the case in which the puzzle skeleton marked in P is white. Let $\mathcal{S}(P)$ have boundary $I^\mathcal{S}, J^\mathcal{S}, K^\mathcal{S}$. By induction, the Lattice Polynomial Algorithm run on $\mathcal{S}(P)$ returns a lattice polynomial $V = p(F_\bullet, G_\bullet, H_\bullet)$ such that

$$\{V\} = \Omega^\circ_{I^\mathcal{S}}(F_\bullet) \cap \Omega^\circ_{J^\mathcal{S}}(G_\bullet) \cap \Omega^\circ_{K^\mathcal{S}}(H_\bullet).$$

Hence, V obeys

$$\dim V \wedge F_x = \ell,$$

for $i^\mathcal{S}_\ell \leq x < i^\mathcal{S}_{\ell+1}$ with the convention that if $\ell + 1 > |I^\mathcal{S}|$, then $i^\mathcal{S}_{\ell+1} = n + 1$. This condition on V shows that the induced flag $V \wedge F_\bullet$ stabilizes at each of the NW boundary white edges at positions in $[n] - I^\mathcal{S}$. Since those white edges where $V \wedge F_\bullet$ stabilizes are the edges shrunk to form $\mathcal{D}^\circ_\rightarrow(P)$, the induced flag label, $F'_\bullet = V \wedge F_\bullet$, on the NW boundary of $\mathcal{D}^\circ_\rightarrow(P)$ forms a complete flag in V. We get similar conditions for $J^\mathcal{S}$ and $K^\mathcal{S}$ and their associated induced flags. Since $F_\bullet, G_\bullet$ and $H_\bullet$ are generic, the induced flags are also generic.

Let $\mathcal{D}^\circ_\rightarrow(P)$ have boundary $I^\mathcal{D}, J^\mathcal{D}, K^\mathcal{D}$. Note that $|I| = |I^\mathcal{D}| = d$, as white edges were shrunk to form $\mathcal{D}^\circ_\rightarrow(P)$. The steps of the induced flag labels around $\mathcal{D}^\circ_\rightarrow(P)$ are indexed using the indices prior to deflation, so we have that F'_{i_ℓ} is the flag label clockwise from the dark edge at position $i^\mathcal{D}_\ell$. We also have that $F'_{i_\ell} = V \wedge F_{i_\ell}$.

By induction, the Lattice Polynomial Algorithm returns a lattice polynomial $W = q(F'_\bullet, G'_\bullet, H'_\bullet)$ such that W is a subspace of V and

$$\{W\} = \Omega^\circ_{I^\mathcal{D}}(F'_\bullet) \cap \Omega^\circ_{J^\mathcal{D}}(G'_\bullet) \cap \Omega^\circ_{K^\mathcal{D}}(H'_\bullet).$$

Therefore, W obeys

$$\dim W \wedge F'_x = \ell$$

for $i_\ell \leq x < i_{\ell+1}$ and $x \notin I^\mathcal{S}$. Since the induced flag $V \wedge F_\bullet$ stabilizes at each $i^\mathcal{S} \in I^\mathcal{S}$, $W \wedge F'_x = W \wedge (V \wedge F_x)$ stabilizes also. The subspace W is a subspace of V, so we have that

$$W \wedge (V \wedge F_x) = W \wedge F_x.$$

Combining these two facts, we have that W obeys

$$\dim W \wedge F'_x = \dim W \wedge F_x = \ell$$

for $i_\ell \leq x < i_{\ell+1}$. Hence, $W \in \Omega^\circ_I(F_\bullet)$. Similar arguments establish that W is in the other two Schubert cells.

When the puzzle skeleton in P is dark, we consider the puzzle with boundary J^*, I^*, K^* that is P reflected in the vertical axis, the dark and white pieces exchanged, and the flag labels changed to $F_\bullet^\perp$, $G_\bullet^\perp$, $H_\bullet^\perp$. The puzzle skeleton is now white, so the above argument suffices to find a lattice polynomial q such that

$$q(F_\bullet^\perp, G_\bullet^\perp, H_\bullet^\perp) \in \Omega^\circ_{I^*}(F_\bullet^\perp) \cap \Omega^\circ_{J^*}(G_\bullet^\perp) \cap \Omega^\circ_{K^*}(H_\bullet^\perp).$$

To return to the I, J, K case, we define the lattice polynomial q^* by the following relations:

$$(a \vee b)^* = a^* \wedge b^*$$
$$(a \wedge b)^* = a^* \vee b^*.$$

To evaluate q^* at the flags, we have that $(F_i^\perp)^* = F_i$, which follows from the fact $\vee$ corresponds to the operation span and the $\wedge$ corresponds to intersection. Hence, there is a lattice polynomial r such that

$$q^*(F_\bullet^\perp, G_\bullet^\perp, H_\bullet^\perp) = r(F_\bullet, G_\bullet, H_\bullet).$$

It is an easy matter to check that the Lattice Polynomial Algorithm returns the lattice polynomial r, and that $W = r(F_\bullet, G_\bullet, H_\bullet)$ is the subspace satisfying

$$\Omega^\circ_I(F_\bullet) \cap \Omega^\circ_J(G_\bullet) \cap \Omega^\circ_K(H_\bullet).$$

□

Corollary 4.3.6. *Let P be a puzzle with boundary I, J, K. Then the Lattice Polynomial Algorithm returns EWF if and only if P is non-rigid.*

Proof. If the Lattice Polynomial Algorithm returns EWF, then some subprocedure returned EWF, and the results of Sections 4.1 and 4.2 show that P is not rigid.

