SET-VALUED TABLEAUX RULE FOR LASCOUX POLYNOMIALS

TIANYI YU

Abstract. Lascoux polynomials generalize Grassmannian stable Grothendieck polynomials and may be viewed as K-theoretic analogs of key polynomials. The latter two polynomials have combinatorial formulas involving tableaux: Lascoux and Schützenberger gave a combinatorial formula for key polynomials using right keys; Buch gave a set-valued tableau formula for Grassmannian stable Grothendieck polynomials. We establish a novel combinatorial rule for Lascoux polynomials involving right keys and set-valued tableaux. Our rule recovers the tableaux formulas of key polynomials and Grassmannian stable Grothendieck polynomials. To prove our rule, we construct a new abstract Kashiwara crystal structure on set-valued tableaux. This construction answers an open problem of Monical, Pechenik and Scrimshaw in the context of abstract Kashiwara crystal.

1. Introduction

In this paper, we establish a combinatorial rule for Lascoux polynomials using a combinatorial proof. Lascoux polynomials, denoted by $\mathfrak{L}^{(\beta)}$, are a $\mathbb{Z}[\beta]-$basis for $\mathbb{Z}[\beta][x_1, x_2, \dots]$ indexed by weak compositions (infinite sequence of non-negative integers with finitely many positive entries). They are related to the following polynomials:

- Schur polynomials: denoted by s_λ, which are symmetric polynomials in $\mathbb{Z}[x_1, x_2, \dots]$ indexed by partitions (finite weakly decreasing sequence of positive integers). They played 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 $\mathbb{Z}[x_1, x_2, \dots]$ indexed by weak compositions. 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 $\mathbb{Z}[\beta][x_1, x_2, \dots]$ indexed by partitions. They are symmetric in the x variables. They represent Schubert classes in the connective K-theory of Grassmannians.

The relations between these four polynomials can be described as follows.

- Key polynomials generalize Schur polynomials. More explicitly, assume α is a weak composition whose first n entries are weakly increasing and all other entries are 0. Let λ be the partition we get when we sort α into a weakly decreasing sequence and remove the trailing 0s. Then

$$\kappa_\alpha = s_\lambda|_{x_{n+1}=x_{n+2}=\cdots=0}$$

Date: September 2021.

- Lascoux polynomials generalize Grassmannian stable Grothendieck polynomials in an analogous way.
- Grassmannian stable Grothendieck polynomials are K-theoretic analogs of Schur polynomials: $G_\lambda^{(0)} = s_\lambda$.
- Extending the above point, Lascoux polynomials may be viewed as K-theoretic analogs of key polynomials: $\mathfrak{L}_\lambda^{(0)} = \kappa_\lambda$.

Their relations are summarized in the following diagram:

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

Here is another perspective to see how Lascoux polynomials fit into the large picture. Lascoux and Schützenberger found an expansion of Schubert polynomials into key polynomials. This expansion was proved by Reiner and Shimozono [RS95]. Grothendieck polynomials are K-theoretic analogs of Schubert polynomials. Buch, Kresch, Shimozono, Tamvakis and Yong [BKS+08] generalized the symmetrized 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 expansions in both [RS95] and [BKS+08]. Shimozono and Yu [SY21] proved this conjecture.

Polynomials in the diagram above have combinatorial formulas that involve tableaux. Schur polynomials are generating functions of semistandard Young tableaux (SSYT):

$$s_\lambda = \sum_T x^{\mathrm{wt}(T)} \tag{1.1}$$

where the sum is over all SSYT with shape λ (see §2 for relevant definitions). Lascoux and Schützenberger generalized Equation (1.1) by providing a combinatorial formula for key polynomials (Equation (2.3)) using right keys. On the other hand, Buch generalized Equation (1.1) by establishing a set-valued tableaux (SVT) formula for Grassmannian stable Grothendieck polynomials (Equation (2.4)). We generalize all three formulas by providing a novel combinatorial formula for Lascoux polynomials involving both right keys and SVT.

There already exist various combinatorial formulas of Lascoux polynomials:

- 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].
- 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 reverse set-valued tableaux rule involving the left keys. It was then rediscovered by Shimozono and Yu [SY21]. This rule can also be rephrased into a form that involves reverse semistandard Young tableaux.
- Ross and Yong [RY13] conjectured a rule that involves diagrams. Their conjectural rule extends the Kohnert diagram rule for key polynomials. In

the special case where all positive numbers in α are the same, this conjecture is proved in [PS19].

We are going to provide another SVT rule for Lascoux polynomials (Theorem 3.17). In general, our rule and the rule in [BSW20] sum over different sets of SVT. Moreover, our rule is easier since it does not involve the Lusztig involution. In addition, we may view (3.17) from the tableau complex viewpoint [KMY08]. For each $\mathfrak{L}_\alpha^{(\beta)}$, the SVT we summed over form a simplicial complex. It is a sub-complex of the *Young tableau complex* in [KMY08].

To prove our result, we put an abstract Kashiwara crystal structure on SVT, which is not isomorphic to the crystal basis of a $U_q(\mathfrak{sl_n})-$representation. Our proof mimics Kashiwara's study of Demazure modules and crystal basis [Kas93]. Based on our crystal, we define i-strings similar to [Kas93]. A key step of our proof is Corollary 4.39, which is an analogous result of [Kas93, Proposition 3.3.5]. Notice that Monical, Pechenik and Scrimshaw [MPS20] have already defined a crystal structure on SVT. However, their construction is not compatible with $K_+(\cdot)$ introduced in §3. Besides being crucial in the proof, our crystal structure is a K-theoretic analogue of the Demazure crystal introduced in [Kas93]. It can also be viewed as a solution to [MPS20][Open Problem 7.1] in the context of abstract Kashiwara crystals.

The rest of the paper is organized as follows. In §2, we will give background. In §3, we define the right keys for SVT and introduce our main result Theorem 3.17. In §4, we construct a Kashiwara crystal on SVT and prove Theorem 3.17. In §5, we explain why our crystal can be viewed as a K-analogue of the Demazure crystal and an answer to [MPS20][Open Problem 7.1].

2. Background

2.1. **Lascoux Polynomials.** The symmetric group S_n acts on the polynomial ring $\mathbb{Z}[\beta][x_1, x_2, \dots]$ by permuting the x variables. Let $s_i \in S_n$ denote the transposition that swaps i and $i+1$. We define four operators on $\mathbb{Z}[\beta][x_1, x_2, \dots]$:

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

These four operators satisfy the braid relations.

A weak composition is an infinite sequence of nonnegative integers with finitely many positive entries. When we write a weak composition, we ignore the trailing 0s. Let α be a weak composition. We use α_i to denote the i^{th} entry of α. 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

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

2.2. **Tableaux.** In this subsection, we define a *tableau* as a filling of a diagram λ/μ with $\mathbb{Z}_{>0}$. A tableau is *normal* (resp. *antinormal*) if it is empty or has a unique northwestmost (resp. southeastmost) corner. A *semistandard Young tableau* (SSYT) is a tableau whose columns are strictly increasing and rows are weakly increasing. Let T be a SSYT. The *weight* of T, denoted by $\mathrm{wt}(T)$, is a weak composition whose i^{th} entry is the number of i in T. The *column order* is a total order on entries of T. It goes from left to right and from bottom to top within each column. The column word of T, denoted by $\mathrm{word}(T)$, is the word we get if we read entries of T in the column order.

