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 description for Lascoux polynomials involving right keys and set-valued tableaux. Our description generalizes the tableaux formulas of key polynomials and Grassmannian stable Grothendieck polynomials. To prove our description, we construct a new abstract Kashiwara crystal structure on set-valued tableaux. This construction answers an open problem of Monical, Pechenik and Scrimshaw.

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. Note that the β is also an indeterminant. These polynomials 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.

- Grassmannian stable Grothendieck polynomials are K-theoretic analogs of Schur polynomials: $G^{(0)}_\lambda = s_\lambda$.
- Extending this viewpoint, Lascoux polynomials may be viewed as K-theoretic analogs of key polynomials: $\mathfrak{L}^{(0)}_\lambda = \kappa_\lambda$.
- Lascoux polynomials generalize Grassmannian stable Grothendieck polynomials in a manner analogous to the generalization of Schur polynomials by key polynomials.

Their relations are summarized in the following diagram:

$$\begin{tikzcd}
\mathfrak{L}^{(\beta)}_\alpha \arrow[r, "\text{specialize}"] \arrow[d, "\beta=0"] & G^{(\beta)}_\lambda \arrow[d, "\beta=0"'] \\
\kappa_\alpha \arrow[r, "\text{specialize}"'] & s_\lambda
\end{tikzcd}$$

Here is another perspective to see how 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 expansions in both [RS95] and [BKS+08]. Shimozono and Yu [SY21] proved this conjecture.

Polynomials in the diagram above have tableaux formulas. 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] conjecture 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 3.16). 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 Theorem 3.16 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 defined operators f_i, e_i on SVT and obtain an abstract Kashiwara crystal structure. Our operators generalize the classical crystal operators on SSYT. However, our construction is not isomorphic to the crystal basis of a $U_q(\mathfrak{sl}_\mathfrak{n})-$representation. In addition, we defined operators f_i', e_i' which can be viewed as "square roots" of our f_i and e_i. Notice that Monical, Pechenik and Scrimshaw [MPS20] have already defined a crystal structure on SVT, which comes from a $U_q(\mathfrak{gl}_\mathfrak{n})-$representation. However, their crystal operators are not compatible with $K_+(\cdot)$ introduced in §3.

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 4.35, which is an analogous result of [Kas93, Proposition 3.3.5]. 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 3.16. In §4, we construct a Kashiwara crystal on SVT and prove Theorem 3.16. 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]. In §6, we extend our main result to Lascoux atoms.

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 $i+1$. Following [LS89] and [Las01], 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 has *normal* (resp. *antinormal*) shape 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 cells 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 the number in each cell of T in the column order.

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 $\mathrm{key}(\cdot)$ 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 SSYT $T^{\searrow}$ with antinormal shape such that $\mathrm{word}(T) \sim \mathrm{word}(T^{\searrow})$.

Each SSYT T with normal shape 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 SSYT T' with antinormal shape:

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

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, $T^{\searrow}_{\geq 2}, T^{\searrow}_{\geq 3}$ 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 $\mathrm{SSYT}(\alpha)$ be the set of all SSYT such that T has the same shape as $\mathrm{key}(\alpha)$ and $K_+(T) \leq \mathrm{key}(\alpha)$ 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.** First, we will introduce *Abstract Kashiwara crystals* [Kas90] [Kas91] following [BS17].

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Proof. Since $\mathcal{B}$ is seminormal, $\varepsilon_i(X_0) = 0$ and $\varphi_i(X_0) = k$. Let a be the $(j+1)^{th}$ entry of $\mathrm{wt}(X_0)$. By (K2), $\mathrm{wt}(X_0) = (\dots, a+k, a, \dots)$. Apply f_i for j times on X_0 and obtain X_j. By (K1), $\mathrm{wt}(X_j) = (\dots, a+k-j, a+j, \dots)$. Thus, $\mathrm{wt}(X_j) = s_i \mathrm{wt}(X_{k-j})$. □

Now we describe a well-known example of an abstract Kashiwara crystal. Let $\mathcal{B}(\lambda, n)$ be the set of all SSYT whose shapes are λ and entries are in $[n]$. Take $T \in \mathcal{B}(\lambda, n)$ 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 2.5. 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. We change this i into $i+1$ and get $f_i(T)$.

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. We change this $i+1$ into i and get $e_i(T)$.

