Title

Tableaux formulas for Lascoux polynomials

Author

Yu, Tianyi

Publication Date

2024

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA SAN DIEGO

Tableaux formulas for Lascoux polynomials

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Mathematics

by

Tianyi Yu

Committee in charge:

Professor Brendon Rhoades, Chair
Professor Russell Impagliazzo
Professor Jonathan Novak
Professor Steven V Sam
Professor Andrew Suk

2024

The dissertation of Tianyi Yu is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2024

DEDICATION

To my parents, who have always encouraged and supported me in pursuing my dreams.

To Brendon, for your unwavering support, guidance, and for always believing in me and helping me through challenges.

EPIGRAPH

Tatakae (Fight on)

— Eren Yeager, Attack on Titan

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to Professor Brendon Rhoades, my thesis advisor, reliable guide on my academic journey, invaluable life mentor, supportive confidant during my struggles, and friend. I appreciate Brendon's continuous support, patience, and encouragement during my graduate study. In the face of challenges, I often feel powerless, but after each weekly meeting, all those negative emotions vanish. Even though I often don't have any meaningful results to share during the meetings, Brendon always patiently listens to me, firmly reassures me that these are all important progress, and consistently provides useful guidance. I am deeply thankful for everything Brendon has done.

I am also grateful to Professor Mark Shimozono from Virginia Tech. I met Mark online and learned about the Lascoux polynomial from him. Mark patiently provided me with the background information I needed, and I thoroughly enjoyed exploring various combinatorial aspects of the Lascoux polynomial. He carefully reviewed my notes, offered valuable suggestions, and provided encouraging praise. Relying on these results and Mark's guidance, I embarked on my career in algebraic combinatorics.

I would like to thank my collaborators for the enjoyable and challenging work we did together on various mathematical problems. Through our collaborations, I gained valuable knowledge and learned how to approach and overcome difficulties.

I would like to thank my committee members and professors who have guided me throughout my research. Their insightful feedback and suggestions have been instrumental in advancing my understanding and shaping my work.

Finally, I must express my profound gratitude to my parents and friends for their unwavering support throughout my years of study and research. This accomplishment would not have been possible without them. Thank you.

Chapter 4, in full, is a reprint of the material as it appears in Set-valued tableaux rule for Lascoux polynomials. Tianyi Yu, Combinatorial Theory, 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 5, in full, is a reprint of the material as it appears in Grothendieck-to-Lascoux expansions. Mark Shimozono and Tianyi Yu, Transactions of the American Mathematical Society, 2023. The dissertation author was the primary investigator and author of this paper.

VITA

2020	B. S. in Mathematics and Computer Science *Summa Cum Laude*, University of California San Diego
2020-2024	Graduate Teaching Assistant, University of California San Diego
2024	Ph. D. in Mathematics, University of California San Diego

PUBLICATIONS

Jianping Pan, Tianyi Yu, "A Bijection between K-Kohnert Diagrams and Reverse Set-Valued Tableaux", *The Electronic Journal of Combinatorics* (2023): P4-26.

Jianping Pan, Tianyi Yu, "Top-degree components of Grothendieck and Lascoux polynomial", *Algebraic Combinatorics* 7.1 (2024): 109-135.

Tianyi Yu, "Set-valued tableaux rule for Lascoux polynomials", *Combinatorial Theory* 3.1 (2023).

Mark Shimozono, Tianyi Yu, "Grothendieck to Lascoux expansions", *Transactions of the American Mathematical Society*, 376 (07), 5181-5220.

Brendon Rhoades, Zehong Zhao, Tianyi Yu, "Harmonic bases for generalized coinvariant algebras", *The Electronic Journal of Combinatorics* (2020): P4-16.

ABSTRACT OF THE DISSERTATION

Tableaux formulas for Lascoux polynomials

by

Tianyi Yu

Doctor of Philosophy in Mathematics

University of California San Diego, 2024

Professor Brendon Rhoades, Chair

Lascoux polynomials simultaneously generalize two famous families of polynomials arising from geometry and representation theory: They are non-symmetric analogs of Grassmannian stable Grothendieck polynomials, which represent Schubert classes in the connective K-theory of Grassmannians. Additionally, they serve as non-homogeneous analogs of key polynomials, the characters of Demazure modules. Both of these families have classical combinatorial formulas involving tableaux. We further generalize several of

these formulas by establishing two combinatorial formulas for Lascoux polynomials.

Chapter 1

Introduction

Fix $n \in \mathbb{Z}_{>0}$. In this thesis, we establish three combinatorial rule for Lascoux polynomials using combinatorial proofs. Lascoux polynomials, denoted by $\mathfrak{L}_\alpha^{(\beta)}$, are a $\mathbb{Z}[\beta]-$basis for $\mathbb{Z}[\beta][x_1, x_2, \ldots, x_n]$ indexed by weak compositions $\alpha \in \mathbb{Z}^n_{\geq 0}$. They are related to the following polynomials:

- Schur polynomials: denoted by s_λ and indexed by partitions (weak compositions which are weakly decreasing). They are symmetric polynomials in $\mathbb{Z}[x_1, \cdots, x_n]$ which are homogeneous polynomials with degree $|\lambda|$, where $|\cdot|$ represents the sum of entries in a weak composition. The Schur polynomials play an important role in representation theory of the symmetric group and the general linear group.

- Key polynomials: denoted by κ_α, which are polynomials in $\mathbb{Z}[x_1, \cdots x_n]$ indexed by weak compositions. They are homogeneous with degree $|\alpha|$. They were introduced by Demazure in [Dem74] for Weyl groups and are characters of Demazure modules.

- Grassmannian stable Grothendieck polynomials: denoted by $G_\lambda^{(\beta)}$, which are polynomials in $\mathbb{Z}[\beta][x_1,\ldots,x_n]$ indexed by partitions. Note that the β is also an indeterminate and works as an grading variable: if a monomial in $G_\lambda^{(\beta)}$ has x-degree d, then the β in it must have degree $d-|\lambda|$. These polynomials are symmetric in the x variables. They represent Schubert classes in the connective K-theory of Grassmannians.

Notice that the β in a Lascoux polynomial $\mathfrak{L}_\alpha^{(\beta)}$ also works as a grading variable: if a monomial in $\mathfrak{L}_\alpha^{(\beta)}$ has x-degree d, then the β in it must have degree $d-|\alpha|$. The relations between these four polynomials can be described as follows.

- Key polynomials generalize Schur polynomials. More explicitly, for a partition λ,

$$\kappa_{\mathsf{rev}(\lambda)} = s_\lambda,$$

  where $\mathsf{rev}(\cdot)$ is the operator that reverses a weak composition.

- The $G_\lambda^{(\beta)}$ is a non-homogeneous analog of s_λ: The lowest degree terms in $G_\lambda^{(\beta)}$ form the Schur polynomial s_λ. In other words, $G_\lambda^{(0)} = s_\lambda$.

- Extending this viewpoint, $\mathfrak{L}_\alpha^{(\beta)}$ is a non-homogeneous analog of κ_α: $\mathfrak{L}_\alpha^{(0)} = \kappa_\alpha$.

- Lascoux polynomials generalize Grassmannian stable Grothendieck polynomials in a manner analogous to the generalization of Schur polynomials by key polynomials:

$$\mathfrak{L}_{\mathsf{rev}(\lambda)}^{(\beta)} = G_\lambda^{(\beta)},$$

Their relations are summarized in the following diagram:

$$\begin{array}{ccc} \mathfrak{L}_{\alpha}^{(\beta)} & \xrightarrow{\text{specialize}} & G_{\lambda}^{(\beta)} \\ \Big\downarrow{\scriptstyle \beta=0} & & {\scriptstyle \beta=0}\Big\downarrow \\ \kappa_{\alpha} & \xrightarrow[\text{specialize}]{} & s_{\lambda} \end{array}$$

Here is another perspective to see how the Lascoux polynomials fit into the larger picture. Lascoux and Schützenberger found an expansion of Schubert polynomials into key polynomials [LS89]. This expansion was proved by Reiner and Shimozono [RS95]. Grothendieck polynomials are K-theoretic analogs of Schubert polynomials [LS82]. Buch, Kresch, Shimozono, Tamvakis and Yong [BKS$^+$08] proved the stable limit version of this expansion. They expanded symmerized Grothendieck polynomials into Grassmannian stable Grothendieck polynomials. Finally, Reiner and Yong [RY21] conjectured an expansion of Grothendieck polynomials into Lascoux polynomials, generalizing both expansions. Shimozono and Yu [SY23] proved this conjecture.

The three polynomials in the diagram have well-known tableaux formulas which come in two different flavors. Schur polynomials are generating functions of semistandard Young tableaux (SSYT), or equivalently reversed semistandard Young tableaux (RSSYTs), of shape λ:

$$s_{\lambda} = \sum_{T \in \mathsf{SSYT}_{\lambda}} x^{\mathsf{wt}(T)} = \sum_{T \in \mathsf{RSSYT}_{\lambda}} x^{\mathsf{wt}(T)}. \tag{1.1}$$

Lascoux and Schützenberger[LS90, LS89] generalized Equation (1.1) by writing κ_{α} as a generating function of of $\mathsf{SSYT}(\alpha) \subseteq \mathsf{SSYT}_{\alpha^+}$ and $\mathsf{RSSYT}(\alpha) \subseteq \mathsf{RSSYT}_{\alpha^+}$ where α^+ is the partition obtained by sorting α:

$$\kappa_\alpha = \sum_{T \in \mathsf{SSYT}(\alpha)} x^{\mathsf{wt}(T)} = \sum_{T \in \mathsf{RSSYT}(\alpha^+)} x^{\mathsf{wt}(T)}.$$

On the other hand, Buch [Buc02] extended Equation (1.1) to $G_\lambda^{(\beta)}$ using set-valued tableaux (SVT), or equivalently reverse set-valued tableaux (RSVT):

$$G_\lambda^{(\beta)} = \sum_{T \in \mathsf{SVT}_\lambda} \beta^{\mathsf{ex}(T)} x^{\mathsf{wt}(T)} = \sum_{T \in \mathsf{RSVT}_\lambda} \beta^{\mathsf{ex}(T)} x^{\mathsf{wt}(T)}.$$

We generalize both flavors of the three formulas by writing $\mathfrak{L}_\alpha^{(\beta)}$ as generating functions of $\mathsf{SVT}(\alpha) \subseteq \mathsf{SVT}_{\alpha^+}$ and $\mathsf{RSVT}(\alpha) \subseteq \mathsf{RSVT}_{\alpha^+}$.

$$\mathfrak{L}_\alpha^{(\beta)} = \sum_{T \in \mathsf{SVT}(\alpha)} \beta^{\mathsf{ex}(T)} x^{\mathsf{wt}(T)} = \sum_{T \in \mathsf{RSVT}(\alpha)} \beta^{\mathsf{ex}(T)} x^{\mathsf{wt}(T)}.$$

Kohnert defined moves on diagrams known as *Kohnert moves*. Every weak composition α is associated with a diagram $D(\alpha)$. Let $\mathsf{KD}(\alpha)$ be the set of diagrams obtained by repeatedly applying K-Kohnert moves on $D(\alpha)$, then Kohnert showed

$$\kappa_\alpha = \sum_{D \in \mathsf{KD}(\alpha)} x^{\mathsf{wt}(D)}.$$

Ross and Yong [RY13] defined a generalization of Kohnert moves which we call *K-Kohnert moves*. Repeatedly applying K-Kohnert moves on $D(\alpha)$ yields a set of diagrams which is denoted as $\mathsf{KKD}(\alpha)$.

Conjecture 1. *[RY13] The Lascoux polynomial $\mathfrak{L}_{\alpha}^{(\beta)}$ is given by*

$$\mathfrak{L}_{\alpha}^{(\beta)} = \sum_{D \in \mathsf{KKD}(\alpha)} \beta^{\mathsf{ex}(D)} x^{\mathsf{wt}(D)}.$$

Pechenik and Scrimshaw [PS19] proved a special case of this conjecture where all positive numbers in α are the same. The third main result of this thesis is to establish this rule.

There also already exist various combinatorial formulas of Lascoux polynomials:

- Another model for key polynomials is the Kohnert diagrams developed by Kohnert [Koh90]. Ross and Yong [RY13] conjectured a generalization of Kohnert diagrams for Lascoux polynomials. Pechenik and Scrimshaw [PS19] proved a special case of this conjecture where all positive numbers in α are the same. The general case was proved by Pan and Yu [PY23].
- Buciumas, Scrimshaw and Weber [BSW20] established a set-valued skyline filling formula, which was first conjectured by Monical [Mon16].
- Buciumas, Scrimshaw and Weber [BSW20] established a SVT rule involving the right keys and the Lusztig involution, which was first conjectured by Pechenik and Scrimshaw [PS19]. In general, our rule and the rule in [BSW20] sum over different sets of SVTs.
- Presnova and Smirnov[PS23] provides a formula for Lascoux polynomials in terms of subdivisions of Gelfand–Zetlin polytopes.

Chapter 2

Tableaux formulas for key polynomials

In this chapter, we first define Lascoux polynomials and key polynomials. We then define G_λ and s_λ as special cases of Lascoux polynomials and key polynomials. Then we give necessary background and describe several classical tableaux formulas: formulas for key polynomials involving SSYT and RSSYT; formulas for $G^{(\beta)}_\lambda$ involving SVT and RSVT. The main goal of this thesis is to generalize these formulas to Lascoux polynomials.

2.1 Defining Lascoux Polynomials

The symmetric group S_n acts on the polynomial ring $\mathbb{Z}[\beta][x_1, \cdots, x_n]$ by permuting the x variables. Let $s_i \in S_n$ denote the transposition that swaps i and $i+1$. Following [LS89] and [Las01], we define three operators on $\mathbb{Z}[\beta][x_1, \cdots, x_n]$:

$$\partial_i(f) = (x_i - x_{i+1})^{-1}(f - s_i f)$$
$$\pi_i(f) = \partial_i(x_i f)$$
$$\pi_i^{(\beta)}(f) = \pi_i(f + \beta x_{i+1} f).$$

These three operators satisfy the braid relations. For instance, ∂_i satisfies:

$$\partial_i \circ \partial_j(f) = \partial_j \circ \partial_i(f) \text{ if } |i-j| > 1,$$

$$\partial_i \circ \partial_{i+1} \circ \partial_i(f) = \partial_{i+1} \circ \partial_i \circ \partial_{i+1}(f).$$

Let α be a weak composition. We use α_i to denote the i^{th} entry of α. Let $s_i\alpha$ be the weak composition obtained by swapping its i^{th} entry and $(i+1)^{th}$ entry. The *Lascoux polynomial* $\mathfrak{L}_\alpha^{(\beta)}$ is defined by [Las04]

$$\mathfrak{L}_\alpha^{(\beta)} = \begin{cases} x^\alpha & \text{if } \alpha \text{ is a partition} \\ \pi_i^{(\beta)} \mathfrak{L}_{s_i\alpha}^{(\beta)} & \text{if } \alpha_i < \alpha_{i+1}. \end{cases} \tag{2.1}$$

The *key polynomial* κ_α is defined by

$$\kappa_\alpha = \mathfrak{L}_\alpha^{(\beta)}|_{\beta=0}. \tag{2.2}$$

Example 2. We compute the Lascoux polynomial of $\alpha = (0,2,1)$.

$$\begin{aligned}
\mathfrak{L}^{(\beta)}_{(2,1,0)} &= x_1^2 x_2, \\
\mathfrak{L}^{(\beta)}_{(2,0,1)} &= \pi_2^{(\beta)}(\mathfrak{L}^{(\beta)}_{(2,1,0)}) \\
&= x_1^2 x_2 + x_1^2 x_3 + \beta x_1^2 x_2 x_3, \\
\mathfrak{L}^{(\beta)}_{(0,2,1)} &= \pi_1^{(\beta)}(\mathfrak{L}^{(\beta)}_{(2,0,1)}) \\
&= x_1^2 x_2 + x_1^2 x_3 + x_1 x_2^2 + x_1 x_2 x_3 + x_2^2 x_3 \\
&\quad + \beta(x_1^2 x_2 x_3 + x_1 x_2^2 x_3 + x_1^2 x_2^2 + x_1^2 x_2 x_3 + x_1 x_2^2 x_3) + \beta^2 x_1^2 x_2^2 x_3.
\end{aligned}$$

Thus, by setting $\beta = 0$, we obtain the key polynomial of α

$$\kappa_\alpha = x_1^2 x_2 + x_1^2 x_3 + x_1 x_2^2 + x_1 x_2 x_3 + x_2^2 x_3$$

We will use the Lascoux polynomial and key polynomial of $(0,2,1)$ as our running example to demonstrate various combinatorial objects in this chapter and the next chapter.