A *key* is a normal SSYT where 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 $\mathrm{key}(\cdot)$ be the map that sends the weak composition to its corresponding key. Its inverse map is simply $\mathrm{wt}(\cdot)$. For instance,

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

The Knuth equivalence $\sim$ is defined on the set of all words 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 T, there exists a unique antinormal SSYT $T^{\searrow}$ such that $\mathrm{word}(T) \sim \mathrm{word}(T^{\searrow})$.

Each normal SSYT T is associated with a key called the *right key*. It has the same shape as T and is denoted by $K_+(T)$. 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 defined as the rightmost column of $T_{\geq j}^{\searrow}$. In §3, we will describe an easier way to compute $K_+(T)$.

Example 2.1. Let T be the following SSYT:

$$\begin{array}{|c|c|c|c|}\hline 1 & 2 & 4 & 7 \\ \hline 3 & 5 & 6 & \\ \cline{1-3} 4 & 8 & & \\ \cline{1-2} 6 & & & \\ \cline{1-1} \end{array}$$

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

$$\begin{array}{cccc} & & & \fbox{2} \\ & & \fbox{3} & \fbox{4} \\ & \fbox{1} & \fbox{5} & \fbox{7} \\ \fbox{4} & \fbox{6} & \fbox{6} & \fbox{8} \end{array}$$

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

$$\begin{array}{ccc} & & 4 \\ & 2 & 7 \\ 5 & 6 & 8 \end{array} \qquad \begin{array}{cc} & 4 \\ 6 & 7 \end{array} \qquad \begin{array}{c} 7 \end{array}$$

Thus, $K_+(T)$ is

$$\begin{array}{cccc} 2 & 4 & 4 & 7 \\ 4 & 7 & 7 & \\ 7 & 8 & & \\ 8 & & & \end{array}$$

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

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

2.3. **Abstract Kashiwara crystal.** We introduce the *Abstract Kashiwara crystals* defined in [BS17].

Definition 2.2. [BS17, Definition 2.13] An *abstract Kashiwara crystal* (for GL_m) is a nonempty set $\mathcal{B}$ together with the following maps:

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

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

$$\mathrm{wt} : \mathcal{B} \to \mathbb{Z}^m$$

where $i \in [m-1]$, satisfying the following:

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

$$\begin{aligned} \varepsilon_i(Y) &= \varepsilon_i(X) - 1, \\ \varphi_i(Y) &= \varphi_i(X) + 1, \\ \mathrm{wt}(Y) &= \mathrm{wt}(X) + v_i - v_{i+1}, \end{aligned}$$

where $v_1, \dots, v_n$ is the standard basis.

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

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

Now we describe a well-known example of the abstract Kashiwara crystal. Let $\mathcal{B}(\lambda, n)$ be the set of all SSYT whose shapes are λ and entries are in $[n]$. Take $T \in \mathcal{B}(\lambda, n)$. We replace each i [resp. $i+1$] in its column word by ")" [resp. "("] and remove all other numbers. The resulting word is called the *i-word* of T. Then we may pair "(" with ")" in the usual way.

Definition 2.3. Define $\epsilon_i(T)$ as the number of unpaired ”(” and $\phi_i(T)$ as the number of unpaired ”)”.

If $\phi_i(T) = 0$, then $f_i(T) := \mathbf{0}$. Otherwise, we can find the i in T that corresponds to the last unpaired ”)” in the i-word. $f_i(T)$ is obtained by changing this i into $i + 1$.

If $\epsilon_i(T) = 0$, 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. $e_i(T)$ is obtained by changing this $i + 1$ into i.

It is a well-known result that $B(n, \lambda)$, together with $e_i, f_i, \phi_i, \epsilon_i$ and wt, form an abstract Kashiwara crystal. Moreover, the operator f_i can be used to compute $SSYT(\alpha)$. Let S be a subset of $B(n, \lambda)$. Define $\mathcal{F}_i S$ as $\{(f_i)^j(T) : T \in S, j \geq 0\} - \{\mathbf{0}\}$.

Theorem 2.4 ([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*

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

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

SSYT(α), together with the maps, is known as a *Demazure crystal*.

2.4. **Set-valued Tableaux.** We start to view a *tableau* as a filling where entries are finite non-empty subsets of $\mathbb{Z}_{>0}$.

Definition 2.5. A *set-valued tableau* (SVT) is a filling of the Young diagram with non-empty subsets of $\mathbb{Z}_{>0}$ such that no matter how we pick one entry in each set, the resulting tableau is a SSYT.

Example 2.6. The following is a SVT:

1	13	36
23	47	
567		

where 23 represents the set $\{2, 3\}$. The following example is not a SVT:

1	14	46
23	47	
567		

Remark 2.7. A SSYT can be viewed as a SVT where each set is a singleton.

Definition 2.8. Let T be a SVT of shape λ. Let $\mathrm{wt}(T)$ be the weak composition whose i^{th} entry is the number of i's in T. Let $\mathrm{ex}(T)$ be the number $|\mathrm{wt}(T)| - |\lambda|$.

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

Now we can state a SVT rule for Grassmannian stable Grothendieck polynomials $G_\lambda^{(\beta)}$. Instead of defining $G_\lambda^{(\beta)}$, we restate its relation with Lascoux polynomials.

Assume α is a weak composition whose first n entries are weakly increasing and all other entries are all 0. Sort α and obtain the partition λ. Then

$$\mathfrak{L}_{\alpha}^{(\beta)} = G_{\lambda}^{(\beta)}|_{x_{n+1}=x_{n+2}=\cdots=0}$$

Buch [SB02] showed that:

$$G_{\lambda}^{(\beta)}|_{x_{n+1}=x_{n+2}=\cdots=0} = \sum_{T} \beta^{\mathrm{ex}(T)} x^{\mathrm{wt}(T)} \tag{2.4}$$

where the sum is over all SVT T whose shape is λ and whose entries are subsets of $[n]$.

3. The right keys

In this section, we first describe a direct way to compute $K_+(T)$. Then we generalize the right key to normal SVT. Finally, we introduce our main result.

3.1. **Compute right keys using the star operator.** Shimozono and Yu [SY21] used the following operator to compute right keys. [1]

Definition 3.1. 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 monoid of words with letters 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 3.2. We have

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

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

Remark 3.3. Similar to [SY21, Remark 4.7], we have $S \star w = S \star w'$, if w and w' are Knuth equivalent.

We may view each finite set of positive integers as one column of a SSYT. Under this convention, we have the following way to compute a right key.

Lemma 3.4. *Column j of $K_+(T_j)$ is $\emptyset \star \mathrm{word}(T_{\geq j})$.*

Proof. By definition, column j of $K_+(T_{\geq j})$ is the last column of $T_{\geq j}^{\searrow}$. Since $T_{\geq j}^{\searrow}$ is antinormal, $\emptyset \star \mathrm{word}(T_{\geq j}^{\searrow})$ equals the last column of $T_{\geq j}^{\searrow}$. Then the proof is finished by $\mathrm{word}(T_{\geq j}) \sim \mathrm{word}(T_{\geq j}^{\searrow})$ and Remark 3.3. □

Remark 3.5. Notice that this right key computing process is essentially the same as Willis' method [Wil13], but stated in a different way.

[1] Notice that we replace "smallest" by "largest" and "at least" by "at most". This is because [SY21] focused on reverse SSYT (tableaux whose rows are weakly decreasing and columns are strictly decreasing) while this paper focused on SSYT.

Example 3.6. Let T be the following SSYT:

$$\begin{array}{|c|c|c|c|}\hline 1 & 2 & 4 & 7 \\ \hline 3 & 5 & 6 \\ \cline{1-3} 4 & 8 \\ \cline{1-2} 6 \\ \cline{1-1}\end{array}$$

Then column 1 of $K_+(T)$ is $\emptyset \star 6431852647 = \{2, 4, 7, 8\}$. Column 2, 3 and 4 of $K_+(T)$ are: $\emptyset \star 852647 = \{4, 7, 8\}$, $\emptyset \star 647 = \{4, 7\}$ and $\emptyset \star 7 = \{7\}$. Thus, $K_+(T)$ is

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

which agrees with Example 2.1.