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

We can use the operator f_i to compute $\mathrm{SSYT}(\alpha)$. 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 2.6 ([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.

$\mathrm{SSYT}(\alpha)$, 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 2.7. A *set-valued tableau* (SVT) is a tableau such that no matter how we pick one entry in each set, the resulting tableau is a SSYT. Let T be a SVT. Define $\mathcal{S}(T)$ to be the set of SSYT obtained by picking one number in each cell of T.

Example 2.8. The following T is a SVT:

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

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

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

The following example is not a SVT

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

since if we pick 4 in both cells of column 2, the resulting filling cannot be a SSYT.

Remark 2.9. A SSYT can be viewed as a SVT where each set is a singleton.

Definition 2.10. 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, $\mathrm{ex}(T)$ counts the number of "extra" numbers in T.

To supplement our introduction, before continuing our development, we state an application of SVT. This is a SVT rule for Grassmannian stable Grothendieck polynomials $G_\lambda^{(\beta)}$ due to Buch [SB02]. 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

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

Theorem 2.11 ([SB02])**.** *Let λ be a partition. Then*

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

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 $K_+(T)$ for SSYT T with normal shape. Then we generalize the right key to all SVT with normal shape. 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] This method is a reformulation of Willis' method [Wil13].

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 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 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 have the following way to compute a right key.

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

Proof. By definition, column j of $K_+(T_{\geq j})$ equals the last column of $T_{\geq j}^{\searrow}$. Since $T_{\geq j}^{\searrow}$ has antinormal shape, $\emptyset \star \mathrm{word}(T_{\geq j}^{\searrow})$ is the set of numbers in 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. $\square$

Example 3.5. 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

2	4	4	7
4	7	7	
7	8		
8			

which agrees with Example 2.1.

[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.

3.2. **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 3.6. 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 3.7. We start with the SVT T. The set $\mathcal{S}(T)$ has two SSYT.

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

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

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

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 3.8. 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 3.9. 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 $S_1, \ldots, S_n$ are entries of T in the column order.

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

Lemma 3.10. *Let T be a normal SVT. Column j of $T_{\max}$ consists of $\emptyset \star \text{word}(T_{\geq j})$, where $T_{\geq j}$ is obtained by removing the first $j-1$ columns of T.*

Example 3.11. Let T be the SVT in example 2.8. Then $\text{word}(T) = 567231471336$. 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,

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

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 2.8. We get

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

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 3.12. 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 3.10. It is enough to assume $j = 1$. By the definition of $T_{\max}$ and Lemma 3.4, column 1 of $T_{\max}$ consists of $\max_{P \in \mathcal{S}(T)} \emptyset \star \mathrm{word}(P)$. Thus, we need to prove

$$\max_{P \in \mathcal{S}(T)} \emptyset \star \mathrm{word}(P) = \emptyset \star \mathrm{word}(T) \tag{3.1}$$

First, we prove (3.1) 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 $1 \leq i \leq k-1$. As P ranges over $\mathcal{S}(T)$, $\mathrm{word}(P)$ ranges over $s_1 \cdots s_k$ with $s_i \in S_i$. Thus, $\emptyset \star \mathrm{word}(P)$ ranges over $\{s_1 > \cdots > s_k\}$ with $s_i \in S_i$. The left hand side of (3.1) is $\{\max(S_1) > \cdots > \max(S_k)\}$. For the right hand side, notice that $\emptyset \star \mathrm{word}(S_1) = \{\max(S_1)\}$. Since $\max(S_1) > \max(S_2)$, $\emptyset \star \mathrm{word}(S_1)\mathrm{word}(S_2) = \{\max(S_1), \max(S_2)\}$. A simple induction on i would yield

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

Since $\mathrm{word}(T) = \mathrm{word}(S_1) \cdots \mathrm{word}(S_k)$, the right hand side of (3.1) is $\{\max(S_1) > \cdots > \max(S_k)\}$. We have established (3.1) for T with one column.