Conversely, if the Lattice Polynomial Algorithm returns a lattice polynomial q for P, then Theorem 4.3.5 shows that $q(F_\bullet, G_\bullet, H_\bullet)$ is in the triple intersection of Schubert varieties for I, J, K and flags $F_\bullet, G_\bullet, H_\bullet$. By Theorem 2.1.4, we have that $c_{IJK} = 1$, so by the puzzle-based Littlewood-Richardson rule, P is rigid. □

Bibliography

[1] Prakash Belkale. Local systems on $\mathbb{P}^1 - S$ for S a finite set. *Compositio Math.*, 129(1):67–86, 2001.

[2] H. Bercovici, B. Collins, K. Dykema, W. S. Li, and D. Timotin. Intersections of schubert varieties and eigenvalue inequalities in an arbitrary finite factor, 2008.

[3] A. D. Berenstein and A. V. Zelevinsky. Triple multiplicities for $\mathrm{sl}(r+1)$ and the spectrum of the exterior algebra of the adjoint representation. *J. Algebraic Combin.*, 1(1):7–22, 1992.

[4] Anders Skovsted Buch. The saturation conjecture (after A. Knutson and T. Tao). *Enseign. Math. (2)*, 46(1-2):43–60, 2000. With an appendix by William Fulton.

[5] Anders Skovsted Buch, Andrew Kresch, and Harry Tamvakis. Littlewood-Richardson rules for Grassmannians. *Adv. Math.*, 185(1):80–90, 2004.

[6] William Fulton. *Young tableaux, with applications to representation theory and geometry*, volume 35 of *London Mathematical Society Student Texts*. Cambridge University Press, Cambridge, 1997.

[7] William Fulton. Eigenvalues of sums of Hermitian matrices (after A. Klyachko). *Astérisque*, (252):Exp. No. 845, 5, 255–269, 1998. Séminaire Bourbaki. Vol. 1997/98.

[8] William Fulton. *Intersection theory*, volume 2 of *Ergebnisse der Mathematik und ihrer Grenzgebiete. 3. Folge. A Series of Modern Surveys in Mathematics [Results in Mathematics and Related Areas. 3rd Series.*

A Series of Modern Surveys in Mathematics]. Springer-Verlag, Berlin, second edition, 1998.

[9] William Fulton. Eigenvalues, invariant factors, highest weights, and Schubert calculus. *Bull. Amer. Math. Soc. (N.S.)*, 37(3):209–249 (electronic), 2000.

[10] Ronald C. King, Christophe Tollu, and Frédéric Toumazet. Factorisation of Littlewood-Richardson coefficients. *J. Combin. Theory Ser. A*, 116(2):314–333, 2009.

[11] Steven L. Kleiman. The transversality of a general translate. *Compositio Math.*, 28:287–297, 1974.

[12] Alexander A. Klyachko. Stable bundles, representation theory and Hermitian operators. *Selecta Math. (N.S.)*, 4(3):419–445, 1998.

[13] Allen Knutson and Terence Tao. The honeycomb model of $\mathrm{GL}_n(\mathbf{C})$ tensor products. I. Proof of the saturation conjecture. *J. Amer. Math. Soc.*, 12(4):1055–1090, 1999.

[14] Allen Knutson and Terence Tao. Puzzles and (equivariant) cohomology of Grassmannians. *Duke Math. J.*, 119(2):221–260, 2003.

[15] Allen Knutson, Terence Tao, and Christopher Woodward. The honeycomb model of $\mathrm{GL}_n(\mathbb{C})$ tensor products. II. Puzzles determine facets of the Littlewood-Richardson cone. *J. Amer. Math. Soc.*, 17(1):19–48 (electronic), 2004.

[16] D. E. Littlewood and A. R. Richardson. Group characters and algebra. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 233:99–141, 1934.

[17] Kevin Purbhoo. Puzzles, tableaux, and mosaics. *J. Algebraic Combin.*, 28(4):461–480, 2008.

[18] Kevin Purbhoo. Personal communication. 2009.

[19] Hermann Schubert. Anzahl-bestimmungen fur lineare raume beliebiger dimension. *Acta. Math.*, 8(22):97–118, 1886.

[20] M.-P. Schützenberger. La correspondance de Robinson. In *Combinatoire et représentation du groupe symétrique (Actes Table Ronde CNRS, Univ. Louis-Pasteur Strasbourg, Strasbourg, 1976)*, pages 59–113. Lecture Notes in Math., Vol. 579. Springer, Berlin, 1977.

[21] John R. Stembridge. A concise proof of the Littlewood-Richardson rule. *Electron. J. Combin.*, 9(1):Note 5, 4 pp. (electronic), 2002.

[22] Andrei Zelevinsky. Littlewood-Richardson semigroups. In *New perspectives in algebraic combinatorics (Berkeley, CA, 1996–97)*, volume 38 of *Math. Sci. Res. Inst. Publ.*, pages 337–345. Cambridge Univ. Press, Cambridge, 1999.

Index