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

Definition 3.7. Let T be a normal SVT. We may pick a number in each entry of T and obtain a normal SSYT. Let $T_1, \ldots, T_M$ be all possible results. Define

$$T_{max} := max(K_+(T_1), \ldots, K_+(T_M))$$

where max is entry-wise.

Example 3.8. We start with the SVT T. If we pick one entry from each cell, we can get 2 possible SSYTs T_1 and T_2.

$$T = \begin{array}{|c|c|}\hline 1 & 23 \\ \hline 3 \\ \cline{1-1}\end{array} \quad , \quad T_1 = \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 \\ \cline{1-1}\end{array} \quad , \quad T_2 = \begin{array}{|c|c|}\hline 1 & 3 \\ \hline 3 \\ \cline{1-1}\end{array}$$

Then we compute the right keys of T_1 and T_2 separately, and get:

$$K_+(T_1) = \begin{array}{|c|c|}\hline 2 & 2 \\ \hline 3 \\ \cline{1-1}\end{array} \quad , \quad K_+(T_2) = \begin{array}{|c|c|}\hline 1 & 3 \\ \hline 3 \\ \cline{1-1}\end{array}$$

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 3.9. 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 3.10. For finite $S \subseteq \mathbb{Z}_{>0}$, let $\text{word}(S)$ be the word we get if we list numbers of S in increasing order. For a SVT T, let $\text{word}(T) := \text{word}(S_1) \cdots \text{word}(S_n)$, where $S_1, \dots, S_n$ are entries of T in the column order.

Now we may introduce the easier way to compute T_{max}:

Lemma 3.11. *Let T be a normal SVT. Column j of T_{max} is $\emptyset \star \text{word}(T_{\geq j})$, where $T_{\geq j}$ is obtained by removing the first $j-1$ columns of T.*

Example 3.12. Let T be the SVT in example 2.6. Then $\text{word}(T) = 567231471336$. Column 1 of T_{max} is $\emptyset \star 567231471336 = \{3, 6, 7\}$. Column 2 and 3 of T_{max} are $\emptyset \star 471346 = \{6, 7\}$ and $\emptyset \star 46 = \{6\}$ Thus,

$T_{max} =$

3	6	6
6	7	
7		

To prove Lemma 3.11, it is enough to assume $j = 1$. Let $S_1, \dots, S_n$ be the entries of T in the column order. Let $\mathcal{W}$ be the set of words $w_1 \cdots w_m$ where $w_i \in S_i$. Then

$$\mathcal{W} = \{\text{word}(T') : T' \text{ is obtained by picking one number in each entry of } T\}.$$

Recall that we are viewing finite subsets of $\mathbb{Z}_{>0}$ as columns of SSYT. While w ranges over $\mathcal{W}$, $\emptyset \star w$ ranges over all column 1 of $K_+(T')$, where T' is obtained by picking one number in each entry of T. Thus, what we want to prove becomes: The entry-wise maximum of $\emptyset \star w$, where w ranges over $\mathcal{W}$, is given by $\emptyset \star \text{word}(S_1) \cdots \text{word}(S_n)$.

In addition, assume column 1 of T has k numbers. Take any $w_1 \cdots w_n \in \mathcal{W}$. We know $\emptyset \star w_1 \cdots w_n$ has size k. Notice that $w_1 > \cdots > w_k$, so $\emptyset \star w_1 \cdots w_k$ also has size k. Thus,

$$|\emptyset \star w_1 \cdots w_i| = max(i, m).$$

It is enough to prove the following lemma.

Lemma 3.13. *Let $S_1, \dots, S_n$ be finite nonempty subsets of $\mathbb{Z}_{>0}$. Let $\mathcal{W}$ be the set of words $w_1 \cdots w_n$ where $w_i \in S_i$. In addition, assume for each $i \in [n]$, $\emptyset \star w_1 \cdots w_i$ has the same size for all $w_1 \cdots w_n \in \mathcal{W}$.*

Then the entry-wise maximum of $\emptyset \star w$, where w ranges over $\mathcal{W}$, is given by $\emptyset \star \text{word}(S_1) \cdots \text{word}(S_n)$.

Proof. We prove by induction on n. When $n = 1$, our claim is immediate. Now we assume $n > 1$. Let k be the size of $\emptyset \star w$ for any $w \in \mathcal{W}$. Use $A = \{a_1 < a_2 < \cdots\}$ to denote $\emptyset \star \text{word}(S_1) \cdots \text{word}(S_{n-1})$. Take any $w = w_1 \cdots w_n \in \mathcal{W}$. Use $B^w = \{b_1^w < b_2^w \cdots\}$ to denote $\emptyset \star w_1 \cdots w_{n-1}$. By the inductive hypothesis, A is the entry-wise maximum of B^w where w ranges over $\mathcal{W}$. We consider two cases:

- Case 1: $|B^w| = k - 1$ for all $w \in W$. Since $|\emptyset \star w| = k$, we know $\emptyset \star w = \{w_n < b_1^w < \cdots < b_{k-1}^w\}$. Thus, $max(S_n) < b_1^w \leq a_1$. We have

$$\begin{aligned} \emptyset \star \text{word}(S_1) \cdots \text{word}(S_n) &= \{a_1 < \cdots < a_{k-1}\} \star \text{word}(S_n) \\ &= \{max(S_n) < a_1 < \cdots < a_{k-1}\}. \end{aligned}$$

  Now we consider the entry-wise maximum of $\emptyset \star w$ where $w \in \mathcal{W}$. The smallest number is $max(S_n)$. The j^{th} smallest number is $max_{w \in \mathcal{W}}(b_{j-1}^w) = a_{j-1}$ for $j > 1$. Thus, the entry-wise maximum is $\{max(S_n) < a_1 < \cdots < a_{k-1}\}$.

- Case 2: $|B^w| = k$ for all $w \in W$. In this case, we claim $min(S_n) \geq a_1$. Otherwise, we may pick $w_1, \dots, w_{n-1}$ such that the smallest number in $\emptyset \star w_1 \cdots w_{n-1}$ is a_1. Then if we pick $w_n = min(S_n)$, $\emptyset \star w_1 \cdots w_n$ would have $k+1$ numbers, contradiction.

  Next, we partition S_n into k parts: $S_n^1, \dots, S_n^k$. Let $S_n^j := S_n \cap [a_j, a_{j+1})$, where $a_{k+1} = \infty$ by convention. Consider the action of $\text{word}(S_n) = \text{word}(S_n^1) \cdots \text{word}(S_n^k)$ on $\{a_1, \dots, a_k\}$. When $\text{word}(S_n^j)$ acts, a_j is still in the set. It will first be bumped by the smallest element in S_n^j, which is then bumped by the second smallest number. Eventually, the action of $\text{word}(S_n^j)$ replaces a_j by $max(S_n^j)$. Thus, if we let $\{\overline{a_1} < \cdots < \overline{a_k}\}$ be $\emptyset \star \text{word}(S_1) \dots \text{word}(S_n)$, we have $\overline{a_j} = max(S_n^j \cup \{a_j\})$.