Now we prove (3.1) 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

$$\mathrm{word}(T) = \mathrm{word}(T')\mathrm{word}(X) \qquad \text{and}$$
$$\{\mathrm{word}(P) : P \in \mathcal{S}(T)\} = \{\mathrm{word}(P')x : P' \in \mathcal{S}(T'), x \in X\}.$$

Our goal (3.1) becomes

$$\max_{P' \in \mathcal{S}(T'), x \in X} \emptyset \star \mathrm{word}(P')x = \emptyset \star \mathrm{word}(T')\mathrm{word}(X). \tag{3.2}$$

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

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

Use $\{a_1 < \cdots < a_k\}$ to denote $\emptyset \star \mathrm{word}(T')$. We know $k = |\emptyset \star \mathrm{word}(P')|$ 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, $k = |\emptyset \star \mathrm{word}(P)|$ 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 $\min(\emptyset \star \mathrm{word}(P')) = a_1$. Then consider the tableau $P \in \mathcal{S}(T)$ with $\mathrm{word}(P) = \mathrm{word}(P')x$. We have $\emptyset \star \mathrm{word}(P) = (\emptyset \star \mathrm{word}(P')) \star x$, 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 $\mathrm{word}(X) = \mathrm{word}(X_1) \cdots \mathrm{word}(X_k)$ 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, $\{a_1, \ldots, a_k\} \star \mathrm{word}(X) = \{\overline{a}_1 < \cdots < \overline{a}_k\}$, 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 (3.2) 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 $\emptyset \star \mathrm{word}(P')x$ 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 $\emptyset \star \mathrm{word}(P')x$ achieves $\overline{a}_i$.

Now we prove these two claims.

- Take any $P' \in \mathcal{S}(T')$ and $x \in X$. Let $\{b_1 < \cdots < b_k\} = \emptyset \star \mathrm{word}(P')$. 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 $\{b_1 < \cdots < b_k\} = \emptyset \star \mathrm{word}(P')$, 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 $\{b_1 < \cdots < b_k\} = \emptyset \star \mathrm{word}(P')$, 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 $b_i = \overline{a}_i$.

$\square$

Corollary 3.13. *$T_{\max}$ is a key .*

Proof. Let j be a positive integer. By Lemma 3.10, it remains to show

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

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

Definition 3.14. The right key of a SVT T is $K_+(T) := T_{\max}$. Let $\mathrm{SVT}(\alpha)$ be the set of all T such that $K_+(T) \leq \mathrm{key}(\alpha)$.

Remark 3.15. By the definition of $K_+(\cdot)$, we may also describe $\mathrm{SVT}(\alpha)$ as all SVT T such that $\mathcal{S}(T) \subseteq \mathrm{SSYT}(\alpha)$.

We end this section by introducing our main result:

Theorem 3.16. *Let α be a weak composition. Then*

$$\mathfrak{L}_{\alpha}^{(\beta)} = \sum_{T \in \mathrm{SVT}(\alpha)} x^{\mathrm{wt}(T)} \beta^{\mathrm{ex}(T)}. \tag{3.3}$$

Example 3.17. Let $\alpha = (1, 0, 2)$. Then $\mathrm{SVT}(\alpha)$ consists of the following:

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

$$\begin{array}{|c|c|}\hline 1 & 12 \\ \hline 2 \\ \cline{1-1}\end{array} \quad \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 23 \\ \cline{1-1}\end{array} \quad \begin{array}{|c|c|}\hline 1 & 13 \\ \hline 2 \\ \cline{1-1}\end{array} \quad \begin{array}{|c|c|}\hline 1 & 3 \\ \hline 23 \\ \cline{1-1}\end{array} \quad \begin{array}{|c|c|}\hline 1 & 13 \\ \hline 3 \\ \cline{1-1}\end{array} \quad \begin{array}{|c|c|}\hline 1 & 23 \\ \hline 2 \\ \cline{1-1}\end{array}$$

$$\begin{array}{|c|c|}\hline 1 & 123 \\ \hline 2 \\ \cline{1-1}\end{array} \quad \begin{array}{|c|c|}\hline 1 & 13 \\ \hline 23 \\ \cline{1-1}\end{array}$$

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

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

Equation (3.3) generalizes the two combinatorial rules in §2:

- If we set $\beta = 0$, then the left hand side of Equation (3.3) becomes κ_α. In the right hand side, only T with $\mathrm{ex}(T) = 0$ can survive in the sum. Clearly, $\{T \in \mathrm{SVT}(\alpha) : \mathrm{ex}(T) = 0\} = \mathrm{SSYT}(\alpha)$. Thus, our rule extends 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, our rule extends Equation (2.4).