Instead of defining $G^{(\beta)}_\lambda$ and s_λ, we restate their relations with Lascoux polynomials $\mathfrak{L}^{(\beta)}_\alpha$ and κ_α:

For a partition λ,

$$G^{(\beta)}_\lambda := \mathfrak{L}^{(\beta)}_{\mathsf{rev}(\lambda)}, \qquad s_\lambda := \kappa_{\mathsf{rev}(\lambda)},$$

where $\mathsf{rev}(\cdot)$ reverses a weak composition.

Example 3. When $\lambda = (2,1,0)$, readers may compute:

$$\begin{aligned} s_\lambda &= x_1^2x_2 + x_1^2x_3 + x_1x_2^2 + 2x_1x_2x_3 + x_2^2x_3 + x_1x_3^2 + x_2x_3^2, \\ G_\lambda &= x_1^2x_2 + x_1^2x_3 + x_1x_2^2 + 2x_1x_2x_3 + x_2^2x_3 + x_1x_3^2 + x_2x_3^2, \\ &+\beta(x_1^2x_2^2 + x_1^2x_3^2 + x_2^2x_3^2 + 3x_1^2x_2x_3 + 3x_1x_2^2x_3 + 3x_1^2x_2x_3^2) \\ &+\beta^2(2x_1^2x_2^2x_3 + 2x_1^2x_2x_3^2 + 2x_1x_2^2x_3^2) + \beta^3x_1^2x_2^2x_3^2 \end{aligned}$$

2.2 Tableaux

Given a partition λ, a *Young diagram* of shape λ is a finite collection of left-justified boxes, where i^{th} row has λ_i boxes. We use English convention for our Young diagrams and tableaux, so the first row is the highest row. For partitions μ and λ, we write $\mu \subseteq \lambda$ if the Young diagram of μ lies in that of λ. In that case, λ/μ denotes the set of boxes in λ but not μ.

We define a *tableau* as a filling of a λ/μ with $[n] := \{1, \cdots, n\}$. A tableau has *normal* (resp. *antinormal*) shape if it is empty or has a unique northwestmost (resp. southeastmost) corner. A *semistandard Young tableau* (SSYT) (resp. *reverse semistandard Young tableau* (RSSYT)) is a tableau whose columns are strictly increasing (resp. decreasing) and rows are weakly increasing (resp. decreasing). Let T be a tableau. The *weight* of T, denoted by $\mathsf{wt}(T)$, is a weak composition whose i^{th} entry is the number of i in T. The *column order* is a total order on cells of T. It goes from left to right and from bottom to top within each column. The column word of T, denoted by $\mathsf{word}(T)$, is the word we get if we read the number in each cell of T in the column order.

Let SSYT_λ (resp. RSSYT_λ) denote the set of SSYTs (resp. RSSYTs) with shape λ. Then it is well-known that Schur polynomials are generating functions of SSYT_λ and RSSYT_λ, see (1.1).

A *key* is a SSYT with normal shape such that each number in the j^{th} column also appears in the $(j-1)^{th}$ column. There are natural bijections between weak compositions and keys. Let $\mathsf{key}(\cdot)$ be the map that sends the weak composition to its corresponding key. Its inverse map is simply $\mathsf{wt}(\cdot)$. For instance, if $n=4$,

$$\mathsf{key}(1,0,3,2) = \begin{array}{|c|c|c|}\hline 1 & 3 & 3 \\ \hline 3 & 4 & \\ \cline{1-2} 4 & & \\ \cline{1-1} \end{array}$$

The Knuth equivalence $\sim$ is defined on the set of all words over the positive integers by the transitive closure of

$$uxzyv \sim uzxyv \text{ if } x \leq y < z,$$

$$uyxzv \sim uyzxv \text{ if } x < y \leq z,$$

where u and v are words. From [Ful97], for each SSYT (resp. RSSYT) T, there exists a unique SSYT $T^{\searrow}$ with antinormal shape such that $\mathsf{word}(T) \sim \mathsf{word}(T^{\searrow})$ (resp. $\mathsf{rev}(\mathsf{word}(T)) \sim \mathsf{rev}(\mathsf{word}(T^{\searrow}))$), where $\mathsf{rev}(\cdot)$ reverses a word. Moreover, the shape of $T^{\searrow}$ is obtained by rotating the shape of T.

Each SSYT or RSSYT T with normal shape is associated with two keys: the *right key* $K_+(T)$ and the *left key* $K_-(T)$. Let $T_{\geq j}$ be the tableau we get if we remove the first

$j-1$ columns of T. Then column j of $K_+(T)$ is consists the numbers from the rightmost column of $T_{\geq j}^{\searrow}$. Let $T_{\leq j}$ be the tableau we get if we only keep the first j columns of T. Then column j of $K_-(T)$ is consists the numbers from the leftmost column of $T_{\geq j}^{\searrow}$.

Example 4. Let T be the following SSYT:

1	2	4	7
3	5	6	
4	8		
6			

Then $T_{\geq 1} = T$. Consider the following SSYT T' with antinormal shape:

			2
		3	4
	1	5	7
4	6	6	8

Notice that $\mathrm{word}(T) = 6431852647 \sim 4616538742 = \mathrm{word}(T')$, so $T' = T^{\searrow}$. Thus, column 1 of $K_+(T)$ consists of $\{2,4,7,8\}$. Similarly, $T_{\geq 2}^{\searrow}, T_{\geq 3}^{\searrow}$ and $T_{\geq 4}^{\searrow}$ are

		4
	2	7
5	6	8

	4
6	7

7

.

Thus, $K_+(T)$ is

$$\begin{array}{|c|c|c|c|}\hline 2 & 4 & 4 & 7 \\ \hline 4 & 7 & 7 & \\ \cline{1-3} 7 & 8 & & \\ \cline{1-2} 8 & & & \\ \cline{1-1}\end{array}.$$

On the other hand, $T^{\searrow}_{\leq 3}, T^{\searrow}_{\leq 2}$ and $T^{\searrow}_{\leq 1}$ are

$$\begin{array}{ccc} & & 2 \\ & 3 & 4 \\ 1 & 5 & 6 \\ 4 & 6 & 8 \end{array} \qquad \begin{array}{cc} & 2 \\ 1 & 3 \\ 4 & 5 \\ 6 & 8 \end{array} \qquad \begin{array}{c} 1 \\ 3 \\ 4 \\ 6 \end{array}.$$

Thus, $K_+(T)$ is

$$\begin{array}{|c|c|c|c|}\hline 1 & 1 & 1 & 4 \\ \hline 3 & 4 & 4 & \\ \cline{1-3} 4 & 6 & & \\ \cline{1-2} 6 & & & \\ \cline{1-1}\end{array}$$

.

Finally, we can introduce a well-known combinatorial rule of key polynomials [LS90, LS89]. Let α be a weak composition. Let $\mathsf{SSYT}(\alpha)$ (resp. $\mathsf{RSSYT}(\alpha)$) be the set of all SSYT such that T has the same shape as $\mathsf{key}(\alpha)$ and $K_+(T) \leq \mathsf{key}(\alpha)$ (resp. $K_-(T) \leq \mathsf{key}(\alpha)$) where the comparison is entry-wise.

Then

$$\kappa_\alpha = \sum_{T \in \mathsf{SSYT}(\alpha)} x^{\mathsf{wt}(T)} = \sum_{T \in \mathsf{RSSYT}(\alpha)} x^{\mathsf{wt}(T)}. \tag{2.3}$$

Example 5. Let $\alpha = (0,2,1)$. Readers may check

$$\mathsf{SSYT}(\alpha) = \left\{ \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 2 \\ \cline{1-1}\end{array}, \quad \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 3 \\ \cline{1-1}\end{array}, \quad \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 2 \\ \cline{1-1}\end{array}, \quad \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 \\ \cline{1-1}\end{array}, \quad \begin{array}{|c|c|}\hline 2 & 2 \\ \hline 3 \\ \cline{1-1}\end{array} \right\}.$$

$$\mathsf{RSSYT}(\alpha) = \left\{ \begin{array}{|c|c|}\hline 2 & 1 \\ \hline 1 \\ \cline{1-1}\end{array}, \quad \begin{array}{|c|c|}\hline 3 & 1 \\ \hline 1 \\ \cline{1-1}\end{array}, \quad \begin{array}{|c|c|}\hline 2 & 2 \\ \hline 1 \\ \cline{1-1}\end{array}, \quad \begin{array}{|c|c|}\hline 3 & 1 \\ \hline 2 \\ \cline{1-1}\end{array}, \quad \begin{array}{|c|c|}\hline 3 & 2 \\ \hline 2 \\ \cline{1-1}\end{array} \right\}.$$

Either formula yields $\kappa_\alpha = x_1^2x_2 + x_1^2x_3 + x_1x_2^2 + x_1x_2x_3 + x_2^2x_3$, which agrees with the computation in Example 2.

Remark 6. Let λ be a partition. Then consider $\mathsf{key}(\mathsf{rev}(\lambda))$. Clearly, $\mathsf{key}(rev(\lambda)) \geq T$ for any key T with shape λ. Thus, $\mathsf{SSYT}(\mathsf{rev}(\lambda)) = \mathsf{SSYT}_\lambda$ and $\mathsf{RSSYT}(\mathsf{rev}(\lambda)) = \mathsf{RSSYT}_\lambda$. This explains why (2.3) recovers (1.1).

2.3 Set-valued Tableaux

A *set-valued tableau* is a filling of λ/μ where entries are non-empty subsets of $[n]$. For a set-valued tableau T, define $\mathcal{S}(T)$ to be the set of tableaux obtained by picking one number in each cell of T. Let $\mathsf{SVT}(\lambda)$ (resp. $\mathsf{RSVT}(\lambda)$) consists of all set-valued tableaux T such that $\mathcal{S}(T) \subseteq \mathsf{SSYT}(\lambda)$ (resp. $\mathcal{S}(T) \subseteq \mathsf{RSSYT}(\lambda)$). We will refer to elements of $\mathsf{SVT}(\lambda)$ (resp. $\mathsf{RSVT}(\lambda)$) as "SVT" (resp. RSVT).

Example 7. The following T is a SVT:

$$T = \begin{array}{|c|c|c|}\hline 1 & 13 & 36 \\ \hline 23 & 47 & \multicolumn{1}{c}{} \\ \cline{1-2} 567 & \multicolumn{2}{c}{} \\ \cline{1-1} \end{array},$$

where 23 represents the set $\{2,3\}$. The set $\mathcal{S}(T)$ consists of 48 SSYT, including

$$\begin{array}{|c|c|c|}\hline 1 & 3 & 6 \\ \hline 3 & 7 & \multicolumn{1}{c}{} \\ \cline{1-2} 7 & \multicolumn{2}{c}{} \\ \cline{1-1} \end{array} \quad \text{and} \quad \begin{array}{|c|c|c|}\hline 1 & 3 & 6 \\ \hline 2 & 4 & \multicolumn{1}{c}{} \\ \cline{1-2} 5 & \multicolumn{2}{c}{} \\ \cline{1-1} \end{array}.$$

The following example is not a SVT

$$\begin{array}{|c|c|c|}\hline 1 & 14 & 46 \\ \hline 23 & 47 & \multicolumn{1}{c}{} \\ \cline{1-2} 567 & \multicolumn{2}{c}{} \\ \cline{1-1} \end{array}$$

since if we pick 4 in both cells of column 2, the resulting filling cannot be a SSYT. The following T_2 is an RSVT:

$$T_2 = \begin{array}{|c|c|c|}\hline 76 & 64 & 321 \\ \hline 4 & 31 & \multicolumn{1}{c}{} \\ \cline{1-2} 321 & \multicolumn{2}{c}{} \\ \cline{1-1} \end{array},$$

where 76 represents the set $\{7,6\}$.

Remark 8. We may view SSYT (resp. RSSYT) as a SVT (RSVT) where each set is a singleton.

Definition 9. Let T be a SVT or RSVT of shape λ. Let $\mathrm{wt}(T)$ be the weak composition whose i^{th} entry is the number of i's in T. Let $\mathsf{ex}(T)$ be the number $|\mathsf{wt}(T)| - |\lambda|$.

It is clear that the definition of $\mathsf{wt}(\cdot)$ agrees with our previous definition when every set in T is a singleton. Intuitively, $\mathsf{ex}(T)$ counts the number of "extra" numbers in T.

The notions of SVT and RSVT were first introduced by Buch [Buc02] to give a combinatorial formula for $G_\lambda^{(\beta)}$.

Theorem 10 ([Buc02])**.** *Let λ be a partition. Then*

$$G_\lambda^{(\beta)} = \sum_{T \in \mathsf{SVT}_\lambda} \beta^{\mathsf{ex}(T)} x^{\mathsf{wt}(T)} = \sum_{T \in \mathsf{RSVT}_\lambda} \beta^{\mathsf{ex}(T)} x^{\mathsf{wt}(T)}. \tag{2.4}$$

Remark 11. We know $s_\lambda = G_\lambda^{(0)}$. By setting $\beta = 0$ in (2.4), the sum only involves $T \in \mathsf{SVT}_\lambda$ or RSVT_λ such that $\mathsf{ex}(T) = 0$. In other words, the sum is only SSYT_λ or RSSYT_λ. Thus, Theorem 10 recovers (1.1).

Chapter 3

Main results

In this chapter, we describe our tableaux formulas for Lascoux polynomials and describe how they recover the corresponding classical formulas of κ_α or $G_\lambda^{(\beta)}$.

3.1 SVT formula

Let α be a weak composition. We define $\mathsf{SVT}(\alpha)$ as the set of SVT T such that $\mathcal{S}(T) \subseteq \mathsf{SSYT}(\alpha)$.

Theorem 12. *For a weak composition* α*, we have*

$$\mathfrak{L}_\alpha^{(\beta)} = \sum_{T \in \mathsf{SVT}(\alpha)} \beta^{\mathsf{ex}(T)} x^{\mathsf{wt}(T)}.$$

Example 13. Let $\alpha = (0,2,1)$. We may write $\mathsf{SVT}(\alpha)$ as

$$\mathsf{SSYT}(\alpha) \bigcup \left\{ \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 23 \\ \cline{1-1}\end{array}, \begin{array}{|c|c|}\hline 1 & 12 \\ \hline 3 \\ \cline{1-1}\end{array}, \begin{array}{|c|c|}\hline 12 & 2 \\ \hline 3 \\ \cline{1-1}\end{array}, \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 23 \\ \cline{1-1}\end{array}, \begin{array}{|c|c|}\hline 1 & 12 \\ \hline 2 \\ \cline{1-1}\end{array}, \begin{array}{|c|c|}\hline 1 & 12 \\ \hline 23 \\ \cline{1-1}\end{array} \right\}.$$

By Theorem 12, $\mathfrak{L}^{(\beta)}_{(0,2,1)} = \kappa_\alpha + \beta(2x_1^2x_2x_3 + 2x_1x_2^2x_3 + x_1^2x_2^2) + \beta^2x_1^2x_2^2x_3$, which agrees with our computation in Example 2.

Remark 14. We show how Theorem 12 recovers the SSYT rule in (2.3) and the SVT rule in (7).