  Now take any $w = w_1 \cdots w_n \in \mathcal{W}$. We check $\emptyset \star w$ is entry-wise weakly less than $\{\overline{a_1} < \cdots < \overline{a_k}\}$. We know $b_i^w \leq a_i \leq \overline{a_i}$. Assume w_n bumps b_j^w when it acts on $\{b_1^w < \cdots < b_k^w\}$. We only need to check $w_n \leq \overline{a_j}$. Notice that $w_n < b_{j+1}^w \leq a_{j+1}$, where $b_{k+1}^w = \infty$ by convention. Thus, $w_n \in S_n \cap (0, a_{j+1})$, so $w_n \leq \overline{a_j}$.

  Finally, we check for each $j \in [k]$, we can find $w = w_1 \cdots w_n \in \mathcal{W}$ such that the j^{th} smallest number in $\emptyset \star w$ is $\overline{a_j}$. If S_n^j is not empty, we know $\overline{a_j} = max(S_n^j)$. By our inductive hypothesis, we may pick $w_1, \dots, w_{n-1}$ such that $b_{j+1}^w = a_{j+1}$. Then if we let $w_n = \overline{a_j}$, it bumps b_j^w when it acts on B^w, since $b_j^w \leq \overline{a_j} < a_{j+1}^w$. Now assume S_n^j is empty, then $\overline{a_j} = a_j$. Pick $w_1, \dots, w_{n-1}$ such that $b_j^w = a_j$. Then we can find $w_n < a_j$ or $w_n \geq a_{j+1}$. In either case, w_n will not bump a_j.

□

Corollary 3.14. *T_{max} is a key .*

Proof. Let j be a positive integer. By Lemma 3.11, it remains to show

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

This is implied by [SY21, Lemma 4.9] . □

Definition 3.15. The right key of a SVT T is $K_+(T) := T_{max}$. Let $\text{SVT}(\alpha)$ be the set of all T such that $K_+(T) \leq \text{key}(\alpha)$.

Remark 3.16. By the definition of $K_+(\cdot)$, we may also describe $\text{SVT}(\alpha)$ as: It consists of all SVT T such that no matter how you pick one number from each entry of T, the result is in $\text{SSYT}(\alpha)$.

We end this section by introducing our main result:

Theorem 3.17. *Let α be a weak composition. Then*

$$\mathfrak{L}_\alpha^{(\beta)} = \sum_{T \in \text{SVT}(\alpha)} x^{\text{wt}(T)} \beta^{\text{ex}(T)}. \tag{3.1}$$

Example 3.18. Let $\alpha = (1, 0, 2)$. Then $\text{SVT}(\alpha)$ consists of the following:

1	1
2	

1	2
2	

1	1
3	

1	3
2	

1	3
3	

1	12
2	

1	1
23	

1	13
2	

1	3
23	

1	13
3	

1	23
2	

1	123
2	

1	13
23	

Thus, we may write $\mathfrak{L}^{(\beta)}_{(1,0,2)}$ as

$$\begin{aligned}&x_1^2x_2 + x_1x_2^2 + x_1^2x_3 + x_1x_2x_3 + x_1x_3^2\\ +&\beta(x_1^2x_2^2 + 2x_1^2x_2x_3 + x_1x_2x_3^2 + x_1^2x_3^2 + x_1x_2^2x_3)\\ +&\beta^2(x_1^2x_2^2x_3 + x_1^2x_2x_3^2)\end{aligned}$$

Equation (3.1) recovers the two combinatorial rules in §2:

- If we set $\beta = 0$, then the left hand side of Equation (3.1) becomes κ_α. In the right hand side, only T with $\mathrm{ex}(T) = 0$ can survive in the sum. Clearly, $\{T \in \mathrm{SVT}(\alpha) : \mathrm{ex}(T) = 0\} = \mathrm{SSYT}(\alpha)$. We recover Equation (2.3).
- Assume α is a weak composition whose first n entries are weakly increasing and the other entries are all 0. In each column of $\mathrm{key}(\alpha)$, there are $n, n-1, n-2, \ldots$. Then $T \in \mathrm{SVT}(\alpha)$ if and only if T has the shape α^+ and entries of T are subsets of $[n]$. Thus, we recover Equation (2.4).

We will prove Theorem 3.17 in the next section.

4. Abstract Kashiwara crystals on SVT

As stated in §2, $\mathrm{SSYT}(\alpha)$ can be computed using the f_i operators. To prove our result, we can construct an abstract Kashiwara crystal for GL_n on the set of SVT with entries in $[n]$. The first step is to construct an abstract Kashiwara crystal on certain words.

4.1. Constructing an abstract Kashiwara crystal on words.

Definition 4.1. Let $\mathcal{B}_{word}$ be a set of finite words over the 3-letter alphabet: "(", ")", and "-". It consists of all concatenations of contiguous subwords: "(", ")", and ") − (".

Example 4.2. $)) - (()) - ()) - (() - ($ is in $\mathcal{B}_{word}$ while $))(-($ is not.

In this subsection, we define an abstract Kashiwara crystal for GL_2 on $\mathcal{B}_{word}$. First, we define $\mathrm{wt}(\cdot)$.

Definition 4.3. Take $w \in \mathcal{B}_{word}$, then $\mathrm{wt}(w)$ is a length-2 vector where its first entry is the number of ")" in w and its second entry is the number of "(" in w.

Next, we define ε_1 and φ_1. Their definitions require the following process. Take $w \in \mathcal{B}_{word}$. Ignore its " − " 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 class.
- For each ") − (", these three characters are in the same class.

It is easy to see that each equivalence class is a contiguous subword. For instance, the word above is partitioned into 4 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, each class must have one of the following forms:

- *null-form*: This class does not have unpaired "(" or ")". For example, "$(()-())-()$".
- *left-form*: This class does not have unpaired ")" but has exactly one unpaired "(". For example, "$(())-($".
- *right-form*: This class does not have unpaired "(" but has exactly one unpaired ")". For example, "$)-()-()$".
- *combined-form*: This class has exactly one unpaired ")" and exactly one unpaired "(". For example, "$)-()-(())-($".

In the previous example, ")" and "$)-(())-()$" are in right-form. "$)-($" is in the combined form. "$()-($" is in the left form.

Next, we introduce a way to transform between left-forms, right-forms and combined-forms. If a class is in the left-form, then it must be "(" or "$(w)-($", where w is some word. Similarly, if a class is in the right-form, then it must be ")" or "$)-(w)$". If a class is in the combined-form, then it must be "$)-($" or "$)-(w)-($". We may describe the transformations between these three forms in the following table.

	Case 1	Case 2
left	$($	$(w)-($
right	$)$	$)-(w)$
combined	$)-($	$)-(w)-($

For instance, if we transform ")", "$)-()-($" and "$)-(()())$" into left-forms, we get "(", "$()-($" and "$(()())-($".

Now, 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. Define $\varepsilon_1(w)$ to be the number of left-forms in w. Define $\varphi_1(w)$ to be the number of right-forms in w. Then we can define $f_1 : \mathcal{B}_{word} \to \mathcal{B}_{word} \sqcup \{\mathbf{0}\}$.

Definition 4.4. Take $w \in \mathcal{B}_{word}$. If $\varphi_1(w) = 0$, then f_1 sends it to $\mathbf{0}$. Otherwise, if there is no combined-form, then f_1 transforms the last right-form into its left-form. If there is a combined-form, then f_1 transforms the combined-form into its left-form and transforms the last right-form into its combined-form.

Example 4.5. We have

$$)-()\,)\,)-(\,(\quad \xrightarrow{f_1} \quad)-()\,)-(\,(\,(.$$

Remark 4.6. Assume $f_1(w) \neq \mathbf{0}$. Then f_1 must do one of the following to w:

- Changes an unpaired ")" into a "(". In this case, w has no combined-forms.
- Changes a "$)-($" into "(" and changes a ")" into "$)-($".