We will prove Theorem 3.16 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 GL_n-crystal on the set of SVT with entries in $[n]$.

4.1. **Constructing an abstract Kashiwara crystal 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. Next, we would like to generalize the i-word defined on SSYT in §2.

Definition 4.1. Let $\mathcal{B}_{\text{word}}$ be the set of finite words generated by "(", ")", and ")−(" under concatenation.

Take $T \in \mathcal{B}_n$ and $i \in [n-1]$. The *i-word* of T is the following element of $\mathcal{B}_{\text{word}}$: 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 ") − (".

Example 4.2. Consider the following element from $\mathcal{B}_4$

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

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

To define an abstract Kashiwara GL_n-crystal on $\mathcal{B}_n$, we first need some definitions on $\mathcal{B}_{\text{word}}$. Take $w \in \mathcal{B}_{\text{word}}$. 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 first word)) − (()) − ()) − (() − (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, 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 "(″. For example, "(()) − (″.
- *right form*: This class does not have unpaired "(″ but starts with an unpaired ")″. For example, ") − () − ()″.
- *combined form*: This class start with an unpaired ")″ and ends with an unpaired "(″. For example, ") − () − (()) − (″.

In the previous example, 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.

We can describe each left form, right form and combined form. If a class is a left form, then it must be "(″ or "(u) − (″, where u is some word. Similarly, if a class is a right form, then it must be ")″ or ") − (u)″. If a class is a combined form, then it must be ") − (″ or ") − (u) − (″. Based on these descriptions, we define a way to change one form into another. The transformations can be described as follows.

$$\begin{array}{ccccc}) & \longleftrightarrow &) - (& \longleftrightarrow & (\,, \\ ) - (u) & \longleftrightarrow &) - (u) - (& \longleftrightarrow & (u) - (\,, \end{array}$$

where u is some word. For instance, if we transform "$)$", "$)-()-($" and "$)-(()())$" into left forms, we get "$($", "$()-($" and "$(()())-($".

Definition 4.3. Take $w \in \mathcal{B}_{\text{word}}$. If w has no right forms, then f sends it to $\mathbf{0}$. Otherwise, if there is no combined form, then f transforms the last right form into a left form. If there is a combined form, then f transforms the combined form into a left form and transforms the last right form into a combined form.

Example 4.4. We have

$$)-()\,)\,()-(\quad \xrightarrow{f} \quad)-()\,(\,()-($$

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

The e operator can be defined similarly.

Definition 4.5. Take $w \in \mathcal{B}_{\text{word}}$. If it has no left forms, then e sends it to $\mathbf{0}$. Otherwise, if there is no combined form, then e transforms the first left form into a right form. If there is a combined form, then e transforms the combined form into a right form and transforms the first left form into a combined form.

The f and e operators defined above have "square roots".

Definition 4.6. Define

$$f', e' : \mathcal{B}_{\text{word}} \sqcup \{\mathbf{0}\} \to \mathcal{B}_{\text{word}} \sqcup \{\mathbf{0}\}$$

based on the following cases:

(1) $f'(\mathbf{0}) = e'(\mathbf{0}) = \mathbf{0}$
(2) Assume $w \in \mathcal{B}_{\text{word}}$ has a combined form. The f' will transform the combined form into a left form. The e' will transform the combined form into a right form.
(3) Assume $w \in \mathcal{B}_{\text{word}}$ has no combined form. If there is a right form, f' will transform the last right form into a combined form. Otherwise, $f'(w) = \mathbf{0}$. If there is a left form, e' will transform the last left form into a combined form. Otherwise, $e'(w) = \mathbf{0}$.

Example 4.7. For example

$$)-()\,)\,()-(\quad \xrightarrow{f'} \quad)-()\,)-(\,()-(\quad \xrightarrow{f'} \quad)-()\,(\,()-($$

We have the following relations between f, e, f' and e'.

Lemma 4.8. *Take w_1, w_2 from B_{word}. Then we have:*

- *$f'(w_1) = w_2$ if and only if $e'(w_2) = w_1$.*
- *$f(w_1) = w_2$ if and only if $f'(f'(w_1)) = w_2$.*
- *$f(w_1) = w_2$ if and only if $e(w_2) = w_1$.*

Proof. Immediate from definitions. □

Remark 4.9. Assume $f'(w) \neq \mathbf{0}$. Then f' must do one of the following to w:

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

As a summary, f' decreases the number of "$($" by 1 or increases the number of "$)$" by 1. Consequently, f decreases the number of "$($" by 1 and increases the number of "$)$" by 1.