- If we set $\beta = 0$ in Theorem 12, the left hand side becomes κ_α. In the right hand side, only T with $\mathsf{ex}(T) = 0$ can survive. Clearly, $\{T \in \mathsf{SVT}(\alpha) : \mathsf{ex}(T) = 0\} = \mathsf{SSYT}(\alpha)$. Thus, our rule recovers the SSYT rule in (2.3).

- Let λ be a partition. By Remark 6, $\mathsf{SSYT}(\mathsf{rev}(\lambda)) = \mathsf{SSYT}_\lambda$. Thus, we know $\mathsf{SSYT}(\mathsf{rev}(\lambda))$ consists of all T such that $\mathcal{S}(T) \subseteq \mathsf{SSYT}_\lambda$, so $\mathsf{SSYT}(\mathsf{rev}(\lambda)) = \mathsf{SVT}_\lambda$. If we set $\alpha = \mathsf{rev}(\lambda)$ in Theorem 12, the left hand side becomes $G^{(\beta)}_\lambda$ and the right hand side becomes a sum over SVT_λ. This is exactly the SVT rule in (2.4).

3.2 RSVT formula

Let T be a RSVT. We define $L(T)$ as the RSSYT obtained by keeping only the largest number in each cell. Let α be a weak composition. We define $\mathsf{RSVT}(\alpha)$ as the set of RSVT T such that $L(T) \in \mathsf{RSSYT}(\alpha)$.

Theorem 15. *For a weak composition* α*, we have*

$$\mathfrak{L}_{\alpha}^{(\beta)} = \sum_{T \in \mathsf{RSVT}(\alpha)} \beta^{\mathsf{ex}(T)} x^{\mathsf{wt}(T)}.$$

Example 16. Let $\alpha = (0,2,1)$. We may write $\mathsf{RSVT}(\alpha)$ as

$$\mathsf{RSSYT}(\alpha) \bigcup \left\{ \begin{array}{|c|c|}\hline 32 & 1 \\ \hline 1 & \\ \cline{1-1} \end{array}, \begin{array}{|c|c|}\hline 3 & 1 \\ \hline 21 & \\ \cline{1-1} \end{array}, \begin{array}{|c|c|}\hline 3 & 21 \\ \hline 2 & \\ \cline{1-1} \end{array}, \begin{array}{|c|c|}\hline 3 & 2 \\ \hline 21 & \\ \cline{1-1} \end{array}, \begin{array}{|c|c|}\hline 2 & 21 \\ \hline 1 & \\ \cline{1-1} \end{array}, \begin{array}{|c|c|}\hline 3 & 21 \\ \hline 21 & \\ \cline{1-1} \end{array} \right\}.$$

By Theorem 15, $\mathfrak{L}_{(0,2,1)}^{(\beta)} = \kappa_{\alpha} + \beta(2x_1^2x_2x_3 + 2x_1x_2^2x_3 + x_1^2x_2^2) + \beta^2 x_1^2x_2^2x_3$, which agrees with our computation in Example 2.

Remark 17. We show how Theorem 15 recovers RSSYT rule in (2.3) and the RSVT rule in (7).

- If we set $\beta = 0$ in Theorem 12, the left hand side becomes κ_{α}. In the right hand side, only T with $\mathsf{ex}(T) = 0$ can survive. Clearly, $\{T \in \mathsf{RSVT}(\alpha) : \mathsf{ex}(T) = 0\} = \mathsf{RSSYT}(\alpha)$. Thus, our rule recovers the SSYT rule in (2.3).

- Let λ be a partition. By Remark 6, $\mathsf{RSSYT}(\mathsf{rev}(\lambda)) = \mathsf{RSSYT}_{\lambda}$. Thus, we know $\mathsf{RSSYT}(\mathsf{rev}(\lambda))$ consists of all T such that $L(T) \in \mathsf{RSSYT}_{\lambda}$, so $\mathsf{RSSYT}(\mathsf{rev}(\lambda))$ $= \mathsf{RSVT}_{\lambda}$. If we set $\alpha = \mathsf{rev}(\lambda)$ in Theorem 12, the left hand side becomes $G_{\lambda}^{(\beta)}$ and the right hand side becomes a sum over RSVT_{λ}. This is exactly the RSVT rule in (2.4).

Chapter 4

Proving the SVT formula

In this chapter, we prove Theorem 12. Our proof mimics Kashiwara's study of Demazure modules and crystal bases [Kas93]. Based on our crystal, we define i-strings similar to [Kas93]. A key step of our proof is Corollary 67, which is a result analogous to [Kas93, Proposition 3.3.5].

4.1 Abstract Kashiwara crystal

We first recall some basic notions about abstract Kashiwara crystals [Kas90, Kas91] following [BS17].

Definition 18. [BS17, Definition 2.13] An *abstract Kashiwara* GL_n *crystal* is a nonempty set $\mathcal{B}$ together with maps:

$$e_i, f_i : \mathcal{B} \to \mathcal{B} \sqcup \{\mathbf{0}\},$$

$$\varepsilon_i, \varphi_i : \mathcal{B} \to \mathbb{Z} \sqcup \{-\infty\},$$

$$\mathsf{wt} : \mathcal{B} \to \mathbb{Z}^n,$$

where $i \in [n-1]$, satisfying the following two conditions.

K1: For all $X, Y \in \mathcal{B}$, we have $e_i(X) = Y$ if and only if $f_i(Y) = X$. If this is the case then

$$\varepsilon_i(Y) = \varepsilon_i(X) - 1,$$
$$\varphi_i(Y) = \varphi_i(X) + 1,$$
$$\mathsf{wt}(Y) = \mathsf{wt}(X) + v_i - v_{i+1},$$

where $v_1, \ldots, v_n$ is the standard basis of $\mathbb{Z}^n$.

K2: For all $X \in \mathcal{B}$, we have

$$\varphi_i(X) = \langle \mathsf{wt}(X), v_i - v_{i+1} \rangle + \varepsilon_i(X).$$

Furthermore, $\mathcal{B}$ is called *seminormal* if

$$\varepsilon_i(X) = \max\{k : e_i^k(X) \neq \mathbf{0}\} \quad \text{and} \quad \varphi_i(X) = \max\{k : f_i^k(X) \neq \mathbf{0}\}$$

for all $X \in \mathcal{B}$ and $i \in B_{n-1}$.

Definition 19. [Kas93] Let $\mathcal{B}$ be an abstract Kashiwara GL_n-crystal. For each $i \in [n-1]$, an *i-string* is a sequence $X_0, \ldots, X_k \in \mathcal{B}$ satisfying:

- $e_i(X_0) = f_i(X_k) = \mathbf{0}$
- $f_i(X_j) = X_{j+1}$ for each $j \in \{0, 1, \ldots, k-1\}$.

We say X_0 is the *source* of its string. Diagrammatically, we can represent the string as:

$$X_0 \xrightarrow{\quad i \quad} X_1 \xrightarrow{\quad i \quad} X_2 \xrightarrow{\quad i \quad} \cdots \xrightarrow{\quad i \quad} X_k$$

It is clear that $\mathcal{B}$ can be broken into a disjoint union of i-strings for each i. If we know $\mathcal{B}$ is seminormal, then we have the following well-known result regarding the weight of elements in an i-string.

Lemma 20. *[BS17, Proposition 2.36] Let $\mathcal{B}$ be a seminormal abstract Kashiwara GL_n-crystal. Consider the i-string $X_0, \ldots, X_k$ for some $i \in [n-1]$. Then $\mathsf{wt}(X_j) = s_i \mathsf{wt}(X_{k-j})$ for each $j \in \{0, 1, \ldots, k\}$, where s_i is the operator that swaps the i^{th} entry and the $(i+1)^{th}$ entry.*

Now we describe a well-known example of an abstract Kashiwara crystal. Take $T \in \mathsf{SSYT}_\lambda$ and consider its column word. We replace each i by ")" and replace each $i+1$ by "(". Then we remove all other numbers. The resulting word is called the *i-word* of T. We may pair "(" with ")" in the usual way.

Definition 21. Define $\varepsilon_i(T)$ as the number of unpaired "(" and $\varphi_i(T)$ as the number of unpaired ")".

If $\varphi_i(T) = 0$, then $f_i(T) := \mathbf{0}$. Otherwise, we can find the i in T that corresponds to the last unpaired ")" in the i-word. We change this i into $i+1$ and get $f_i(T)$.

If $\varepsilon_i(T) = 0$, then $e_i(T) := \mathbf{0}$. Otherwise, we can find the $i+1$ in T that corresponds to the first unpaired "(" in the i-word. We change this $i+1$ into i and get $e_i(T)$.

It is a well-known result that SSYT_λ, together with $e_i, f_i, \varphi_i, \varepsilon_i$ and wt, form a seminormal abstract Kashiwara GL_n-crystal. Moreover, they correspond to the crystal basis of the irreducible highest weight $U_q(\mathfrak{gl_n})$ module of highest weight λ.

We can use the operator f_i to compute $\mathsf{SSYT}(\alpha)$. Let S be a subset of SSYT_λ. Define $\mathcal{F}_i S$ as $\{(f_i)^j(T) : T \in S, j \geq 0\} - \{\mathbf{0}\}$.

Theorem 22 ([Kas93]). *Let α be a weak composition such that $\alpha^+ = \lambda$ and $\alpha_i > 0$ for $i > n$. We can write α as $s_{i_1} \dots s_{i_k}\lambda$, where k is minimized. Then we have*

$$\mathsf{SSYT}(\alpha) = \mathcal{F}_{i_1} \dots \mathcal{F}_{i_k}\{u_\lambda\}.$$

Here, u_λ is the SSYT with shape λ such that its r^{th} row only has r.

The set $\mathsf{SSYT}(\alpha)$, together with the maps, is known as a *Demazure crystal*.

4.2 The right keys

In this section, we first describe a direct way to compute $K_+(T)$ for SSYT T with normal shape. Then we generalize the right key to all SVT with normal shape.

4.2.1 Compute right keys using the star operator

We use the following operator to compute right keys. This method is a reformulation of Willis' method [Wil13].

Definition 23. First, we define $S \star m$ for $S \subseteq \mathbb{Z}$ and $m \in \mathbb{Z}$. Let m' be the largest number in S such that $m' \leq m$. If m' does not exist, we let $S \star m = S \sqcup \{m\}$. Otherwise, we define $S \star m = (S - \{m'\}) \sqcup \{m\}$.

More generally, we may define $\star$ to be a right action of the free monoid of words with characters in the set $\mathbb{Z}$, on the power set of $\mathbb{Z}$. If $w = w_1 \cdots w_n$ is a word of integers, we define $S \star w = (\cdots((S \star w_1) \star w_2) \cdots \star w_n)$, and $S \star w = S$ if w is the empty word.

Example 24. We have

$$\{2,4,5,7\} \star 3462 = \{2,3,4,6,7\},$$

$$\{2,4,5,7\} \star 1284 = \{1,2,4,5,8\}.$$

Lemma 25. *We have $S \star w = S \star w'$, if w and w' are Knuth equivalent.*

Proof. Routine case studies. □

We have the following way to compute a right key.

Lemma 26. *Column j of $K_+(T)$ consists of $\emptyset \star \mathsf{word}(T_{\geq j})$.*

Proof. By definition, column j of $K_+(T_{\geq j})$ equals the last column of $T^{\searrow}_{\geq j}$. Since $T^{\searrow}_{\geq j}$ has antinormal shape, $\emptyset \star \mathsf{word}(T^{\searrow}_{\geq j})$ is the set of numbers in the last column of $T^{\searrow}_{\geq j}$. Then the proof is finished by $\mathsf{word}(T_{\geq j}) \sim \mathsf{word}(T^{\searrow}_{\geq j})$ and Lemma 25. □

Example 27. Let T be the following SSYT:

1	2	4	7
3	5	6	
4	8		
6			

Then column 1 of $K_+(T)$ consists of $\emptyset \star 6431852647 = \{2,4,7,8\}$. Column 2, 3 and 4 of

$K_+(T)$ consist of: $\emptyset \star 852647 = \{4,7,8\}$, $\emptyset \star 647 = \{4,7\}$ and $\emptyset \star 7 = \{7\}$. Thus, $K_+(T)$ is

$$\begin{array}{|c|c|c|c|} \hline 2 & 4 & 4 & 7 \\ \hline 4 & 7 & 7 & \multicolumn{1}{c}{} \\ \cline{1-3} 7 & 8 & \multicolumn{2}{c}{} \\ \cline{1-2} 8 & \multicolumn{3}{c}{} \\ \cline{1-1} \end{array}$$

which agrees with Example 4.

4.2.2 Generalizing $K_+(\cdot)$ to SVT

In this subsection, we assign a SSYT to each SVT with normal shape. Then we explains why this assignment naturally generalizes $K_+(\cdot)$.

Definition 28. Let T be a SVT with normal shape. Define

$$T_{\max} := \max_{P \in \mathcal{S}(T)} (K_+(P))$$

where max is entry-wise.

Example 29. We start with the SVT T. The set $\mathcal{S}(T)$ has two SSYT.

$$T = \begin{array}{|c|c|} \hline 1 & 23 \\ \hline 3 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}, \quad \mathcal{S}(T) = \{ \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 3 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}, \quad \begin{array}{|c|c|} \hline 1 & 3 \\ \hline 3 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array} \}.$$

We compute the right keys of the two tableaux in $\mathcal{S}(T)$ and get:

$$\begin{array}{|c|c|} \hline 2 & 2 \\ \hline 3 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array} \quad \text{and} \quad \begin{array}{|c|c|} \hline 1 & 3 \\ \hline 3 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}.$$

Take the maximum of each entry and obtain:

$$T_{\max} = \begin{array}{|c|c|}\hline 2 & 3 \\ \hline 3 & \\ \cline{1-1}\end{array}.$$

Remark 30. Readers might wonder whether $T_{\max}$ can be computed as follows: Pick the largest number in each entry and compute the right key of this SSYT. The previous example shows that this approach does not work. If we pick the largest number in each entry, we obtain

$$\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 3 & \\ \cline{1-1}\end{array}$$

whose right key is not $T_{\max}$.

From the definition of $T_{\max}$, it is an entry-wise maximum of several SSYT. Thus, $T_{\max}$ is also a SSYT. Next, we find an easier way to compute $T_{\max}$ and show it is a key. We start with a definition.

Definition 31. For finite $S \subseteq \mathbb{Z}_{>0}$, let $\mathsf{word}(S)$ be the word we get if we list numbers of S in increasing order. For a SVT T, let $\mathsf{word}(T) := \mathsf{word}(S_1)\cdots\mathsf{word}(S_n)$, where $S_1, \dots, S_n$ are entries of T in the column order.

Now we may introduce an easier way to compute $T_{\max}$:

Lemma 32. *Let T be a normal SVT. Column j of $T_{\max}$ consists of $\emptyset \star \mathsf{word}(T_{\geq j})$, where $T_{\geq j}$ is obtained by removing the first $j-1$ columns of T.*

Example 33. Let T be the SVT in example 7. Then $\mathsf{word}(T) = 567231471336$. Column 1 of $T_{\max}$ consists of $\emptyset \star 567231471336 = \{3,6,7\}$. Column 2 and 3 of $T_{\max}$ consist of

$\emptyset \star 471336 = \{6,7\}$ and $\emptyset \star 36 = \{6\}$. Thus,

$$T_{\max} = \begin{array}{|c|c|c|} \hline 3 & 6 & 6 \\ \hline 6 & 7 & \multicolumn{1}{c}{} \\ \cline{1-2} 7 & \multicolumn{2}{c}{} \\ \cline{1-1} \end{array} .$$

We may check this agrees with the definition of $T_{\max}$. First, we compute the right keys of the two SSYT from $\mathcal{S}(T)$ in example 7. We get

$$\begin{array}{|c|c|c|} \hline 1 & 6 & 6 \\ \hline 6 & 7 & \multicolumn{1}{c}{} \\ \cline{1-2} 7 & \multicolumn{2}{c}{} \\ \cline{1-1} \end{array} \quad \text{and} \quad \begin{array}{|c|c|c|} \hline 3 & 3 & 6 \\ \hline 4 & 6 & \multicolumn{1}{c}{} \\ \cline{1-2} 6 & \multicolumn{2}{c}{} \\ \cline{1-1} \end{array} ,$$

whose entry-wise maximum is the key above. The right keys of the other 46 SSYT in $\mathcal{S}(T)$ are entry-wise less than or equal to this key.