Thus, it always increases the number of "(" by 1 and decreases the number of ")" by 1.

The e_1 operator can be defined similarly.

Definition 4.7. Take $w \in \mathcal{B}_{word}$. If $\varepsilon_1(w) = 0$, then e_1 sends it to $\mathbf{0}$. Otherwise, if there is no combined-form, then e_1 transforms the first left-form into its right-form. If there is a combined-form, then e_1 transforms the combined-form into its right-form and transforms the first left-form into its combined-form.

Lemma 4.8. *$\mathcal{B}_{word}$ together with $f_1, e_1, \varepsilon_1, \varphi_1$ and* wt *is an abstract Kashiwara crystal.*

Proof. We check the axioms.

K1: Take $X, Y \in \mathcal{B}_{word}$. $e_1(X) = Y$ if and only if $f_1(Y) = X$ is immediate. Now assume this is the case. Y has one more right-form than X, so $\varphi_1(Y) = \varphi_1(X) + 1$. Y has one less left-form than X, so $\varepsilon_1(Y) = \varepsilon_1(X) - 1$.

Finally, we need to check Y has one more ")" and one less "(" than X. This is done by Remark 4.6.

K2: Take $X \in \mathcal{B}_{word}$. $\langle \mathrm{wt}(X), (1, -1)\rangle$ is the number of ")" in X minus the number of "(" in X.

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, $\langle \mathrm{wt}(X), (1, -1)\rangle$ is the number of right-forms in X minus the number of left-forms in X, which is $\varphi_1(X) - \varepsilon_1(X)$. □

Beyond these axioms, f_1 and e_1 have some other useful properties. We start with a definition.

Definition 4.9. A *string* is a sequence of words $w_0, \dots, w_k \in \mathcal{B}_{word}$ satisfying:

- $e_1(w_0) = f_1(w_k) = \mathbf{0}$
- $f_1(w_j) = w_{j+1}$ for each $j \in \{0, 1, \dots, k-1\}$.

We say w_0 is the *source* of its string. Diagrammatically, we can represent the string as:

$$\mathbf{0} \xleftarrow{e_1} w_0 \xrightarrow{f_1} w_1 \xrightarrow{f_1} w_2 \cdots \xrightarrow{f_1} w_k \xrightarrow{f_1} \mathbf{0}$$

Clearly, $\mathcal{B}_{word}$ can be broken into a disjoint union of strings. The weight of each string satisfies the following.

Lemma 4.10. *Let $w_0, \dots, w_k$ be a string. Then $\mathrm{wt}(w_j) = s_1 \mathrm{wt}(w_{k-j})$ for each $j \in \{0, 1, \dots, k\}$, where s_1 is the operator that swaps the two entries.*

Proof. By (K2), $\mathrm{wt}(w_0) = (a + k, a)$ for some a. Then we apply f_1 for j times and obtain w_j. By (K1), $\mathrm{wt}(w_j) = (a + k - j, a + j)$. Thus, $\mathrm{wt}(w_j) = s_1 \mathrm{wt}(w_{k-j})$. □

Another useful property of f_1 and e_1 is that they have "square roots":

Definition 4.11. Define

$$f_1', e_1' : \mathcal{B}_{word} \cup \{\mathbf{0}\} \to \mathcal{B}_{word} \cup \{\mathbf{0}\}$$

based on the following cases:

1. $f_1'(\mathbf{0}) = e_1'(\mathbf{0}) = \mathbf{0}$
2. Assume $w \in \mathcal{B}_{word}$ has a combined form. f_1' transforms the combined-form into a left-form. e_1' transforms the combined-form into a right-form.
3. Assume $w \in \mathcal{B}_{word}$ has no combined-form. f_1' transforms the last right-form into its combined-form, or sends w to $\mathbf{0}$ if it has no right-form. e_1' transforms the first left-form into its combined-form, or sends w to $\mathbf{0}$ if it has no left-form.

Lemma 4.12. *For any $X, Y \in \mathcal{B}_{word}$, we have $f_1'(X) = Y$ if and only if $e_1'(Y) = X$.*

Proof. Immediate from the definition. ☐

Example 4.13. For example

$$)-()))-((\xrightarrow{f_1'})-())((\xrightarrow{f_1'})-())-(((\,$$

Compare the example above with Example 4.5, we observe that applying f_1' twice on this word is the same as applying f_1. This is true in general:

Lemma 4.14. *For any $w \in \mathcal{B}_{word}$, $f_1'(f_1'(w)) = f_1(w)$ and $e_1'(e_1'(w)) = e_1(w)$.*

Proof. Immediate from the definitions. ☐

This lemma allows us to view f_1' and e_1' as the "square roots" of f_1 and e_1. Next, we make a remark analogous to Remark 4.6. It explicitly describes what f_1' does.

Remark 4.15. Assume $f_1'(w) \neq \mathbf{0}$. Then f_1' must do one of the following to w:

- If w has no combined form, f_1' changes the last character of the last right-form, which is ")", into a ") − (".
- If w has a combined form, f_1' changes the first 3 characters of the combined-form, which are ") − (", into "(".

Thus, it always decreases the number of "(" by 1 or increases the number of ")" by 1.

With f_1', we may define an analogy of strings.

Definition 4.16. A *double string* is a sequence of words $w_0, \ldots, w_k \in \mathcal{B}_{word}$ satisfying:

- $e_1'(w_0) = f_1'(w_k) = \mathbf{0}$
- $f_1'(w_j) = w_{j+1}$ for each $j \in \{0, 1, \ldots, k-1\}$.

We say w_0 is the *source* of its double string. Diagrammatically, we can represent the double string as:

$$\begin{array}{ccccccccccc}
w_0 & \longrightarrow & w_2 & \longrightarrow & w_4 & \longrightarrow & \cdots & \longrightarrow & w_{k-2} & \longrightarrow & w_k \\
\downarrow & \nearrow & \downarrow & \nearrow & \downarrow & & & & \downarrow & \nearrow & \\
w_1 & \longrightarrow & w_3 & \longrightarrow & w_5 & \longrightarrow & \cdots & \longrightarrow & w_{k-1} & &
\end{array}$$

where solid arrow represents f_1 and dash arrow represents f_1'.

A double string can be viewed as a refinement of the "i-K-string" in [MPS20]. If we remove all f_1' arrows except the one from w_0 to w_1, we get an 1-K-string.

Lemma 4.17. *Let $w_0, \ldots, w_k$ be a double string. Then we have the following:*

(1) *k is even.*
(2) *w_j has a combined form if j is odd and has no combined form if j is even.*
(3) *$w_0, w_2, \ldots, w_k$ form a string and $w_1, w_3, \ldots, w_{k-1}$ form a string.*
(4) $\mathrm{wt}(w_{2j+1}) = \mathrm{wt}(w_{2j}) + (0, 1)$.
(5) $\mathrm{wt}(w_{2j}) = \mathrm{wt}(w_{2j-1}) + (-1, 0)$.

Proof. We know w_0 and w_k have no combined-form. By the definition f_1', w_{j+1} has a combined-form if and only if w_j has no combined-form. Thus, we have (1) and (2). (3) is immediate from the diagram. (4) and (5) follow from Remark 4.15.

☐

4.2. **Constructing abstract Kashiwara crystals on SVT.** Let $\mathcal{B}_n$ be the set of SVT with entries in $[n]$. We would like to turn $\mathcal{B}_n$ into an abstract Kashiwara crystal. First, we let $\mathrm{wt}(\cdot)$ be the weight function on SVT defined above. Now we define the remaining maps.