Finally, we are ready to define an abstract Kashiwara GL_n-crystal on $\mathcal{B}_n$. We start with $\varepsilon(\cdot)$ and $\varphi(\cdot)$.

Definition 4.10. Take $T \in \mathcal{B}_n$ and take $i \in [n-1]$. Let $\varepsilon_i(T)$ (resp. $\varphi_i(T)$) be the number of left forms (resp. right forms) in the i-word of T.

Now we would like to define the crystal operators f_i and e_i on $\mathcal{B}_n$. First, we define their "square roots" f_i' and e_i':

Definition 4.11. Define f_i' on $\mathcal{B}_n \cup \{\mathbf{0}\}$. First, $f_i'(\mathbf{0}) = \mathbf{0}$. Now take $T \in \mathcal{B}_n$. Apply f' on the i-word of T. Change T accordingly and obtain $f_i'(T)$. More explicitly, based on Remark 4.9, we may describe f_i' as the following:

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

Example 4.12. Consider T in Example 4.2. The operator f' would turn its 2-word "$())-())-($" into "$())-()($", by changing the last "$)-($" into "$($". Accordingly, we remove 2 from the set corresponding to this "$)-($", which is the highest entry in column 4 of T. Thus, we have

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

Now $f_2'(T)$ has 2-word "$())-()($". The operator f' would turn this word into "$())-()-(($" by changing the last "$)$" into "$)-($". Accordingly, we add 3 to the set corresponding to this "$)$", which is the highest entry in column 3 of $f_2'(T)$. Thus, we have

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

Lemma 4.13. *The operator f_i' is well-defined. That is, for any $T \in \mathcal{B}_n$, $f_i'(T)$ is a SVT or $\mathbf{0}$.*

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

- Assume f' 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 this r. The operator f_i' will put $i+1$ into S. 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' 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 4.14. Define $f_i : \mathcal{B}_n \to \mathcal{B}_n \sqcup \{\mathbf{0}\}$ as $f_i(T) = f'_i(f'_i(T))$. Equivalently, we may define $f_i(T)$ as: Apply f on the i-word of T and change T accordingly.

Example 4.15. Following Example 4.12, we have

$$T = \begin{array}{|c|c|c|c|}\hline 1 & 1 & 2 & 23 \\ \hline 2 & 23 \\ \cline{1-2} 34 \\ \cline{1-1}\end{array} \quad \xrightarrow{f_2} \quad \begin{array}{|c|c|c|c|}\hline 1 & 1 & 23 & 3 \\ \hline 2 & 23 \\ \cline{1-2} 34 \\ \cline{1-1}\end{array} = f_2(T).$$

We can define e'_i and e_i on $\mathcal{B}_n$ similarly.

Definition 4.16. Define e'_i on $\mathcal{B}_n \sqcup \{\mathbf{0}\}$. First, $e'_i(\mathbf{0}) = \mathbf{0}$. Now take $T \in \mathcal{B}_n$. Apply e' on the i-word of T. Change T accordingly and obtain $e'_i(T)$. Then $e_i(T) := e'_i(e'_i(T))$.

Similar to Lemma 4.13, we can show e'_i is also well-defined.

Remark 4.17. When we restrict f_i, e_i, φ_i and ε_i on a SSYT, we obtain the classical construction described in §2.

Lemma 4.18. *$\mathcal{B}_n$, together with $f_i, e_i, \varepsilon_i, \varphi_i$ and* wt*, is a seminormal abstract Kashiwara GL_n-crystal.*

Proof. First, we check the axioms of an abstract Kashiwara crystal.

- K1: Take $X, Y \in \mathcal{B}_n$. Clearly, $e_i(X) = Y$ if and only if $f_i(Y) = X$. Now assume this is the case. The i-word of Y has one more right form and one less left form than the i-word of X, so $\varphi_1(Y) = \varphi_1(X) + 1$ and $\varepsilon_1(Y) = \varepsilon_1(X) - 1$.