To prove the lemma, we need the entry-wise maximum of sets:

Definition 34. Let C be a finite collection of sets such that all sets in C have the same size k. We may view each element of C as a column of a SSYT and take the entry-wise maximum. Then $\max_{S \in C} S$ is the set corresponding to the resulting column. More explicitly, $\max_{S \in C} S$ is the set with size k such that its i^{th} smallest number is

$$\max_{S \in C} (i^{th} \text{ smallest number in } S).$$

Proof of Lemma 32. It is enough to assume $j = 1$. By the definition of $T_{\max}$ and Lemma 26, column 1 of $T_{\max}$ consists of $\max_{P\in\mathcal{S}(T)} \emptyset \star \mathsf{word}(P)$. Thus, we need to prove

$$\max_{P\in\mathcal{S}(T)} \emptyset \star \mathsf{word}(P) = \emptyset \star \mathsf{word}(T) \tag{4.1}$$

First, we prove (4.1) for T that only has one column. Let $S_1,\ldots,S_k$ be the entries of T, enumerated from bottom to top. We have $\min(S_i) > \max(S_{i+1})$ for $1 \le i \le k-1$. As P ranges over $\mathcal{S}(T)$, $\mathsf{word}(P)$ ranges over $s_1 \cdots s_k$ with $s_i \in S_i$. Thus, $\emptyset \star \mathsf{word}(P)$ ranges over $\{s_1 > \cdots > s_k\}$ with $s_i \in S_i$. The left hand side of (4.1) is $\{\max(S_1) > \cdots > \max(S_k)\}$. For the right hand side, notice that $\emptyset \star \mathsf{word}(S_1) = \{\max(S_1)\}$. Since $\max(S_1) > \max(S_2)$, $\emptyset \star \mathsf{word}(S_1)\mathsf{word}(S_2) = \{\max(S_1), \max(S_2)\}$. A simple induction on i would yield

$$\emptyset \star \mathsf{word}(S_1)\cdots\mathsf{word}(S_i) = \{\max(S_1) > \cdots > \max(S_i)\}.$$

Since $\mathsf{word}(T) = \mathsf{word}(S_1)\cdots\mathsf{word}(S_k)$, the right hand side of (4.1) is $\{\max(S_1) > \cdots > \max(S_k)\}$. We have established (4.1) for T with one column.

Now we prove (4.1) for all SVT T. We perform an induction on the number of entries of T that are not in column 1. For the base case, we assume T has no such entries. In other words, T has only one column. This case is checked above.

Now assume T has more than one column. Let X be the highest entry in the rightmost column of T. We may remove X from T and raise all entries below X. The resulting filling, T', is clearly a SVT. We have

$$\mathsf{word}(T) = \mathsf{word}(T')\mathsf{word}(X) \qquad \text{and}$$

$$\{\mathsf{word}(P) : P \in \mathcal{S}(T)\} = \{\mathsf{word}(P')x : P' \in \mathcal{S}(T'), x \in X\}.$$

Our goal (4.1) becomes

$$\max_{P' \in \mathcal{S}(T'), x \in X} \emptyset \star \mathsf{word}(P')x = \emptyset \star \mathsf{word}(T')\mathsf{word}(X). \tag{4.2}$$

To show this equality, we first find an alternative way to write its right hand side. By the inductive hypothesis,

$$\max_{P' \in \mathcal{S}(T')} \emptyset \star \mathsf{word}(P') = \emptyset \star \mathsf{word}(T').$$

Use $\{a_1 < \cdots < a_k\}$ to denote $\emptyset \star \mathsf{word}(T')$. We know $k = |\emptyset \star \mathsf{word}(P')|$ for any $P' \in \mathcal{S}(T')$. Thus, k is the number of entries in column 1 of $K_+(P')$, which is also the number of rows in T' and T. Consequently, $k = |\emptyset \star \mathsf{word}(P)|$ for any $P \in \mathcal{S}(T)$.

Next, we show $\min(X) \geq a_1$ by contradiction. Assume there exists $x \in X$ with $x < a_1$. We may pick $P' \in \mathcal{S}(T')$ such that $\min(\emptyset \star \mathsf{word}(P')) = a_1$. Then consider the tableau $P \in \mathcal{S}(T)$ with $\mathsf{word}(P) = \mathsf{word}(P')x$. We have $\emptyset \star \mathsf{word}(P) = (\emptyset \star \mathsf{word}(P')) \star x$, which has more than k numbers. Contradiction.

Since $\min(X) \geq a_1$, we may partition X as $X_1 \sqcup \cdots \sqcup X_k$ by $X_i = X \cap [a_i, a_{i+1})$, where $a_{k+1} = \infty$ by convention. Consider the action of $\mathsf{word}(X) = \mathsf{word}(X_1) \cdots \mathsf{word}(X_k)$ on $\{a_1, \ldots, a_k\}$. When X_i acts, a_i is still in the set. If X_i is non-empty, a_i will be bumped by $\min(X_i)$, which is then bumped by the second smallest number in X_i. Eventually, the action of X_i replaces a_i by $\max(X_i)$. Thus, $\{a_1, \ldots, a_k\} \star \mathsf{word}(X) = \{\overline{a}_1 < \cdots < \overline{a}_k\}$, where $\overline{a}_i = \max(X_i)$ if $X_i \neq \emptyset$ and $\overline{a}_i = a_i$ otherwise.

We have turned the right hand side of (4.2) into $\{\overline{a}_1 < \cdots < \overline{a}_k\}$. It remains to establish the following two statements:

- For any $P' \in \mathcal{S}(T'), x \in X$ and $1 \leq i \leq k$, the i^{th} smallest number of $\emptyset \star \mathsf{word}(P')x$ is at most $\overline{a}_i$.

- For any $1 \leq i \leq k$, we may find $P' \in \mathcal{S}(T')$ and $x \in X$ such that the i^{th} smallest number of $\emptyset \star \mathsf{word}(P')x$ achieves $\overline{a}_i$.

Now we prove these two claims.

- Take any $P' \in \mathcal{S}(T')$ and $x \in X$. Let $\{b_1 < \cdots < b_k\} = \emptyset \star \mathsf{word}(P')$. Our inductive hypothesis implies $b_i \leq a_i$ for all $1 \leq i \leq k$. Now, assume x bumps b_j when acting on $\{b_1 < \cdots < b_k\}$, becoming the j^{th} smallest number in the resulting set. We only need to check $x \leq \overline{a}_j$. Notice that $x < b_{j+1} \leq a_{j+1}$ with $b_{k+1} = \infty$ by convention. Thus, $x \in X_1 \sqcup \cdots \sqcup X_j$. If $X_j \neq \emptyset$, $\overline{a}_j = \max(X_j) \geq x$. Otherwise, $x \in X_1 \sqcup \cdots \sqcup X_{j-1}$, so $x < a_j = \overline{a}_j$.

- Take $1 \leq i \leq k$. First, assume $X_i \neq \emptyset$. By the inductive hypothesis, we may pick $P' \in \mathcal{S}(T')$ such that if we let $\{b_1 < \cdots < b_k\} = \emptyset \star \mathsf{word}(P')$, then $b_{i+1} = a_{i+1}$. Pick $x = \max(X_i)$, so $b_{i+1} = a_{i+1} > x \geq a_i \geq b_i$. When x acts on $\{b_1 < \cdots < b_k\}$, it will bump the b_i. The i^{th} smallest number in the resulting set is $x = \overline{a}_i$. Finally, assume $X_i = \emptyset$, so $\overline{a}_i = a_i$. Pick $P' \in \mathcal{S}(T')$ such that if we let $\{b_1 < \cdots < b_k\} = \emptyset \star \mathsf{word}(P')$, then $b_i = a_i$. Pick any $x \in X$. If $x < a_i$, x will not bump b_i when acting on $\{b_1 < \cdots < b_k\}$. Otherwise, we know $x \geq a_{i+1}$ because $X_i = \emptyset$. Since $b_{i+1} \leq a_{i+1} \leq x$, x will not bump b_i when acting on $\{b_1 < \cdots < b_k\}$. In either case, the i^{th} largest number of $\{b_1 < \cdots < b_k\} \star x$ remains to be $b_i = \overline{a}_i$. □

Corollary 35. *$T_{\max}$ is a key .*

Proof. Let j be a positive integer. By Lemma 32, it remains to show

$$\emptyset \star \mathsf{word}(T_{\geq j}) \supseteq \emptyset \star \mathsf{word}(T_{\geq j+1}).$$

It remains to show $S \star j \supseteq S' \star j$ if $S \supseteq S'$, which is a routine case study. □

Definition 36. The right key of a SVT T is $K_+(T) := T_{\max}$.

Remark 37. From the definition, it is clear that $\mathsf{SVT}(\alpha) = \{T : K_+(T) \leq \mathsf{key}(\alpha)\}$.

4.3 Abstract Kashiwara crystals on SVTs

To prove our SVT rule, we construct an abstract Kashiwara GL_n-crystal on the set of SVTs.

4.3.1 Constructing an abstract Kashiwara crystal on SVT

The goal of this subsection is to turn SVT_λ into an abstract Kashiwara GL_n-crystal. First, we let $\mathsf{wt}(\cdot)$ be the weight function on SVT defined earlier. To define the maps ϕ_i, ε_i, f_i and e_i, we need to generalize the i-word defined on SSYT.

Definition 38. Take $T \in \mathsf{SVT}_\lambda$ and $i \in [n-1]$. The *i-word* of T is a word built by "(", ")", and ")−(" under concatenation. It is created as follows.

Read through entries of T in the column order. Whenever we see a set containing i but not $i+1$, we write ")". Whenever we see a set containing $i+1$ but not i, we write "(". Whenever we see a set containing i and $i+1$, we write ")−(".

Example 39. Consider the following element from $\mathcal{B}_4$

$$T = \begin{array}{|c|c|c|c|}\hline 1 & 1 & 2 & 23 \\ \hline 2 & 23 & \multicolumn{2}{c}{} \\ \cline{1-2} 34 & \multicolumn{3}{c}{} \\ \cline{1-1} \end{array}.$$

It has 1-word $()()(($. It has 2-word $())-())-($. It has 3-word $)-((($.

Take $T \in \mathsf{SVT}_\lambda$. Next, we describe a way to break the i-word of T into continuous sub-words. Ignore all "$-$" and pair the "(" with ")" in the usual way. Then we construct an equivalence relation on all characters. This relation is generated by the following two requirements.

- If an "(" is paired with ")", then these two characters and everything between them should be in the same equivalence class.
- For each "$)-($", these three characters are in the same equivalence class.

It is easy to see that each equivalence class is a contiguous sub-word.

Example 40. Assume a SVT has i-word $))-(())-())-(()-($. Then it is partitioned into four equivalence classes:

$$) \qquad)-(())-() \qquad)-(\qquad ()-($$

Notice that any unpaired ")" must be the first character in its class. Any unpaired "(" must be the last character in its class. Thus, we may classify each class by whether it starts with an unpaired ")" and whether it ends with an unpaired "(".

- *null form*: This class does not have unpaired "(" or ")". For example, "$(()-())-()$".

- *left form*: This class does not have unpaired ")" but ends with an unpaired "(". This class is either "(" or "$(u)-($" for some word u. For example, "$(())-($" is in this class.

- *right form*: This class does not have unpaired "(" but starts with an unpaired ")". This class is either ")" or "$)-(u)$" for some word u. For example, "$)-()-()$" is in this class.

- *combined form*: This class start with an unpaired ")" and ends with an unpaired "(". This class is either "$)-($" or "$)-(u)-($" for some word u. For example, "$)-()-(())-($" is in this class.

In Example 40, the first two classes are right forms. The third class is a combined form and the last class is a left form. In general, if we ignore the null-forms in a word, then we have several right forms, followed by zero or one combined form, followed by several left forms. This idea allows us to define φ_i and ε_i on SVT_λ.

Definition 41. Take $T \in \mathsf{SVT}_\lambda$ and take $i \in [n-1]$. Let $\varphi_i(T)$ (resp. $\varepsilon_i(T)$) be the number of right forms (resp. left forms) in the i-word of T.

Then we can check they satisfy the condition (K2) in Definition 18.

Lemma 42. *Take $T \in B_n$ and $i \in [n-1]$. Then*

$$\varphi_i(T) - \varepsilon_i(T) = \langle \mathsf{wt}(T), v_i - v_{i+1} \rangle,$$

where $v_1, \ldots, v_n$ is the standard basis of $\mathbb{Z}^n$.

Proof. Consider the i-word of T. The left hand side is the number of right forms minus the number of left forms. Next, observe the following.

- In each right form, there is one more ")" than "(".
- In each left form, there is one more "(" than ")".
- In each combined form or null form, the numbers of "(" and ")" are equal.

Thus, $\varphi_i(T) - \varepsilon_i(T)$ is also the number of ")" minus the number of "(" in the i-word of T. Correspondingly, it is the number of i in T minus the number of $i+1$ in T, which is $\langle \mathsf{wt}(T), v_i - v_{i+1} \rangle$. □

To define f_i and e_i, we first define operators f_i' and e_i' on $\mathsf{SVT}_\lambda \sqcup \{\mathbf{0}\}$. They can be viewed as "square roots" of f_i and e_i: Later, we will define $f_i(T)$ as $f_i'(f_i'(T))$ and $e_i(T)$ as $e_i'(e_i'(T))$.

Definition 43. Define f_i', e_i' on $\mathsf{SVT}_\lambda \cup \{\mathbf{0}\}$. First, $f_i'(\mathbf{0}) = e_i'(\mathbf{0}) = \mathbf{0}$. Now take $T \in \mathsf{SVT}_\lambda$. To define $f_i'(T)$, consider the following cases.

- Case 1: If its i-word has a combined form, we find the entry in T that corresponds to ") – (" in the beginning of this combined form. We remove i from this entry and obtain $f_i'(T)$.
- Case 2: Otherwise, if its i-word has no right forms, we set $f_i'(T) = \mathbf{0}$.
- Case 3: Otherwise, find the last right form in its i-word. Find the entry in T that corresponds to ")" at the end of this right form. Add $i+1$ to this entry and obtain $f_i'(T)$.

To define $e'_i(T)$, consider the following cases.

- Case 1: If its i-word has a combined form, we find the entry in T that corresponds to "$)-($" in the end of this combined form. We remove $i+1$ from this entry and obtain $e'_i(T)$.

- Case 2: Otherwise, if its i-word has no left forms, we set $e'_i(T) = \mathbf{0}$.

- Case 3: Otherwise, find the first left form in its i-word. Find the entry in T that corresponds to "$($" at the start of this left form. Add i to this entry and obtain $e'_i(T)$.

Example 44. Consider T in Example 39. Its 2-word "$())-())-($" has a null form "$()$", a right form "$)-()$" and a combined form "$)-($". The "$)-($" in the beginning of this combined form corresponds to the entry in column 4 of T. We remove the 2 in it and obtain $f'_2(T)$.

$$T = \begin{array}{|c|c|c|c|}\hline 1 & 1 & 2 & 23 \\ \hline 2 & 23 \\ \cline{1-2} 34 \\ \cline{1-1}\end{array} \quad \xrightarrow{f'_2} \quad \begin{array}{|c|c|c|c|}\hline 1 & 1 & 2 & 3 \\ \hline 2 & 23 \\ \cline{1-2} 34 \\ \cline{1-1}\end{array} = f'_2(T).$$

Now $f'_2(T)$ has 2-word "$())-()($". It has no combined form and the last right form is "$)-()$". The "$)$" at the end of this right form corresponds to the entry in column 3 of $f'_2(T)$. We add a 3 to it and obtain $f'_2(f'_2(T))$.

$$f'_2(T) = \begin{array}{|c|c|c|c|}\hline 1 & 1 & 2 & 3 \\ \hline 2 & 23 \\ \cline{1-2} 34 \\ \cline{1-1}\end{array} \quad \xrightarrow{f'_2} \quad \begin{array}{|c|c|c|c|}\hline 1 & 1 & 23 & 3 \\ \hline 2 & 23 \\ \cline{1-2} 34 \\ \cline{1-1}\end{array} = f'_2(f'_2(T)).$$

Before further investigating f'_i and e'_i, we need to make sure when they do not yield $\mathbf{0}$, the resulting tableau is indeed a SVT.