Definition 4.18. Let $T \in \mathcal{B}_n$. We pick an $i \in [n-1]$ and define the *i-word* of T as the following element of $\mathcal{B}_{word}$:

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 ") − (".

Definition 4.19. $\varepsilon_i(T)$ and $\varphi_i(T)$ are defined as ε_1 and φ_1 of T's i-word.

Now we would like to define f_i on $\mathcal{B}_n$. We do so by first defining f_i':

Definition 4.20. Define f_i' on $\mathcal{B}_n \cup \{\mathbf{0}\}$. $f_i'(\mathbf{0}) = \mathbf{0}$. Now take $T \in \mathcal{B}_n$. Apply f_1' on the i-word of T. Change T accordingly and obtain $f_i'(T)$. More explicitly, based on Remark 4.15, we may describe f_i' as the following:

- Assume f_1' sends the i-word of T to $\mathbf{0}$. Then $f_i'(T) = \mathbf{0}$.
- Assume f_1' changes a ")" into a ") − (". Find the set in T that corresponds to this ")" and add $i+1$ to this set.
- Assume f_1' changes a ") − (" into "(". Find the set in T that corresponds to this ") − (" and remove the i in it.

Example 4.21. For instance, if $i = 2$ and :

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

Then the 2-word is "()) − ()) − (". f_1' would change this word into "()) − ()(". Change T correspondingly and get:

$$f_2'(T) = \begin{array}{|c|c|c|c|}\hline 1 & 1 & 2 & 3 \\ \hline 2 & 23 \\ \cline{1-2} 34 \\ \cline{1-1}\end{array}$$

Lemma 4.22. *f_i' is well-defined. That is, for any $T \in \mathcal{B}_n$, $f_i'(T)$ is a SVT or $\mathbf{0}$.*

Proof. We assume $f_i'(T) \neq \mathbf{0}$. We consider what f_i' does on the i-word of T.

- Assume f_1' changes a ")" into a ") − (". Let r denote this ")". In the i-word of T, we know there is no combined-form and r is the last character of the last right-form. Let S be the entry in T corresponding to r. Let $S_\downarrow$ be the entry below S and $S_\rightarrow$ be the entry on the right of S. We need to check (1) $S_\downarrow$, if exists, has no $i+1$; (2) $S_\rightarrow$, if exists, has no i.

  Assume (1) is false. In the i-word of T, there is a "(" immediately before r. Then r cannot be the last character of a right-form. Contradiction.

  Assume (2) is false. Since there is no $i+1$ in $S_\downarrow$ if it exists, there are 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 r. Contradiction.

- Assume f'_1 changes a ")$-$(" into "(". Then f'_i removes i from a set containing both i and $i+1$. It will not cause any violation.

$\square$

Definition 4.23. Define $f_i : \mathcal{B}_n \to \mathcal{B}_n \cup \{\mathbf{0}\}$ as $f_i(T) = f'_i(f'_i(T))$.

Remark 4.24. Equivalently, we may define $f_i(T)$ as: Apply f_1 on the i-word of T and change T accordingly.

We may define e'_i and e_i similarly.

Remark 4.25. When we restrict f_i, e_i, ϕ_i and ϵ_i on a SSYT, we can recover the classical construction described in §2.

Lemma 4.26. *$\mathcal{B}_n$ together with $f_i, e_i, \varepsilon_i, \varphi_i$ and* wt *is an abstract Kashiwara crystal.*

Proof. Follows directly from Lemma 4.8. $\square$

Similar to $\mathcal{B}_{word}$, we can make the following definition:

Definition 4.27. An *i-string* is a sequence $T_0, \dots, T_k \in \mathcal{B}_n$ satisfying:

- $e_i(T_0) = f_i(T_k) = \mathbf{0}$
- $f_i(T_j) = T_{j+1}$ for each $j \in \{0, 1, \dots, k-1\}$.

We say T_0 is the *source* of its i-string.
A *double i-string* is a sequence $T_0, \dots, T_k \in \mathcal{B}_n$ satisfying:

- $e'_i(T_0) = f'_i(T_k) = \emptyset$
- $f'_i(T_j) = T_{j+1}$ for each $j \in \{0, 1, \dots, k-1\}$.

We say T_0 is the *source* of its double i-string.

Analogous results of strings and double strings in $\mathcal{B}_{word}$ hold:

Lemma 4.28. *Let $T_0, \dots, T_k$ be an i-string. Then $\mathrm{wt}(w_j) = s_i \mathrm{wt}(w_{k-j})$ for each $j \in \{0, 1, \dots, k\}$, where s_i is the operator that swaps the i^{th} and $(i+1)^{th}$ entries.*

Proof. Immediate from Lemma 4.10. $\square$

Lemma 4.29. *Let $T_0, \dots, T_k$ be a double string. Then we have the following:*

1. *k is even.*
2. *i-string of T_j has a combined form if j is odd and has no combined form if j is even.*
3. *$T_0, T_2, \dots, T_k$ form an i-string and $T_1, T_3, \dots, T_{k-1}$ form an i-string.*
4. $\mathrm{wt}(T_{2j+1}) = \mathrm{wt}(T_{2j}) + v_{i+1}$.
5. $\mathrm{wt}(T_{2j}) = \mathrm{wt}(T_{2j-1}) - v_i$.

Proof. Immediate from Lemma 4.17. $\square$

4.3. **Double i-string and the right key.** This subsection investigates how the right key is changed in an double i-string. More explicitly, we prove:

Lemma 4.30. *Let $T_0, T_1, \dots, T_k$ be a double i-string. Assume $K_+(T_0) = \mathrm{key}(\alpha)$. Then $\alpha_i \geq \alpha_{i+1}$, and there are two possibilities:*

- $K_+(T_1) = \cdots = K_+(T_k) = \mathrm{key}(\alpha)$.
- $K_+(T_1) = \cdots = K_+(T_k) = \mathrm{key}(s_i \alpha)$.

The main goal of this subsection is to prove Lemma 4.30. by investigating how f'_i and e'_i change the right key of a SVT whose i-string has a combined-form. First, we need a few lemmas about the $\star$ operator.

Lemma 4.31. *Let S be a 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. We perform an induction on length of w. If w is the empty word, $S \star iw = S \star i$, which must have i.

Now assume w has positive length. We may write w as $w'j$, where $j \in \mathbb{Z} - \{i+1\}$. By the inductive hypothesis, there are 2 possibilities: $i+1 \notin S \star iw'$ or $i, i+1 \in S \star iw'$. Now we consider the action of j on $S \star iw'$ in these two cases:

(1) Assume $i+1 \notin S \star iw'$. The action of j will not make $i+1$ appear, so $i+1 \notin S \star iw'j$
(2) Assume $i, i+1 \in S \star iw'$. No matter what j is, i will always be in $S \star iw'j$. □

Lemma 4.32. *Let S_1, S_2 be two subsets of $\mathbb{Z}$. We say $S_2 = \partial_i S_1$ if one of the following happens:*

- *$i, i+1 \in S_1$ and $S_1 = S_2$;*
- *$i \notin S_1$, $i+1 \notin S_1$ and $S_1 = S_2$;*
- *$i \in S_1$, $i+1 \notin S_1$ and $S_2 = (S_1 - \{i\}) \cup \{i+1\}$.*

Now assume $S_2 = \partial_i S_1$. Then

(1) *for any $x \neq i$ or $i+1$, we have $S_2 \star x = \partial_i(S_1 \star x)$ or $S_2 \star x = S_1 \star x$;*
(2) *$S_2 \star (i+1) = S_1 \star (i+1)$.*