  Finally, by Remark 4.9, the i-word of Y has one more “)” and one less “(” than the i-word of X. In other words, Y has one more i and one less $i+1$ than X, so $\mathrm{wt}(Y) = \mathrm{wt}(X) + v_i - v_{i+1}$.
- K2: Take $X \in \mathcal{B}_n$. The expression $\langle \mathrm{wt}(X), (v_i, -v_{i+1}) \rangle$ is the number of i in X minus the number of $i+1$ in X. Thus, it is also the number of “)” in w minus the number of “(” in w, where w is the i-word of 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, $\langle \mathrm{wt}(X), (v_i, -v_{i+1}) \rangle$ is also the number of right forms in w minus the number of left forms in w, which is $\varphi_1(X) - \varepsilon_1(X)$.

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

$\square$

4.2. **Double i-strings.** In this subsection, we introduce and investigate double i-strings, which can be viewed as analogues of i-strings. Recall that an i-string in $\mathcal{B}_n$ 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 4.19. The following are 2-strings in $\mathcal{B}_3$.

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

$$\begin{array}{|c|c|}\hline 1 & 23 \\ \hline 2 \\ \cline{1-1}\end{array} \xrightarrow{2} \begin{array}{|c|c|}\hline 1 & 3 \\ \hline 23 \\ \cline{1-1}\end{array}$$

Now we are ready to introduce the double i-string.

Definition 4.20. Take $i \in [n-1]$. A *double i-string* is a sequence of SVT $T_0, \ldots, T_k \in \mathcal{B}_n$ satisfying:

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

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

$$\begin{tikzcd}
T_0 \arrow[r, "i"] \arrow[d, dashed, "i"] & T_2 \arrow[r, "i"] \arrow[d, dashed, "i"] & T_4 \arrow[r, "i"] \arrow[d, dashed, "i"] & \cdots \arrow[r, "i"] & T_{k-2} \arrow[r, "i"] \arrow[d, dashed, "i"] & T_k, \\
T_1 \arrow[r, "i"] \arrow[ur, dashed, "i"] & T_3 \arrow[r, "i"] \arrow[ur, dashed, "i"] & T_5 \arrow[r, "i"] & \cdots \arrow[r, "i"] & T_{k-1} \arrow[ur, dashed, "i"] &
\end{tikzcd}$$

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

Remark 4.21. 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.

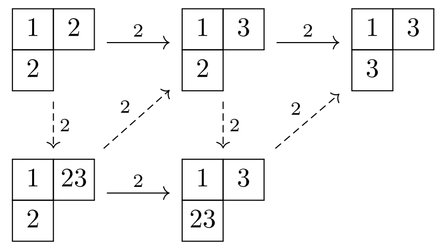
Lemma 4.22. *Let $T_0, \ldots, T_k$ be a double i-string. Then we have the following.*

(1) *k is even.*
(2) *If $k \geq 2$, then this double i-string consists of two i-strings: $T_0, T_2, \ldots, T_k$ and $T_1, T_3, \ldots, T_{k-1}$. 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) $\mathrm{wt}(T_{2j+1}) = \mathrm{wt}(T_{2j}) + v_{i+1}$.
(4) $\mathrm{wt}(T_{2j}) = \mathrm{wt}(T_{2j-1}) - v_i$.

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. Thus, we have (1) and (2). The other two statements follow from Remark 4.9.

□

Example 4.23. Notice that 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. **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 4.24. *Let $T_0, T_1, \dots, T_{2k}$ be a double i-string in $\mathcal{B}_n$. Assume $K_+(T_0) = \mathrm{key}(\alpha)$. Then $\alpha_i \geq \alpha_{i+1}$, and there are two possibilities:*

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

Example 4.25. Let $T_0, \dots, T_4$ be the double 2-string of $\mathcal{B}_3$ in Example 4.23. We have $K_+(T_0) = \mathrm{key}(\alpha)$ and $K_+(T_1) = \cdots = K_+(T_4) = \mathrm{key}(s_i\alpha)$, where $\alpha = (1, 2, 0)$

The main goal of this subsection is to prove Lemma 4.24. 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 4.26. *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 $i+1$. □

Definition 4.27. 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 4.28. *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 $S_2 \star (i+1) = S_1 \star (i+1)$.