Lemma 45. *For any $T \in \mathsf{SVT}_\lambda$ and $i \in [n-1]$, $f_i'(T), e_i'(T) \in \mathsf{SVT}_\lambda \sqcup \{\mathbf{0}\}$.*

Proof. We check $f_i'(T) \in \mathsf{SVT}_\lambda \sqcup \{\mathbf{0}\}$. The proof for $e_i'(T)$ is similar. Assume $f_i'(T) \neq \mathbf{0}$ and consider what f_i' does on T. If it removes an i from an entry of T, we know that entry corresponds to an ")$-$(" in the i-word of T. Thus, this entry has both i and $i+1$. Removing i from it will yield a valid SVT.

Now, assume f' adds an $i+1$ to an entry S in T. We know S corresponds to an ")" in the i-word of T. Moreover, it is the last character in the last right form. We know S contains i but not $i+1$. Now let $S_\downarrow$ (resp. $S_\rightarrow$) be the entry below S (resp. right of S) in T. We need to check the following two statements.

- The entry $S_\downarrow$, if exists, has no $i+1$: Assume it is not true. Clearly, i is not in $S_\downarrow$, so $S_\downarrow$ corresponds to an "(" in the i-word of T. It is immediately before the ")" that corresponds to S. Then this ")" cannot be the last character in a right form. Contradiction.

- The entry $S_\rightarrow$, if exists, has no i: Assume it is not true. Since there is no $i+1$ in $S_\downarrow$ if it exists, there is no $i+1$ below $S_\rightarrow$. We know $S_\rightarrow$ corresponds to ")$-$(" or ")". In either case, the ")" is unpaired. It must be part of a right form or a combined form. However, there is no combined form or right form after the ")" that corresponds to S. Contradiction.

□

We can also check f_i' and e_i satisfy the following property.

Lemma 46. *Take $T_1, T_2 \in \mathsf{SVT}_\lambda$ and $i \in [n-1]$. Then $f_1'(T_1) = T_2$ if and only if $e_2'(T_2) = T_1$.*

Proof. Let w_1 (resp. w_2) be the i-word of T_1 (resp. T_2). Assume $f_1'(T_1) = T_2$, so T_1 and T_2 differ at exactly one entry, say S. Consider what f_i' does. First, assume f_i' removes i from S. Then S in T_1 corresponds to the ")$-$(" at the beginning of the combined form in w_1. The word w_2 is obtained from w_1 by turning this ")$-$(" into "(". The combined form in w_1 becomes a left form in w_2. Moreover, it is the first left form in w_2. The "(" at the beginning of this left form corresponds to S in T_2. If we apply e_i' on T_2, it will add i to S and yield T_1.

Now assume f_i' puts $i+1$ into S. This entry corresponds to the ")" at the end of the the last right form in w_1. The word w_2 is obtained from w_1 by turning this ")" into ")$-$(". This right form in w_1 becomes a combined form in w_2. The ")$-$(" at the end of this combined form corresponds to S in T_2. If we apply e_i' on T_2, it will remove $i+1$ in S and yield T_1.

Consequently, we know $f_i'(T_1) = T_2$ implies $e_i'(T_2) = T_1$. The other direction can be proved similarly. □

Finally, we define f_i and e_i on SVT_λ.

Definition 47. For $T \in \mathsf{SVT}_\lambda$, $f_i(T) := f_i'(f_i'(T))$ and $e_i(T) := e_i'(e_i'(T))$.

We can make sure f_i and e_i changes ε_i, φ_i and wt correctly.

Lemma 48. *Take $T \in \mathsf{SVT}_\lambda$ and $i \in [n-1]$. Assume $f_i'(T), f_i(f_i'(T)) \in \mathsf{SVT}_\lambda$. Let $v_1, \ldots, v_n$ be the standard basis of $\mathbb{Z}^n$. Then*

$$\varepsilon_i(f_i(T)) = \varepsilon_i(T) + 1,$$

$$\varphi_i(f_i(T)) = \varphi_i(T) - 1,$$

$$\mathsf{wt}(f_i(T)) = \mathsf{wt}(T) - v_i + v_{i+1}.$$

Proof. Consider what f_i' does on T. If T has a combined form in its i-word, then f_i' removes an i from T. The combined form in the i-word of T becomes a left form. Thus,

$$\varepsilon(f_i'(T)) = \varepsilon(T) + 1, \quad \varphi(f_i'(T)) = \varphi(T), \quad \mathsf{wt}(f_i'(T)) = \mathsf{wt}(T) - v_i.$$

Otherwise, if T has no combined form in its i-word, then f_i' adds an $i+1$ to T. A right form in the i-word of T becomes a combined form. Thus,

$$\varepsilon(f_i'(T)) = \varepsilon(T), \quad \varphi(f_i'(T)) = \varphi(T) - 1, \quad \mathsf{wt}(f_i'(T)) = \mathsf{wt}(T) + v_{i+1}.$$

Now consider T and $f_i'(T)$. Exactly one of these two has a combined form in its i-word. We know $\mathsf{wt}(f_i(T))$ is either $\mathsf{wt}(f_i'(T)) - v_i = \mathsf{wt}(T) + v_{i+1} - v_i$ or $\mathsf{wt}(f_i'(T)) + v_{i+1} = \mathsf{wt}(T) - v_i + v_{i+1}$. Our claim of $\varphi_i(f_i(T))$ and $\varepsilon_i(f_i(T))$ can be checked similarly.

□

Now we can establish the main result of this subsection.

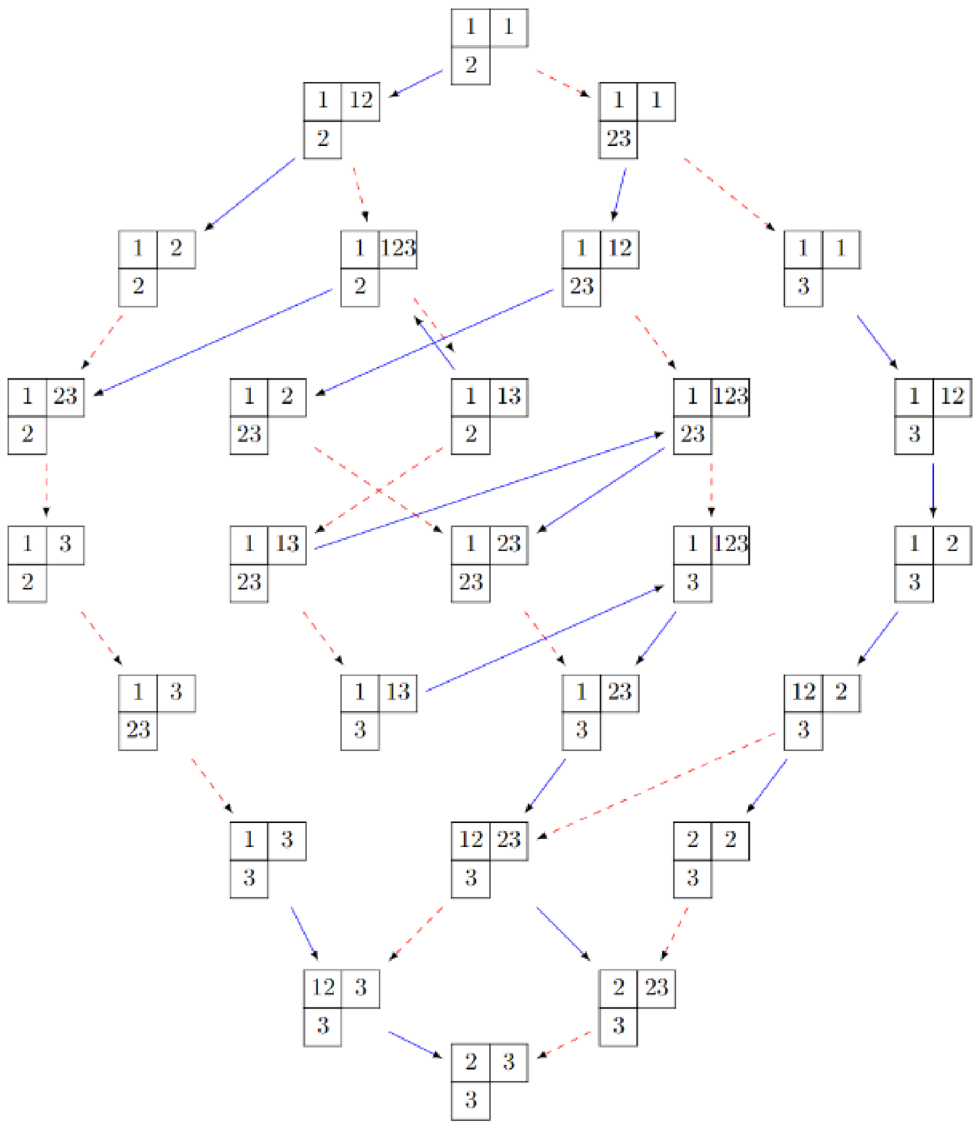
Theorem 49. *The set* SVT_λ*, together with maps* $f_i, e_i, \varepsilon_i, \varphi_i$ *and* wt*, is a seminormal abstract Kashiwara* GL_n*-crystal.*

Proof. We have established the two axioms in Definition 18: Axiom (K1) follows from Lemma 46 and Lemma 48; Axiom (K2) is checked in Lemma 42.

Next, we check it is seminormal. Take $T \in \mathsf{SVT}_\lambda$. Each time we apply e_i, the i-word of T would lose one left form. Thus, $e_1^{\varepsilon_i(T)}(T)$ has no left form. We have $\varepsilon_i(T) = \max\{k : e_i^k(T) \neq \mathbf{0}\}$. The other equality can be proved similarly. □

Example 50. In the picture on the next page, we depict the 27 elements of $\mathcal{B}_3$ with shape $(2,1)$. A blue solid arrow represents the f_1' operator and a red dashed arrow represents the f_2' operator. Thus, by following two consecutive blue solid arrows, one can move from T to $f_1(T)$. By following two consecutive red dashed arrows, one can move from T to $f_2(T)$.

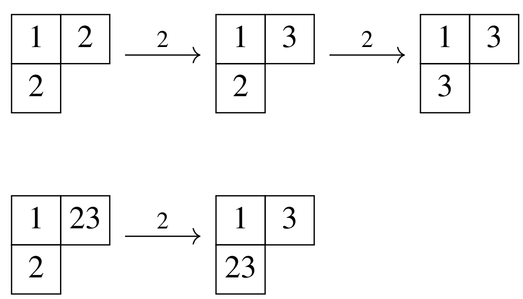
One can see how our crystal operators differ from the crystal operators defined in [MPS20] by comparing the picture next page with Figure 1 in [MPS20]. For example, consider the first SVT in the fourth row of picture next page. The f_2 in our definition sends it to the first SVT in the sixth row. The f_2 in [MPS20] would send it to the third SVT in the sixth row.



4.3.2 Double i-strings

In this subsection, we introduce and investigate double i-strings, which can be viewed as analogues of i-strings. Based on the definition in Chapter 2, an i-string in SVT_λ is a sequence of SVT $T_0,\ldots,T_k$ such that $e_i(T_0) = f_i(T_k) = \mathbf{0}$ and $f_i(T_j) = T_{j+1}$ for $j = 0,1,\ldots,k-1$.

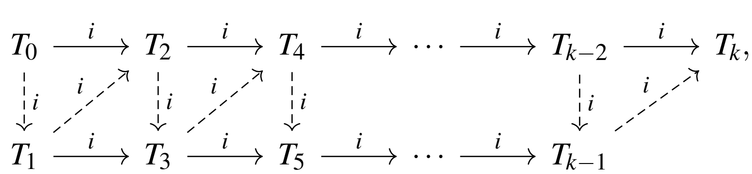
Example 51. The following are 2-strings in $\mathcal{B}_3$.



Now we generalize this notion and define a double i-string. We simply replace e_i and f_i in the definition of an i-string by e'_i and f'_i.

Definition 52. Take $i \in [n-1]$. A *double i-string* is a sequence $T_0,\ldots,T_k \in \mathsf{SVT}_\lambda$ such that $e'_i(T_0) = f'_i(T_k) = \mathbf{0}$ and $f'_i(T_j) = T_{j+1}$ for each $j \in \{0,1,\ldots,k-1\}$.

We say T_0 is the *source* of its double i-string. Diagrammatically, we can represent the double i-string as:



where solid arrow represents f_i and dash arrow represents f'_i.

Remark 53. A double i-string can be viewed as a refinement of the "i-K-string" in [MPS20]. If we remove all dash arrows except the one from T_0 to T_1, we get an i-K-string.

We make some basic observations about a double i-string.

Lemma 54. *Let $T_0, \dots, T_k$ be a double i-string. Then we have the following.*

1. *The number k is even.*

2. *If $k \geq 2$, then this double i-string consists of two i-strings: $T_0, T_2, \dots, T_k$ and $T_1, T_3, \dots, T_{k-1}$. An element in the former i-string has no combined form in its i-word. An element in the latter i-string has a combined form in its i-word.*

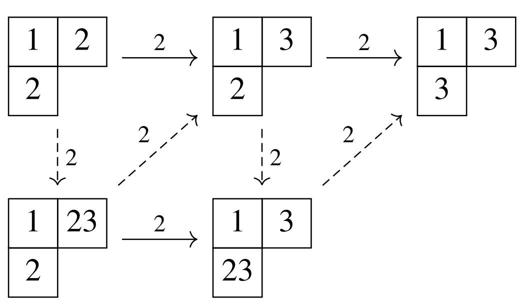
3. $\mathsf{wt}(T_{2j+1}) = \mathsf{wt}(T_{2j}) + v_{i+1}$.

4. $\mathsf{wt}(T_{2j}) = \mathsf{wt}(T_{2j-1}) - v_i$.

Proof. We know T_0 and T_k have no combined forms in their i-word. By the definition f_i', the i-word of T_{j+1} has a combined form if and only if the i-word of T_j has no combined form. This implies (1). (2) is immediate. (3) and (4) follow from the proof of Lemma 48.

□

Example 55. The following is a double 2-string in $\mathcal{B}_3$.



This double 2-string consists of two 2-strings that appear in the previous example. Observe that the three SVT in the first row do not have combined forms in their 2-words, while the two SVT on the second row have.

4.3.3 Double i-string and the right key

This subsection investigates how the right key is changed in a double i-string. More explicitly, we prove:

Lemma 56. *Let $T_0, T_1, \ldots, T_{2k}$ be a double i-string in SVT_λ. Assume $K_+(T_0) = \mathsf{key}(\alpha)$. Then $\alpha_i \geq \alpha_{i+1}$, and there are two possibilities:*

- $K_+(T_1) = \cdots = K_+(T_k) = \mathsf{key}(\alpha)$*, or*
- $K_+(T_1) = \cdots = K_+(T_k) = \mathsf{key}(s_i\alpha)$*.*

Example 57. Let $T_0, \ldots, T_4$ be the double 2-string of $\mathcal{B}_3$ in Example 55. We have $K_+(T_0) = \mathsf{key}(\alpha)$ and $K_+(T_1) = \cdots = K_+(T_4) = \mathsf{key}(s_i\alpha)$, where $\alpha = (1,2,0)$

To prove Lemma 56, we study how f_i' and e_i' change $K_+(T)$, where $T \in \mathsf{SVT}_\lambda$ has a combined form in its i-word. We start with a few basic properties about the $\star$ operator.