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 $S_1 \neq S_2$. We know S_2 is obtained by changing the i in S_1 into $i+1$. 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 $S_2 \star x = S_1 \star x$. Similarly, $i+1$ must bump i in S_1 and $i+1$ in S_2, so $S_2 \star (i+1) = S_1 \star (i+1)$. □

Lemma 4.33. *$K_+(f'_i(T)) = K_+(T)$ if T has a combined-form in its i-string.*

Proof. Assume f'_i removes i from the entry S, which is in column c of T. Assume S is in column c. Then clearly column j of $K_+(T)$ and $K_+(f'_i(T))$ 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 $K_+(f'_i(T))$ is $\emptyset \star u_1 \;\; (i+1) \;\; u_2$. Thus, it remains to prove:

$$(\emptyset \star u_1) \star \;\; i \;\; (i+1) = (\mathbf{0} \star u_1) \star \;\; (i+1)$$

Now we think about 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 4.31, 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.

$$(\emptyset \star u_1) \star i \ \ (i+1) = \emptyset \star u_1 = (\emptyset \star u_1) \star (i+1)$$

□

Similarly, for e_i', we have:

Lemma 4.34. *Assume T has a combined-form in its i-string. Assume $K_+(T) = \text{key}(\alpha)$. If T also has a left-form, then $K_+(e_i'(T)) = \text{key}(\alpha)$. If T has no left-form, then $K_+(e_i'(T)) = \text{key}(\alpha)$ or $\text{key}(s_i\alpha)$.*

Proof. Assume e_i' removes $i+1$ from the entry S, which is in column c of T. Assume S is in column c. 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 $u = \mathit{word}(T_{\geq j})$. 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 $S_2 = (\emptyset \star u_1) \star i \ (i+1)$. 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 is obtained by changing the i in S_1 into $i+1$. Using notation in Lemma 4.32, we may say $S_2 = \partial_i S_1$.

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, u_2 has two possibilities: has neither i nor $i+1$, or has the form $u_2^1 \ \ (i+1) \ \ u_2^2$, where u_2^1 has no i. By Lemma 4.32, we have $S_1 \star u_2 = S_2 \star u_2$, or $S_2 \star u_2$ is obtained from $S_1 \star u_2$ by changing i into $i+1$. Moreover, 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 4.30.

Proof of Lemma 4.30. 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 4.31, 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 4.33, we know $K_+(T_j) = K_+(T_{j+1})$ where $j = 1, 3, \ldots, k-1$. By Lemma 4.34 we know $K_+(T_j) = K_+(T_{j-1})$ where $j = 3, 5, \ldots, k-1$. Thus, $T_1, \ldots, T_k$ all have the same right key. By Lemma 4.34, since T_1 has no left-form, we have $K_+(T_1) = \text{key}(\alpha)$ or $\text{key}(s_i\alpha)$. □

Corollary 4.35. *Assume T is a SVT such that $f_i(T) \neq \mathbf{0}$. If $K_+(T) \neq K_+(f_i(T))$, then T must be the source of its double i-string.*

4.4. **proof of Theorem 3.17.** In this subsection, we derive a few lemmas and then use them to prove Theorem 3.17. First, we notice that the generating function of each double i-string behaves nicely under π_i.

Lemma 4.36. *For each i-string $T_0, \ldots, T_k$, we have*

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

Proof. Write $x^{\mathrm{wt}(T_0)}$ as $m x_i^a x_{i+1}^b$, where m is a monomial with no x_i or x_{i+1}. By Lemma 4.28, $x^{\mathrm{wt}(T_0)} = m x_i^b x_{i+1}^a$. Thus, $k = b - a$. Finally, we have

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

□

Similarly, the generating function of each double i-string behaves nicely under $\pi_i^{(\beta)}$. The following lemma is known to authors of [MPS20].

Lemma 4.37. *For each double i-string $T_0, \ldots, T_{2k}$, we have*

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

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

Proof. For the first equation, notice that

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

Thus, its left hand side becomes

$$\pi_i(x^{\mathrm{wt}(T_0)}\beta^{\mathrm{ex}(T_0)} + \beta x_{i+1} x^{\mathrm{wt}(T_0)}\beta^{\mathrm{ex}(T_0)}) = \pi_i(x^{\mathrm{wt}(T_0)}\beta^{\mathrm{ex}(T_0)} + x^{\mathrm{wt}(T_1)}\beta^{\mathrm{ex}(T_1)}).$$

Then the first equation is established by Lemma 4.36.

For the second equation, notice that $\sum_{j=0}^{2k} x^{\mathrm{wt}(T_j)}\beta^{\mathrm{ex}(T_j)}$ 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 $\mathrm{SVT}(\alpha)$ in terms of double i-strings.

Lemma 4.38. *Take any weak composition α. For each double i-string $T_0, \ldots, T_{2k}$, if $T_i \in \mathrm{SVT}(\alpha)$ with $i > 0$, then $T_0, \ldots, T_{2k} \in \mathrm{SVT}(\alpha)$.*

Proof. We know $K_+(T_i) \leq \mathrm{key}(\alpha)$. Since $T_1, \ldots, T_{2k}$ all have the same right key, they are all in $\mathrm{SVT}(\alpha)$. By Lemma 4.30, $K_+(T_0) \leq K_+(T_i)$, so $T_0 \in \mathrm{SVT}(\alpha)$. □

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

Corollary 4.39. *Take any weak composition α. For each double i-string $T_0, \dots, T_{2k}$, $\mathrm{SVT}(\alpha)$ contains all of them, or none of them, or only T_0.*

Lemma 4.40. *Let α be a weak composition such that $\alpha_i > \alpha_{i+1}$. We can decompose $\mathrm{SVT}(s_i\alpha)$ into a disjoint union of double i-strings. For each of the double i-string in $\mathrm{SVT}(s_i\alpha)$, $\mathrm{SVT}(\alpha)$ either contains its source or all of it.*

Proof. Let $T_0, \dots, T_{2k}$ be a double i-string that intersects with $\mathrm{SVT}(s_i\alpha)$. Corollary 4.39 implies $T_0 \in \mathrm{SVT}(s_i\alpha)$. Let $\gamma = wt(K_+(T_0))$, then $\mathrm{key}(\gamma) \leq \mathrm{key}(s_i\alpha)$. We know each SVT in this double i-string has right key $\mathrm{key}(\gamma)$ or $\mathrm{key}(s_i(\gamma))$. Since $\alpha_i > \alpha_{i+1}$, $\mathrm{key}(s_i\gamma) \leq \mathrm{key}(s_i\alpha)$. Thus, the whole double i-string is in $\mathrm{SVT}(s_i\alpha)$.

Lemma 4.30 implies that $\gamma_i \geq \gamma_{i+1}$, so $\mathrm{key}(\gamma) \leq \mathrm{key}(\alpha)$. We have $T_0 \in \mathrm{SVT}(\alpha)$. By Corollary 4.39, $\mathrm{SVT}(\alpha)$ either contains T_0 or the whole double i-string. □

Now we are ready to prove our main result:

Proof of Theorem 3.17. Only need to check $\sum_{T\in\mathrm{SVT}(\alpha)} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)}$ satisfies the recursive definition of $\mathfrak{L}_\alpha^{(\beta)}$. In other words, we need to prove:

- If α is a partition, then
$$\sum_{T\in\mathrm{SVT}(\alpha)} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)} = x^\alpha.$$
- If $\alpha_i > \alpha_{i+1}$, then
$$\pi_i^{(\beta)}\Big(\sum_{T\in\mathrm{SVT}(\alpha)} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)}\Big) = \sum_{T\in\mathrm{SVT}(s_i\alpha)} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)} \tag{4.1}$$

The first statement is immediate. For the second one, we break $\mathrm{SVT}(\alpha)$ into $A \sqcup B$. The set A consists of all T whose whole double i-string is in $\mathrm{SVT}(\alpha)$. The set B contains all $T \in \mathrm{SVT}(\alpha)$ such that part of its double i-string is not in $\mathrm{SVT}(\alpha)$. Let $\overline{B}$ be the union of double i-strings who intersect with B. By Lemma 4.40, elements in B are sources of double i-string and $\mathrm{SVT}(s_i\alpha) = A \sqcup \overline{B}$. Now by Lemma 4.37,
$$\pi_i^{(\beta)}\Big(\sum_{T\in A} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)}\Big) = \sum_{T\in A} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)},$$
$$\pi_i^{(\beta)}\Big(\sum_{T\in B} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)}\Big) = \sum_{T\in \overline{B}} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)}.$$

Equation 4.1 is obtained by summing up the two equations above.

□

5. K-theory crystal

In this section, we describe some similarities between our abstract Kashiwara crystal and the Demazure crystal. Then we explain why our crystal can be viewed as an answer to [MPS20][Open Problem 7.1].

Similar to the $\mathcal{F}_i$ defined in §2, we define $\mathcal{F}'_i S$ as $\{(f'_i)^j(T) : T \in S, j \geq 0\} - \{\mathbf{0}\}$. Then we have an analogue of 2.4:

Theorem 5.1. *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*
$$\mathrm{SVT}(\alpha) = \mathcal{F}'_{i_1} \dots \mathcal{F}'_{i_k}\{u_\lambda\}.$$
Here, u_λ is the SSYT with shape λ such that its r^{th} row only has r.

Proof. Prove by induction on k. The base case is when $k = 0$ and $\alpha = \lambda$. The equation becomes $\mathrm{SVT}(\alpha) = \{u_\lambda\}$, which is immediate.

The inductive step is to prove the following.

$$\mathrm{SVT}(s_i\alpha) = \mathcal{F}'_i \,\mathrm{SVT}(\alpha), \tag{5.1}$$

where $\alpha_i > \alpha_{i+1}$. We prove each side of (Equation 5.1) contains the other side.

- Take $T \in \mathrm{SVT}(\alpha)$. By Lemma 4.40, the double i-string of T is completely in $\mathrm{SVT}(s_i\alpha)$. Thus, $(f'_i)^j(T)$ is in $\mathrm{SVT}(s_i\alpha)$ if it is not $\mathbf{0}$.
- Suppose $T \in \mathrm{SVT}(s_i\alpha)$. By Lemma 4.40, the source of its double i-string is in $\mathrm{SVT}(\alpha)$. Thus, T is in the right hand side of (Equation 5.1).

□

If we slightly rephrase this theorem, we get the following statements, which correspond to axioms K1 and K2 in section 7 of [MPS20].

Corollary 5.2. *Let α be a weak composition such that $\alpha^+ = \lambda$ and $\alpha_i > 0$ for $i > n$.*

(1) *For each $T \in SVT(\alpha)$, we can obtain T by applying f'_i on u_λ.*
(2) *If $\alpha = s_{i_1} \dots s_{i_k}\lambda$ where k is minimized, then*

$$\mathrm{SVT}(\alpha) \sqcup \{\mathbf{0}\} = \{f'^{j_1}_{i_1} \dots f'^{j_k}_{i_k} u_\lambda : j_1, \dots, j_k \geq 0\}$$

The third axiom in [MPS20] corresponds to our main result. Thus, we claim our construction is an answer to [MPS20, Open Problem 7.1] in the context of abstract Kashiwara crystals.

6. Future directions

In this section, we introduce a few problems related to our main result.

6.1. **Finding a bijective proof of Theorem 3.17.** As mentioned in §1, there exist various combinatorial formulas for Lascoux polynomials. We would like to describe one of them.

A *reverse semistandard Young tableau* (RSSYT) is a filling of a Young diagram with $\mathbb{Z}_{>0}$, such that each column is strictly decreasing and each row is weakly decreasing. A *reverse set-valued tableau* (RSVT) is a filling of Young is a filling of a Young diagram with non-empty subsets of $\mathbb{Z}_{>0}$, such that no matter how we pick we number in each entry, the resulting tableau is a RSSYT. We may define $\mathrm{wt}(\cdot)$ and $\mathrm{ex}(\cdot)$ for RSVT analogously. We also define a map $L(\cdot)$ from RSVT to RSSYT. $L(\cdot)$ picks the largest number in each entry.

A *reverse key* is a RSSYT, where each number in column j is also in column $j - 1$. Clearly, reverse keys are in bijection with weak compositions. We let key^R be the map that sends a weak composition to its corresponding reverse key. Each RSSYT T is associated with a reverse key called its left key, denoted by $K_-(T)$.

There is a weight-preserving map from SSYT to RSSYT called reverse complement [LS90]. It is anti-rectification, followed by 180^o rotation. Moreover, if T is a SSYT with right key $\mathrm{key}(\alpha)$, then the left key of T's image is $\mathrm{key}^R(\alpha)$. Under this bijection, we may transform the SSYT rule for Demazure character (2.3) into a RSSYT rule:

$$\kappa_\alpha = \sum_{K_-(T) \leq \mathrm{key}^R(\alpha)} x^{\mathrm{wt}(T)}, \tag{6.1}$$

where T is a RSSYT.

This rule is generalized to Lascoux polynomials by [BSW20, SY21]:

$$\mathfrak{L}_\alpha^{(\beta)} = \sum_{K_-(L(T)) \leq \mathrm{key}^R(\alpha)} x^{\mathrm{wt}(T)} \beta^{\mathrm{ex}(T)}, \tag{6.2}$$

where T is a RSVT.

One can prove Theorem 3.17 by building an appropriate bijection between the SVT appeared in (3.1) and the RSVT in (6.2). More explicitly, we may describe the problem as follows.

Problem 1. *Find a map Φ that sends a SSYT to a RSSYT, satisfying:*

(1) *Φ preserves $\mathrm{wt}(\cdot)$ and $\mathrm{ex}(\cdot)$.*
(2) *If T is a RSVT with $K_+(T) = \mathrm{key}(\alpha)$, then $L(\Phi(T))$ has left key $\mathrm{key}^R(\alpha)$.*

6.2. **Tableau complexes.** As mentioned in §1, we can view $\mathrm{SVT}(\alpha)$ as a sub-complex of the Young tableau complex. Knutson, Miller and Yong [KMY08] showed that the Young tableau complex is homeomorphic to a ball or a sphere.

Problem 2. *Determine whether the sub-complex $\mathrm{SVT}(\alpha)$ is also homeomorphic to a ball or a sphere.*

7. Acknowledgments

The author is grateful to Mark Shimozono, Brendon Rhoades, Alexander Yong, Travis Scrimshaw and Jianping Pan for helpful conversations.

References

[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.

[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.

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

[KMY08] Allen Knutson, Ezra Miller, and Alexander Yong. Tableau complexes. *Israel Journal of Mathematics*, 163(1):317–343, 2008.

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

[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.

[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.

[SB02] Anders Skovsted Buch. A Littlewood-Richardson rule for the K-theory of Grassmannians. *Acta Mathematica*, pages 37–78, 2002.

[SY21] Mark Shimozono and Tianyi Yu. Grothendieck to Lascoux expansions. *arXiv preprint arXiv:2106.13922*, 2021.

[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.

Department of Mathematics, University of California, San Diego, La Jolla, CA, 92093, USA

Email address: tiy059@ucsd.edu