Lemma 58. *Let S be a finite subset of $\mathbb{Z}$. Pick $i \in \mathbb{Z}$ and assume w is a word of $\mathbb{Z}$ with no $i+1$. Then if $S \star iw$ contains $i+1$, it must also contain i.*

Proof. If $i+1 \notin S \star i$, then $i+1 \notin S \star iw$ since w has no $i+1$. We are done in this case. Otherwise, $i, i+1 \in S \star i$. When w acts on $S \star i$, to change the i, it first needs to bump the $i+1$. Thus, i remains in $S \star iw$ if it contains $i+1$. $\square$

Definition 59. Let S a finite subset of $\mathbb{Z}$. We define the set $\partial_i S$ according to the following cases

- If $i, i+1 \in S$, then $\partial_i S = S$.
- If $i \notin S$ and $i+1 \notin S_1$, then $\partial_i S = S$.
- If $i \in S$ and $i+1 \notin S_1$, then $\partial_i S = S - \{i\} \sqcup \{i+1\}$.
- If $i \notin S$ and $i+1 \in S_1$, then $\partial_i S$ is undefined.

Lemma 60. *Let S_1 be a set such that $\partial_i(S_1)$ is defined. Let $S_2 = \partial_i S_1$. Then we have the following.*

- *For any $x \neq i$ or $i+1$, the set $S_2 \star x$ is $S_1 \star x$ or $\partial_i(S_1 \star x)$;*
- $S_2 \star (i+1) = S_1 \star (i+1)$.

Proof. If $S_1 = S_2$, then clearly $S_2 \star x = S_1 \star x$ and $S_2 \star (i+1) = S_1 \star (i+1)$. Now assume $S_1 \neq S_2$ (i.e. $i \in S$ and $i+1 \notin S$). We know S_2 is obtained by changing the i in S_1 into $i+1$. We check the two statements.

- If x bumps some $y \neq i$ in S_1 or adds itself to S_1, then x would do the same in S_2, so $S_2 \star x = \partial_i(S_1 \star x)$. Now if x bumps i in S_1, then it would bump $i+1$ in S_2, so $S_2 \star x = S_1 \star x$.
- The $i+1$ must bump i in S_1 and $i+1$ in S_2, so $S_2 \star (i+1) = S_1 \star (i+1)$.

□

With these basic tools, we can investigate how f_i' and e_i' affects the right key.

Lemma 61. *For $T \in B_n$ and $i \in [i-1]$, $K_+(f'_i(T)) = K_+(T)$ if T has a combined form in its i-word.*

Proof. Assume f'_i removes i from the entry S, which is in column c of T. Then clearly $K_+(T)$ and $K_+(f'_i(T))$ must agree on column j if $j > c$. We only need to worry about column j of $K_+(T)$ and $K_+(f'_i(T))$ for $j \leq c$. Let $T_{\geq j}$ be the SVT obtained by removing the first $j-1$ columns of T. Let $u = word(T_{\geq j})$. Recall that column j of $K_+(T)$ is $\emptyset \star u$.

We may write u as $u_1 \ i \ i+1 \ u_2$, where the i and $i+1$ correspond to the i and $i+1$ in S. Then column j of $K_+(f'_i(T))$ is $\emptyset \star u_1 \ (i+1) \ u_2$. Thus, it remains to prove:

$$(\emptyset \star u_1) \star \ i \ (i+1) = (\emptyset \star u_1) \star \ (i+1) \tag{4.3}$$

Consider the i-word of T. The combined form must follow a right form or a null-form or nothing. Thus, the character before the combined form must be ")" or nothing. In other words, u_1 has two possibilities: has neither i nor $i+1$, or has the form $u_1^1 \ i \ u_1^2$, where u_1^2 has no $i+1$. By Lemma 58, we have either $i+1 \notin \emptyset \star u_1$ or $i, i+1 \in \emptyset \star u_1$. Now we study these two cases.

1. Assume we have the former case. If we let i act on $\emptyset \star u_1$, it will change a number into i, or add itself to it. Then if we let $i+1$ act on the result, it will replace the i by $i+1$, which is the same as $(\emptyset \star u_1) \star (i+1)$.

2. Assume we have the latter case. Action of i or $i+1$ on $\emptyset \star u_1$ will not do anything. Both sides of (4.3) must agree with $\emptyset \star u_1$.

□

Similarly, for e'_i, we have:

Lemma 62. *Take $T \in B_n$ and $i \in [i-1]$. Assume T has a combined form in its i-string. Assume $K_+(T) = \mathsf{key}(\alpha)$. If T also has a left form, then $K_+(e'_i(T)) = \mathsf{key}(\alpha)$. If T has no left form, then $K_+(e'_i(T)) = \mathsf{key}(\alpha)$ or $\mathsf{key}(s_i\alpha)$.*

Proof. Assume e'_i removes $i+1$ from the entry S, which is in column c of T. Then clearly column j of $K_+(T)$ and $K_+(e'_i(T))$ must agree if $j > c$. We only need to worry about column j of $K_+(T)$ and $K_+(f'_i(T))$ for $j \leq c$. Let $T_{\geq j}$ be the SVT obtained by removing the first $j-1$ columns of T. Let $u = word(T_{\geq j})$. Recall that column j of $K_+(T)$ is $\emptyset \star u$.

We may break u into $u_1 \ i \ (i+1) \ u_2$, where the i and $i+1$ correspond to the i and $i+1$ in S. Then column j of $K_+(e'_i(T))$ is $\emptyset \star u_1 \ i \ u_2$. Thus, it remains to compare:

$$(\emptyset \star u_1) \star \ i \ (i+1) \ u_2 \text{ and } (\emptyset \star u_1) \star \ i \ u_2.$$

Let $S_1 = (\emptyset \star u_1) \star i$ and $S_2 = (\emptyset \star u_1) \star i\,(i+1)$. Clearly, $i \in S_1$. If $i+1 \in S_1$, then $i, i+1 \in S_1$ and $S_1 = S_2$. If $i+1 \notin S_1$, then $S_2 = S_1 - \{i\} \sqcup \{i+1\}$. In either case, we have $S_2 = \partial_i S_1$.

Now we think about the i-word of T. The combined form must be followed by a left form or a null-form or nothing. Thus, the character after the combined form must be "(" or nothing. In other words, we have two cases.

- Case 1: The word u_2 can be written as $u_2^1 \ (i+1) \ u_2^2$. By Lemma 60, $S_2 \star u_2^1 = \partial_i(S_1 \star u_2^1)$. Then $S_2 \star u_2^1 \ (i+1) = S_1 \star u_2^1 \ (i+1)$, so $S_2 \star u_2 = S_1 \star u_2$.

- Case 2: The word u_2 has no i or $i+1$. By Lemma 60, $S_2 \star u_2 = \partial_i(S_1 \star u_2)$.

The second case is possible only when the i-word of T has no left form. This is exactly what we need to prove. □

Now we are ready to prove Lemma 56.

Proof of Lemma 56. First, we consider T_0. Since it has neither combined form nor left form, its last character in the i-string, if exists, must be ")". Thus, columns of $K_+(T_0)$ will be $\emptyset \star u_1\, i\, u_2$ or $\emptyset \star u$, where u_2 and u have no i or $i+1$. By Lemma 58, if a column of $K_+(T_0)$ has $i+1$, it must also have i. Thus, $\alpha_i \geq \alpha_{i+1}$.

Now by Lemma 61, we know $K_+(T_{2j-1}) = K_+(T_{2j})$ where $j \in [k]$. By Lemma 62 we know $K_+(T_{2j}) = K_+(T_{2j+1})$ where $j \in [k]$. Thus, $T_1, \ldots, T_{2k}$ all have the same right key.

Finally, notice that T_1 is the source of its i-string, so it has no left form. By Lemma 62 again, $K_+(T_1) = \mathsf{key}(\alpha)$ or $\mathsf{key}(s_i\alpha)$, where $\alpha = K_+(T_0)$. □

Corollary 63. *Let T be a SVT. If $f_i(T) \neq \mathbf{0}$ and $K_+(T) \neq K_+(f_i(T))$, then T must be the source of its double i-string.*

4.3.4 Proof of Theorem 12

In this subsection, we derive a few lemmas and then use them to prove Theorem 12. First, we describe a well-known result that is implicit in [Kas93]. It states that the generating function of each i-string behaves nicely under π_i. For the sake of completeness, we provide a brief proof.

Lemma 64. *For each i-string $T_0, \ldots, T_k$, we have*

$$\pi_i(x^{\mathsf{wt}(T_0)}) = \sum_{j=0}^{k} x^{\mathsf{wt}(T_j)}.$$

Proof. Write $x^{\mathsf{wt}(T_0)}$ as $mx_i^a x_{i+1}^b$, where m is a monomial with no x_i or x_{i+1}. By Lemma 20, $x^{\mathsf{wt}(T_k)} = mx_i^b x_{i+1}^a$. Thus, $k = b - a$. Finally, we have

$$\begin{aligned}\pi_i(x^{\mathsf{wt}(T_0)}) &= m\pi_i(x_i^a x_{i+1}^b) \\ &= m\sum_{j=0}^{b-a} x_i^{a-j} x_{i+1}^{b+j} \\ &= \sum_{j=0}^{k} x^{\mathsf{wt}(T_j)}.\end{aligned}$$

□

As mentioned earlier, double i-string can be viewed as a refinement of i-K-string in [MPS20]. Authors of [MPS20] knew that the generating function of an i-K-string behaves nicely under $\pi_i^{(\beta)}$: Applying $\pi_i^{(\beta)}$ on the weight of the source yields the generating function of a whole i-K-string. This property is also satisfied by double i-strings. The following is implicit in [MPS20, Theorem 7.5].

Lemma 65. *For each double i-string $T_0, \ldots, T_{2k}$, we have*

$$\pi_i^{(\beta)}(x^{\mathsf{wt}(T_0)}\beta^{\mathsf{ex}(T_0)}) = \sum_{j=0}^{2k} x^{\mathsf{wt}(T_j)}\beta^{\mathsf{ex}(T_j)},$$

$$\pi_i^{(\beta)}(\sum_{j=0}^{2k} x^{\mathsf{wt}(T_j)}\beta^{\mathsf{ex}(T_j)}) = \sum_{j=0}^{2k} x^{\mathsf{wt}(T_j)}\beta^{\mathsf{ex}(T_j)}.$$

Proof. First, we establish the first equation using the argument in [MPS20]. Notice that

$$\pi_i^{(\beta)}(f) = \pi_i(f + \beta x_{i+1} f).$$

Thus, its left hand side becomes

$$\pi_i(x^{\mathsf{wt}(T_0)}\beta^{\mathsf{ex}(T_0)} + \beta x_{i+1} x^{\mathsf{wt}(T_0)}\beta^{\mathsf{ex}(T_0)}).$$

Notice that $x^{\mathsf{wt}}(T_1) = x^{\mathsf{wt}}(T_0)x_{i+1}$ and $\mathsf{ex}(T_1) = \mathsf{ex}(T_0) + 1$. We can further simplify the left hand side into

$$\begin{aligned}&\pi_i(x^{\mathsf{wt}(T_0)}\beta^{\mathsf{ex}(T_0)} + x^{\mathsf{wt}(T_1)}\beta^{\mathsf{ex}(T_1)})\\ =&\pi_i(x^{\mathsf{wt}(T_0)}\beta^{\mathsf{ex}(T_0)}) + \pi_i(x^{\mathsf{wt}(T_1)}\beta^{\mathsf{ex}(T_1)}).\end{aligned}$$

Then the first equation is established by Lemma 64.

For the second equation, notice that $\sum_{j=0}^{2k} x^{\mathsf{wt}(T_j)}\beta^{\mathsf{ex}(T_j)}$ is symmetric in x_i and x_{i+1}. Then the equation is established by the fact: $\pi_i^{(\beta)}(f) = f$ if $s_i(f) = f$.

□

Next, we describe $\mathsf{SVT}(\alpha)$ in terms of double i-strings.

Lemma 66. *Take any weak composition α. For each double i-string $T_0, \ldots, T_{2k}$, if $T_i \in \mathsf{SVT}(\alpha)$ with $i > 0$, then $T_0, \ldots, T_{2k} \in \mathsf{SVT}(\alpha)$.*

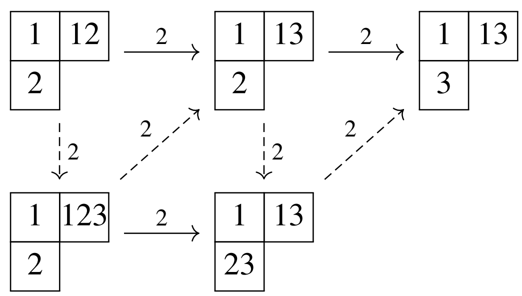
Proof. We know $K_+(T_i) \leq \mathsf{key}(\alpha)$. Since $T_1, \ldots, T_{2k}$ all have the same right key, they are all in $\mathsf{SVT}(\alpha)$. By Lemma 56, $K_+(T_0) \leq K_+(T_i)$, so $T_0 \in \mathsf{SVT}(\alpha)$. □

The following is analogous to [Kas93, Proposition 3.3.5].

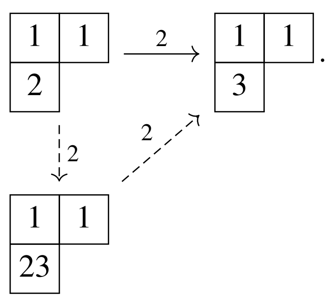
Corollary 67. *For a weak composition α and a double i-string $S = \{T_0, \ldots, T_{2k}\}$, we know $\mathsf{SVT}(\alpha) \cap S$ is S, $\emptyset$, or $\{T_0\}$.*

Lemma 68. *Let α be a weak composition such that $\alpha_i > \alpha_{i+1}$. We can decompose* $\mathsf{SVT}(s_i\alpha)$ *into a disjoint union of double i-strings. For each of the double i-string in* $\mathsf{SVT}(s_i\alpha)$*,* $\mathsf{SVT}(\alpha)$ *either contains its source or all of it.*

Example 69. When $\alpha = (1,2,0)$, the set $\mathsf{SVT}(s_2\alpha)$ is a disjoint union of three double 2-strings. Besides the double 2-string in Example 55, it also contains



and



For each of these three double 2-strings, the set $\mathsf{SVT}(\alpha)$ only contains the source.

Proof. Let $T_0, \ldots, T_{2k}$ be a double i-string that intersects with $\mathsf{SVT}(s_i\alpha)$. Corollary 67 implies $T_0 \in \mathsf{SVT}(s_i\alpha)$. Let $\gamma = wt(K_+(T_0))$, then $\mathsf{key}(\gamma) \leq \mathsf{key}(s_i\alpha)$. We know each SVT in this double i-string has right key $\mathsf{key}(\gamma)$ or $\mathsf{key}(s_i(\gamma))$. Since $\alpha_i > \alpha_{i+1}$, $\mathsf{key}(s_i\gamma) \leq \mathsf{key}(s_i\alpha)$. Thus, the whole double i-string is in $\mathsf{SVT}(s_i\alpha)$.

Lemma 56 implies that $\gamma_i \geq \gamma_{i+1}$, so $\mathsf{key}(\gamma) \leq \mathsf{key}(\alpha)$. We have $T_0 \in \mathsf{SVT}(\alpha)$. By

Corollary 67, $\mathsf{SVT}(\alpha)$ either contains T_0 or the whole double i-string. $\square$

Now we are ready to prove our first main result:

Proof of Theorem 12. We only need to check $\sum_{T\in\mathsf{SVT}(\alpha)} x^{\mathsf{wt}(T)}\beta^{\mathsf{ex}(T)}$ satisfies the recursive definition of $\mathfrak{L}_\alpha^{(\beta)}$. In other words, we need to prove:

- If α is a partition, then

$$\sum_{T\in\mathsf{SVT}(\alpha)} x^{\mathsf{wt}(T)}\beta^{\mathsf{ex}(T)} = x^\alpha.$$

- If $\alpha_i > \alpha_{i+1}$, then

$$\pi_i^{(\beta)}\Big(\sum_{T\in\mathsf{SVT}(\alpha)} x^{\mathsf{wt}(T)}\beta^{\mathsf{ex}(T)}\Big) = \sum_{T\in\mathsf{SVT}(s_i\alpha)} x^{\mathsf{wt}(T)}\beta^{\mathsf{ex}(T)} \tag{4.4}$$

The first statement is immediate. For the second one, we break $\mathsf{SVT}(\alpha)$ into $A\sqcup B$. The set A consists of all T whose whole double i-string is in $\mathsf{SVT}(\alpha)$. The set B contains all $T\in\mathsf{SVT}(\alpha)$ such that part of its double i-string is not in $\mathsf{SVT}(\alpha)$. Let $\overline{B}$ be the union of double i-strings who intersect with B. By Lemma 68, elements in B are sources of double i-string and $\mathsf{SVT}(s_i\alpha) = A\sqcup\overline{B}$. Now by Lemma 65,

$$\pi_i^{(\beta)}\Big(\sum_{T\in A} x^{\mathsf{wt}(T)}\beta^{\mathsf{ex}(T)}\Big) = \sum_{T\in A} x^{\mathsf{wt}(T)}\beta^{\mathsf{ex}(T)},$$

$$\pi_i^{(\beta)}\Big(\sum_{T\in B} x^{\mathsf{wt}(T)}\beta^{\mathsf{ex}(T)}\Big) = \sum_{T\in \overline{B}} x^{\mathsf{wt}(T)}\beta^{\mathsf{ex}(T)}.$$

Equation (4.4) is obtained by summing up the two equations above.

$\square$

Chapter 4, in full, is a reprint of the material as it appears in Set-valued tableaux rule for Lascoux polynomials. Tianyi Yu, Combinatorial Theory, 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 5

Proving the RSVT formula

In this section, we prove Theorem 15. The first step is to rewrite it as a rule involving RSSYT, instead of RSVT, using the following notion:

Definition 70. For a RSSYT T, define $\mathsf{WT}(T)$ by

$$\mathsf{WT}(T) = \sum_{T'} \beta^{\mathsf{ex}(T')} x^{\mathsf{wt}(T')},$$

where the sum is over all RSVT T' with $L(T') = T$.

Then Theorem 15 can be rewritten as:

Theorem 71. *For a weak composition* α*, we have*

$$\mathfrak{L}_{\alpha}^{(\beta)} = \sum_{T \in RSSYT(\alpha)} \mathsf{WT}(T)$$

It is clear that Theorem 15 and Theorem 71 are equivalent. Readers may still insist that Theorem 71 involves RSVTs, since they appear in how we defined $\mathsf{WT}(T)$. The following lemma resolves this issue:

Lemma 72. *For any RSSYT of shape* λ*,*

$$\mathsf{WT}(T) = x^{\mathsf{wt}(T)} \prod_{(s,k)} (1+\beta x_k) \tag{5.1}$$

where (s,k) *runs over pairs such that* s *is a box in* λ*,* k *is less than the value of* T *in that box, and replacing the* s*-th entry of* T *by* k *results in a RSSYT.*

Proof. Consider the following way of turning T into a RSVT in $L^{-1}(T)$. Let a be an entry in T. Let b be the entry on its right and $b = 1$ if such an entry does not exist. Let c be the entry below a and $c = 0$ if such an entry does not exist. We turn a into $\{a\}$, and then add some numbers to this set. We may add any k such that $a > k$, $k > c$ and $k \geq b$. Not adding this k will contribute 1 and adding this k will contribute an βx_k. Thus, each such k contributes $(1+\beta x_k)$. Clearly, the choices are independent and any element in $L^{-1}(T)$ can be obtained this way. □

In the rest of this chapter, we show Theorem 71 using only RSSYTs. The idea is to show the sum in Theorem 71 satisfies the defining recursion of Lascoux polynomials. Fix an i throughout the rest of this chapter In section 5.1, we partition all RSSYTs into several families. In section 5.2, we investigate the left keys of RSSYTs in a family. In section 5.3, we derive a few identities regarding the operators π_i and $\pi^{(\beta)}$. Finally, in section 5.4, we study the weight-generating function over RSSYTs in a family and particularly how the operators apply to them. Then we can prove Theorem 71.

5.1 Partitioning RSSYTs

Let T be a RSSYT. We classify its i and $i+1$ into 3 categories: "ignorable", "frozen", and "free". First, we find all pairs of $i+1$ and i that appear in the same column. We pair them and say they are "ignorable". Next, we find non-ignorable i and $i+1$ such that:

1. i is on the left of $i+1$.

2. Any column between them must have an ignorable pairs.

We pair them and say they are "frozen". Other non-ignorable i and $i+1$ are called "free".

Example 73. When $i=3$, consider the following RSSYT:

6	6	6	6	4	4
5	4	3	3		
4	3				

The red entries are ignorable and blue entries are frozen. Other 3 and 4 are free.

Based on this labelling, we may partition RSSYTs into families.

Definition 74. A *family* is an equivalence class under the transitive closure of the following: two RSSYTs are related if they differ by changing a single i into an $i+1$ (or vice versa) where the changed letters are free in both tableaux.

Example 75. Consider the reverse tableau in the previous example. Its family also includes:

6	6	6	6	4	3
5	4	3	3		
3	3				

6	6	6	6	4	3
5	4	3	3		
4	3				

6	6	6	6	4	3
5	4	4	3		
4	3				

6	6	6	6	4	4
5	4	3	3		
3	3				

6	6	6	6	4	4
5	4	4	3		
4	3				

However, the following is in another family:

6	6	6	6	4	3
5	4	4	3		
3	3				

Given an RSSYT, how can we write down all the elements from its family? Clearly, to obtain its family members, we can only change its free entries. We also need to make sure they are still free after our changes. In other words, assume a and b are two free entries. If a is on the left of b and all columns between them have ignorable pairs, then we cannot change a into i and b into $i+1$. This criterion leads to the following definition.

Definition 76. Let T be a RSSYT. We partition its free i and $i+1$ into "blocks". Two entries are in the same block iff all columns between them have ignorable pairs.

Thus, to enumerate the family of a RSSYT T, we just replace entries in each block by a weakly decreasing (from left to right) sequence of i and $i+1$. The reader may check the enumeration of the family in the previous example.

5.2 Families and left keys

This subsection aims to describe the left keys of a family. This idea is formalized in the following lemma:

Lemma 77. *Let $\mathcal{F}$ be a family. Then its elements can have at most 2 different left keys. If they all have the same left key γ, then $\gamma_i \geq \gamma_{i+1}$.*

If they have two different left keys, then they must be γ and $s_i\gamma$, where $\gamma_i > \gamma_{i+1}$. In this case, we also have:

1. *$T \in \mathcal{F}$ has left key γ iff T's leftmost block only has i.*

2. *All columns before the first block must have ignorable pairs.*

Before proving the lemma, we need to introduce an algorithm that computes the left key. The algorithm is introduced in section 5 of [Wil13]. Here we describe this algorithm in a slightly different way.

Definition 78. Given two sets C_1, C_2 of numbers, we define the set $C_1 \triangleleft C_2$ as follows. Assume $C_2 = \{a_1 < a_2 < \cdots < a_m\}$. We find the smallest b_1 in C_1 such that $b_1 \geq a_1$. Then we find the smallest b_2 in C_1 such that $b_2 \geq a_2$ and $b_2 > b_1$. Similarly, we find $b_3, \ldots b_m$. Let $C_1 \triangleleft C_2 = \{b_1 < b_2 < \cdots < b_m\}$.

More generally let $C_1, C_2, \ldots, C_k$ be finite sets of numbers. Observe that the following expression is well-defined when $j \leq k$

$$C_j \triangleleft \cdots \triangleleft C_k := C_j \triangleleft (C_{j+1} \triangleleft \cdots \triangleleft C_k)$$

where the base case is

$$C_k \lhd \cdots \lhd C_k := C_k$$

For a RSSYT T, let C_j be the set of numbers in column j of T. Then column k of $K_-(T)$ consists of $C_1 \lhd \cdots \lhd C_k$ by [Wil13].

To study this algorithm, we need to classify columns of T. We may identify a column of a RSSYT with the set of numbers in it. Each column can be classified as follows:

1. Type 1 column: It has neither i nor $i+1$.

2. Type 2 column: It has i but no $i+1$.

3. Type 3 column: It has $i+1$ but no i.

4. Type 4 column: It has both i and $i+1$.

Now we make several observations.

Lemma 79. *If C_1 has type 4 and C_2 does not have type 3, then $C_1 \lhd C_2$ cannot have type 3.*

Proof. Assume $C_1 \lhd C_2$ has type 3. Then we must pick $i+1$ in C_1 for some m in C_2. Moreover, i in C_1 is never picked. Thus, m must be $i+1$ and C_2 cannot have i. C_2 has type 3, contradiction. □

Lemma 80. *Let T be a RSSYT with no free $i+1$. Assume $\gamma = K_-(T)$. Then $\gamma_i \geq \gamma_{i+1}$.*

Proof. Let $C_1, C_2, \ldots$ be columns of T. Consider column k of $K_-(T)$. We only need to prove it cannot have type 3.

Suppose $C_1, \ldots, C_k$ all have type 4. Then Lemma 79 guarantees $C_1 \lhd \cdots \lhd C_k$ cannot have type 3. Otherwise, we can find $j \leq k$ such that $C_1, \ldots, C_{j-1}$ have type 4 and C_j does not have type 4. Since T has no free $i+1$, C_j must have type 1 or 2. Then $C_j \triangleleft \cdots \triangleleft C_k$ must also have type 1 or 2. By Lemma 79, $C_1, \ldots, C_{j-1}$ cannot turn it into type 3. □

Lemma 81. *Assume C_2 has type 2. We change its i into $i+1$ and obtain C_2'. Assume concatenating C_1 and C_2' yields a RSSYT. Then,*

1. *If C_1 has type 4, then $C_1 \triangleleft C_2 = C_1 \triangleleft C_2'$, or $C_1 \triangleleft C_2'$ is obtained from $C_1 \triangleleft C_2$ by changing an i into $i+1$.*

2. *If C_1 has type 1 or 3, then $C_1 \triangleleft C_2 = C_1 \triangleleft C_2'$.*

Proof. We do a case study based on the type of C_1.

1. Assume C_1 has type 4. When we consider i in C_2, there are 3 possibilities: The i in C_1 is picked; or a number larger than it is picked; or the i is still available.

   In the first 2 cases, clearly this i in C_2 behaves as if it is an $i+1$. Then $C_1 \triangleleft C_2 = C_1 \triangleleft C_2'$. In the last case, i in C_2 picks i, and $i+1$ in C_2' picks $i+1$. Our claim is clear.

2. Assume C_1 has type 1 or 3. Clearly the i in C_2 behaves as if it is an $i+1$, so $C_1 \triangleleft C_2 = C_1 \triangleleft C_2'$.

□

Lemma 82. *Let T be a RSSYT. Assume column j of T has a free i, which is the leftmost free i in its block. We change this i into $i+1$ and get T'. If $\gamma = K_-(T)$, then $K_-(T') = \gamma$ or $s_i\gamma$. Moreover, if the latter case happens, we must have:*

1. *The i we changed is in the leftmost block of T.*

2. *Each of column $1, \ldots, j-1$ of T has ignorable pairs.*

Proof. Let $C_1, C_2, \ldots$ be the columns of T. Let $D_1, D_2, \ldots$ be the columns of T'. Consider column k of $K_-(T)$ and $K_-(T')$. If $k < j$, then clearly they are the same. Now assume $k > j$. Let $C = C_{j+1} \lhd \cdots \lhd C_k$. Because the i in column j is free, we know that $C_{j+1}, \ldots, C_k$ all have type 4, or the leftmost non-type-4 column among them has type 1 or 2. Similar to the proof of Lemma 80, C cannot have type 3. Next, we compare $C_j \lhd C$ and $D_j \lhd C$. If i in C_j is picked by x in C, then this x will pick $i+1$ in D_j. Thus, $D_j \lhd C$ is obtained by changing i in $C_j \lhd C$ into $i+1$. If i in C_j is not picked, the $i+1$ in D_j will not be picked. Then $C_j \lhd C = D_j \lhd C$.

Consequently, if $k \geq j$, $C_j \lhd \cdots \lhd C_k$ agrees with $D_j \lhd \cdots \lhd D_k$, or the latter differs from the former by changing an i into $i+1$. In Lemma 81, we showed this difference might be preserved or corrected by type 4 columns. If $C_1, \ldots, C_{j-1}$ all have type 4, then we know column k of $K_-(T)$ agrees with column k of $K_-(T')$, or the latter differs from the former by changing an i into $i+1$. Otherwise, we let l be the largest such that $l < j$ and C_l does not have type 4. Since the i in column j of T is the leftmost i in its block, C_l must have type 1 or 3. By Lemma 81,

$$C_l \lhd \cdots \lhd C_k = D_l \lhd \cdots \lhd D_k$$

Thus, each column of $K_-(T')$ either agrees with the corresponding column in $K_-(T)$, or differs by changing an i into $i+1$. Since $K_-(T')$ is a key, we have $K_-(T') = \gamma$ or $s_i\gamma$. In the latter case we know $C_1, \ldots, C_{j-1}$ have type 4. Our claims are immediate. $\square$

Now we may prove Lemma 77.

Proof. First pick T from $\mathcal{F}$ that has no free $i+1$. Assume $\gamma = K_-(T)$. By Lemma 80,

$\gamma_i \geq \gamma_{i+1}$.

Then we enumerate other elements in $\mathcal{F}$ by changing free i in T into $i+1$. As long as we do not change the first block, the left key will still be γ. Once we change the first i in the first block, the left key might be fixed, or turned into $s_i\gamma$. The latter case is possible only when all columns before the first blocks have ignorable pairs. After that, no matter which i we change, the left key will be fixed. □

5.3 π_i and $\pi_i^{(\beta)}$

In this subsection, we derive some basic facts about π_i and $\pi_i^{(\beta)}$. Define $X_i = x_i(1+\beta x_{i+1})$ and $X_{i+1} = x_{i+1}(1+\beta x_i)$. Then we have

1. $s_i(X_i) = X_{i+1}$
2. $\pi_i(f) = \partial_i(x_i f)$ and $\pi_i^{(\beta)}(f) = \partial_i(X_i f)$
3. $\partial_i(X_i) = \partial_i(x_i) = 1$.

The following lemma describes how ∂_i acts on a product of several x_i and X_i:

Lemma 83. *Assume we have $u_1, \ldots, u_n$, where each u_j is either x_i or X_i. Then*

$$\partial_i(u_1 \ldots u_n) = \sum_{j=1}^{n} s_i(u_1 \ldots u_{j-1}) u_{j+1} \ldots u_n$$

For instance,

$$\partial_i(x_i X_i x_i X_i) = X_i x_i X_i + x_{i+1} x_i X_i + x_{i+1} X_{i+1} X_i + x_{i+1} X_{i+1} x_{i+1}$$

Proof. Notice:

$$\begin{aligned}\partial_i(u_1\dots u_n) &= \partial_i(u_1)u_2\dots u_n + s_i(u_1)\partial_i(u_2\dots u_n)\\ &= u_2\dots u_n + s_i(u_1)\partial_i(u_2\dots u_n)\end{aligned}$$

Then the proof is finished by induction. □

Corollary 84. *Assume we have $u_1,\dots,u_n$, where each u_j is either x_i or X_i. Then*

$$\pi_i(u_1\dots u_n) = u_1\dots u_n + x_{i+1}\sum_{j=1}^{n} s_i(u_1\dots u_{j-1})u_{j+1}\dots u_n \tag{5.2}$$

$$\pi_i^{(\beta)}(u_1\dots u_n) = u_1\dots u_n + X_{i+1}\sum_{j=1}^{n} s_i(u_1\dots u_{j-1})u_{j+1}\dots u_n \tag{5.3}$$

5.4 $\mathsf{WT}(T)$ and Family

In this subsection, we investigate how $\mathsf{WT}(T)$ works and how it changes within a family. More explicitly, the goal is to understand: $\sum_{T\in\mathcal{F}}\mathsf{WT}(T)$ where $\mathcal{F}$ is a family.

The first step is to understand what governs the power of $(1+\beta x_j)$ in $\mathsf{WT}(T)$. Based on our definition, each row can have at most one entry that contributes $(1+\beta x_j)$ for a fixed j. How is it determined whether a row has such a contributor? The following lemma answers this question. To make it concise, we adopt the following convention throughout the rest of this section: a 0 is appended below each column in a RSSYT.

Lemma 85. *A row has an entry that contributes $(1+\beta x_j)$ iff we can find an entry j' on this row such that:*

1. $j' > j$

2. The entry below j' *is less than* j*.*

Proof. Assume an entry m contributes $1+\beta x_j$. Then clearly $m > j$ and the entry below m is less than j. The row of m clearly satisfies the requirement.

Conversely, assume a row has j' that satisfies the two requirements. Moreover, we pick the rightmost j' among all such j' on this row. Then the entry to the right of j' either does not exist or is at most j. Changing this j' to j will make T a valid anti-SSYT. Thus, this entry contributes $(1+\beta x_j)$. □

With this lemma, we may ascribe contributions of $(1+\beta x_j)$ to rows, instead of entries. However, we would like to ascribe contributions of $(1+\beta x_i)$ and $(1+\beta x_{i+1})$ to specific entries, but the rule is different from our previous criterion. If a row contributes $(1+\beta x_i)$, then we may find the leftmost entry on this row satisfying:

1. It is larger than i.

2. The entry below it is less than i.

We say this entry contributes an $(1+\beta x_i)$. Similarly, if a row contributes $(1+\beta x_{i+1})$, then we may find the rightmost entry on the row below satisfying:

1. It is less than $i+1$.

2. The entry above it is larger than $i+1$.

We say this entry contributes an $(1+\beta x_{i+1})$. To illustrate our new “contribution system”, consider the following example:

Example 86.

6	6	6	6	4	4
5	4	4	3	0	0
4	3	0	0		
0	0				

When $i = 3$, each blue 4 contributes $x_4(1+\beta x_3)$. The red 3 contributes $x_3(1+\beta x_4)$.

Now we fix an arbitrary family $\mathcal{F}$ throughout this subsection. Take any $T \in \mathcal{F}$. Let m be the number of blocks in T. Then we may break $\mathsf{WT}(T)$ into a product:

$$\mathsf{WT}(T) = g^T f_1^T \dots f_m^T$$

Here, f_j^T is the contribution of the j^{th} block in T from left to right. g^T contains the contribution of x_i, x_{i+1}, $(1+\beta x_i)$ and $(1+\beta x_{i+1})$ from all other entries. It also contains powers of x_j and $(1+\beta x_j)$ with $j \neq i$ or $i+1$. Next, we analyze these polynomials. Let us start with g^T:

Lemma 87. *g^T is invariant within the family. Moreover, $s_i g^T = g^T$.*

Proof. Clearly, changing free entries will not affect powers of x_j and $(1+\beta x_j)$ with $j \neq i$ or $i+1$. Let us focus on powers of x_i, x_{i+1}, $(1+\beta x_i)$ and $(1+\beta x_{i+1})$. Each ignorable pair contributes $x_i x_{i+1}$. Now, consider a frozen i. The column on its right must have an ignorable pair or a frozen $i+1$. In either case, if we look at the entry the entry above it and the entry on its top right:

a	b
i	

We must have $a > i+1 \geq b$. Thus, a frozen i always contributes $x_i(1+\beta x_{i+1}) = X_i$.

Similarly, a frozen $i+1$ always contributes X_{i+1}. Thus, each frozen pair contributes X_iX_{i+1}.

Now, we still need to look at contributions of $(1+\beta x_i)$ and $(1+\beta x_{i+1})$ by entries that are not i or $i+1$. Assume j is an entry that contributes $(1+\beta x_{i+1})$ and j is not i or $i+1$. Let j' be the entry above j. Then $j<i$ and $j'>i+1$. There is a k' on the row of j' such that k' contributes $(1+\beta x_i)$. Also, k' is weakly left of j'. The diagram looks like:

k'	$\dots$	j'
k	$\dots$	j

with $k'>i+1$ and $k<i$. We pair this j with k'. Similarly, given such k', we can find its corresponding j. In other words, we pair $(1+\beta x_i)$ contributors with $(1+\beta x_{i+1})$ contributors that are not i or $i+1$. This pairing is clearly invariant under changing free entries. □

Due to this result, we may change our notation g^T into $g^{\mathcal{F}}$, since it only depends on $\mathcal{F}$. The next step is to study each f_j^T. Clearly, a free i contributes either x_i or X_i. How can we determine its contribution? Consider the following lemma:

Lemma 88. *Choose a free i in T. If it is not the last entry in its block, then it contributes x_i iff it is contiguous to the next free i. If it is the last entry in its block, then it contributes x_i iff one of the following happens:*

1. *It is in the highest row.*
2. *There is a b on its top right:* $\begin{array}{cc} & \boxed{b} \\ \boxed{i} & \end{array}$ *with $b>i+1$.*

Proof. First, assume i is not the last entry in its block. We study the entry on its right:

1. The column on its right has ignorable pair. Then we look at $\begin{array}{|c|c|}\hline a & b \\ \hline i & \\ \cline{1-1}\end{array}$ where our chosen i is red. We must have $a>i+1\geq b$. This i contributes X_i.

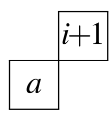
2. The column on its right has a free i and this free i is in the same row. Then we have $\begin{array}{|c|c|}\hline & a \\ \hline i & i \\ \hline\end{array}$ with $a > i+1$, or our chosen i is in the top row. In either case, it contributes x_i.

3. The column on its right has a free i and this free i is not on the same row as our chosen i. Then we have $\begin{array}{|c|c|}\hline a & b \\ \hline i & \\ \hline\end{array}$ with $a > i+1$ and $b \leq i$. Our chosen i contributes X_i.

Now assume i is the last entry in its block. If it is in the top row, then it clearly contributes x_i. Otherwise, we look at: $\begin{array}{|c|c|}\hline a & b \\ \hline i & \\ \hline\end{array}$ We know $a > i+1$. If b exists and $b > i+1$, then clearly our i contributes x_i. Otherwise, our i contributes X_i. $\square$

Similarly, for $i+1$, we have:

Lemma 89. *Choose a free $i+1$ in T. If it is not the first entry in its block, then it contributes x_{i+1} iff it is contiguous to the previous free $i+1$. If it is the first entry in its block, then it contributes x_{i+1} iff there is an a on its lower left with $a < i$.*



We omit the proof since it is basically the same as the previous one.

Now we understand how the free entries contribute. Clearly, the contribution of one block is independent from other blocks. This implication allows us to simplify $\sum_{T \in \mathcal{F}} \mathsf{WT}(T)$. In this family $\mathcal{F}$, there are $a_j + 1$ ways to fill the block j, where a_j is the number of entries in block j. Let f_j^l be the contribution of this block when the number of

$(i+1)$'s is l. l ranges between 0 and a_j. Then we have the following:

$$\sum_{T\in\mathcal{F}} \mathsf{WT}(T) = g^{\mathcal{F}} \prod_{j=1}^{m} \left(\sum_{l=0}^{a_j} f_j^l \right)$$

Then we have

Lemma 90.

$$\sum_{l=0}^{a_j} f_j^l = \pi_i(f_j^0) \text{ or } \pi_i^{(\beta)}(f_j^0)$$

Moreover, take any $T \in \mathcal{F}$ such that its j^{th} block has an $i+1$. Then we are in the second case iff the first $i+1$ in the j^{th} block of T contributes X_{i+1}.

Proof. First, assume block j only has i. Let u_p be the contribution of the p^{th} free entry. Then $f_j^0 = u_1 \dots u_{a_j}$ and each $u_p = X_i$ or x_i.

We change the first free i into $i+1$. By Lemma 88, this change only affects the first entry's contribution. Then $f_j^1 = vu_2 \dots u_{a_j}$ with $v = x_{i+1}$ or X_{i+1}. If $a_j = 1$, we are done by Corollary 84. Otherwise, we change the second free i into $i+1$. The second $i+1$ contributes x_{i+1} iff it is contiguous to the first free entry. Also, $u_1 = x_i$ iff the first entry is contiguous to the second entry. Thus, we know the second entry contributes $s_i u_1$. $f_j^2 = vs_i(u_1)u_3 \dots u_{a_j}$. Continuing this argument, we have $f_j^l = vs_i(u_1 \dots u_{l-1})u_{l+1} \dots u_{a_j}$. The proof is finished by invoking Corollary 84. □

By this result, $\sum_{l=0}^{a_j} f_j^l$ must be symmetric in i and $i+1$. Recall that we have shown $g^{\mathcal{F}}$ is symmetric in i and $i+1$. Thus, $\sum_{T\in\mathcal{F}} \mathsf{WT}(T)$ is symmetric in i and $i+1$. Finally, we have enough results to prove Theorem 71.

Proof. Let α be a weak composition with $\alpha_i > \alpha_{i+1}$. Let $A := \{T \in \mathcal{F} : K_-(T) \leq \alpha\}$ and $B := \{T \in \mathcal{F} : K_-(T) \leq s_i\alpha\}$.

We only need to show

$$\pi_i^{(\beta)} \left(\sum_{T \in A} \mathsf{WT}(T) \right) = \sum_{T \in B} \mathsf{WT}(T) \tag{5.4}$$

This is clearly true when $B = \emptyset$. Now assume $B \neq \emptyset$. If $A = B$, then $A = B = \mathcal{F}$, (5.4) is true since $\sum_{T \in \mathcal{F}} \mathsf{WT}(T)$ is symmetric in x_i and x_{i+1}.

Finally, assume A is a proper subset of B. We can find γ with $\gamma_i > \gamma_{i+1}$ such that elements in A has left key γ and elements in B has left key $s_i\gamma$. Then $s_i\gamma \leq s_i\alpha$ and $\gamma \leq \alpha$. By Lemma 77, A has elements whose first block only has i. We have:

$$\sum_{T \in A} \mathsf{WT}(T) = \left(g^{\mathcal{F}} \prod_{j=2}^{m} \left(\sum_{l=0}^{a_j} f_j^l \right) \right) f_1^0$$

Take $T \in B$. Consider its $i+1$ in the first block. There are two possibilities: It is in the first column, or the column on its left has an ignorable pair. In either case, this $i+1$ contributes X_{i+1}, so

$$\pi_i^{(\beta)}(f_1^0) = \sum_{l=0}^{a_1} f_1^l$$

Finally, letting $f = g^{\mathcal{F}} \prod_{j=2}^{m}(\sum_{l=0}^{a_j} f_j^l)$, we have

$$\begin{aligned}
\pi_i^{(\beta)}\left(\sum_{T\in A}\mathsf{WT}(T)\right) =& \pi_i^{(\beta)}(f f_1^0) \\
=& f\,\pi_i^{(\beta)}(f_1^0) \\
=& f\sum_{l=0}^{a_1} f_1^l \\
=& \sum_{T\in B}\mathsf{WT}(T)
\end{aligned}$$

□

Chapter 5, in full, is a reprint of the material as it appears in Grothendieck-to-Lascoux expansions. Mark Shimozono and Tianyi Yu, Transactions of the American Mathematical Society, 2023. The dissertation author was the primary investigator and author of this paper.

Bibliography

[BKS+08] Anders Skovsted Buch, Andrew Kresch, Mark Shimozono, Harry Tamvakis, and Alexander Yong. Stable Grothendieck polynomials and K-theoretic factor sequences. *Mathematische Annalen*, 340(2):359–382, 2008.

[BS17] Daniel Bump and Anne Schilling. *Crystal bases: representations and combinatorics*. World Scientific Publishing Company, 2017.

[BSW20] Valentin Buciumas, Travis Scrimshaw, and Katherine Weber. Colored five-vertex models and Lascoux polynomials and atoms. *Journal of the London Mathematical Society*, 102(3):1047–1066, 2020.

[Buc02] Anders Skovsted Buch. A Littlewood-Richardson rule for the K-theory of Grassmannians. *Acta Mathematica*, pages 37–78, 2002.

[Dem74] Michel Demazure. Une nouvelle formule des caracteres. *Bull. Sci. Math*, 2(98):163–172, 1974.

[Ful97] William Fulton. *Young tableaux: with applications to representation theory and geometry*. Number 35. Cambridge University Press, 1997.

[Kas90] Masaki Kashiwara. Crystalizing the q-analogue of universal enveloping algebras. *Communications in Mathematical Physics*, 133(2):249–260, 1990.

[Kas91] Masaki Kashiwara. On crystal bases of the q-analogue of universal enveloping algebras. *Duke Mathematical Journal*, 63(2):465–516, 1991.

[Kas93] Masaki Kashiwara. The crystal base and Littelmann's refined Demazure character formula. *Duke Mathematical Journal*, 71(3):839–858, 1993.

[Koh90] Axel Kohnert. *Weintrauben, polynome, tableaux*. PhD thesis, Universität Bayreuth, 1990.

[Las01] Alain Lascoux. Transition on grothendieck polynomials. In *Physics and combinatorics*, pages 164–179. World Scientific, 2001.

[Las04] Alain Lascoux. Schubert & Grothendieck: un bilan bidécennal. *Séminaire Lotharingien de Combinatoire*, 50:B50i, 2004.

[LS82] Alain Lascoux and Marcel-Paul Schützenberger. Structure de hopf de l'anneau de cohomologie et de l'anneau de grothendieck d'une variété de drapeaux. *CR Acad. Sci. Paris Sér. I Math*, 295(11):629–633, 1982.

[LS89] Alain Lascoux and Marcel-Paul Schützenberger. Tableaux and noncommutative Schubert polynomials. *Functional Analysis and Its Applications*, 23(3):223–225, 1989.

[LS90] Alain Lascoux and Marcel-Paul Schützenberger. Keys & standard bases. *Institute for Mathematics and Its Applications*, 19:125–144, 1990.

[Mon16] Cara Monical. Set-valued skyline fillings. *arXiv preprint arXiv:1611.08777*, 2016.

[MPS20] Cara Monical, Oliver Pechenik, and Travis Scrimshaw. Crystal structures for symmetric Grothendieck polynomials. *Transformation Groups*, pages 1–51, 2020.

[PS19] Oliver Pechenik and Travis Scrimshaw. K-theoretic crystals for set-valued tableaux of rectangular shapes. *arXiv preprint arXiv:1904.09674*, 2019.

[PS23] Ekaterina Presnova and Evgeny Smirnov. Lascoux polynomials and subdivisions of Gelfand-Zetlin polytopes. *arXiv preprint arXiv:2312.01417*, 2023.

[PY23] Jianping Pan and Tianyi Yu. A bijection between k-kohnert diagrams and reverse set-valued tableaux. *The Electronic Journal of Combinatorics*, pages P4–26, 2023.

[RS95] Victor Reiner and Mark Shimozono. Key polynomials and a flagged Littlewood—Richardson rule. *Journal of Combinatorial Theory, Series A*, 70(1):107–143, 1995.

[RY13] Colleen Ross and Alexander Yong. Combinatorial rules for three bases of polynomials. *arXiv preprint arXiv:1302.0214*, 2013.

[RY21] Victor Reiner and Alexander Yong. The "Grothendieck to Lascoux" conjecture. *arXiv preprint arXiv:2102.12399*, 2021.

[SY23] Mark Shimozono and Tianyi Yu. Grothendieck-to-Lascoux expansions. *Transactions of the American Mathematical Society*, 2023.

[Wil13] Matthew J Willis. A direct way to find the right key of a semistandard young tableau. *Annals of Combinatorics*, 17(2):393–400, 2013.