∎

Lemma 4.29. *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:

$$(\emptyset \star u_1) \star\ \ i\ \ (i+1) = (\emptyset \star u_1) \star\ \ (i+1) \tag{4.1}$$

Now we 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 4.26, 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 (4.1) must agree with $\emptyset \star u_1$.

∎

Similarly, for e_i', we have:

Lemma 4.30. *Take $T \in B_n$ and $i \in [i-1]$. 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. 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 4.28, $S_2 \star u_2^1 = \partial_i(S_1 \star u_2^1)$. 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 4.28, $S_2 \star u_2 = \partial_i(S_1 \star u_2)$.

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 4.24.

Proof of Lemma 4.24. 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 4.26, 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 4.29, we know $K_+(T_{2j-1}) = K_+(T_{2j})$ where $j \in [k]$. By Lemma 4.30 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 4.30 again, $K_+(T_1) = \mathrm{key}(\alpha)$ or $\mathrm{key}(s_i\alpha)$, where $\alpha = K_+(T_0)$. □

Corollary 4.31. *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.*

4.4. **Proof of Theorem 3.16.** In this subsection, we derive a few lemmas and then use them to prove Theorem 3.16. 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 4.32. *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 2.4, $x^{\mathrm{wt}(T_k)} = m x_i^b x_{i+1}^a$. Thus, $k = b - a$. Finally, we have

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

□

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 4.33. *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. 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

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

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

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

Then the first equation is established by Lemma 4.32.

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

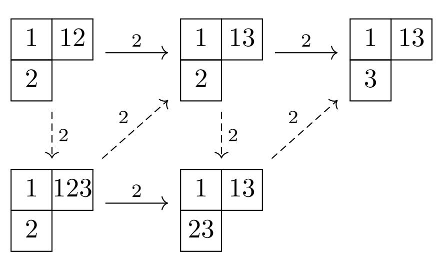
Proof. We know $K_+(T_i) \leq \mathrm{key}(\alpha)$. Since $T_1, \dots, T_{2k}$ all have the same right key, they are all in $\mathrm{SVT}(\alpha)$. By Lemma 4.24, $K_+(T_0) \leq K_+(T_i)$, so $T_0 \in \mathrm{SVT}(\alpha)$. □

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

Corollary 4.35. *Take any weak composition α. For each double i-string $S = \{T_0, \dots, T_{2k}\}$, then $\mathrm{SVT}(\alpha) \cap S$ is S, $\emptyset$, or $\{T_0\}$.*

Lemma 4.36. *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.*

Example 4.37. When $\alpha = (1, 2, 0)$, the set $\mathrm{SVT}(s_2\alpha)$ is a disjoint union of three double 2-strings. Besides the double 2-string in Example 4.23, it also contains



and

$$\begin{tikzcd} \begin{array}{|c|c|}\hline 1&1\\\hline 2\\\cline{1-1}\end{array} \arrow[r,"2"] \arrow[d, dashed, "2"] & \begin{array}{|c|c|}\hline 1&1\\\hline 3\\\cline{1-1}\end{array}. \\ \begin{array}{|c|c|}\hline 1&1\\\hline 23\\\cline{1-1}\end{array} \arrow[ur, dashed, "2"] & \end{tikzcd}$$

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

Proof. Let $T_0, \ldots, T_{2k}$ be a double i-string that intersects with $\mathrm{SVT}(s_i\alpha)$. Corollary 4.35 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 4.24 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 4.35, $\mathrm{SVT}(\alpha)$ either contains T_0 or the whole double i-string. $\square$

Now we are ready to prove our main result:

Proof of Theorem 3.16. We 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
$$\pi_i^{(\beta)}(\sum_{T\in \mathrm{SVT}(\alpha)} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)}) = \sum_{T\in \mathrm{SVT}(s_i\alpha)} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)} \tag{4.2}$$

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 4.36, elements in B are sources of double i-string and $\mathrm{SVT}(s_i\alpha) = A \sqcup \overline{B}$. Now by Lemma 4.33,

$$\pi_i^{(\beta)}(\sum_{T\in A} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)}) = \sum_{T\in A} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)},$$
$$\pi_i^{(\beta)}(\sum_{T\in B} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)}) = \sum_{T\in \overline{B}} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)}.$$

Equation 4.2 is obtained by summing up the two equations above.

$\square$

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 $\{(f'_i)^j(T) : T \in S, j \geq 0\} - \{\mathbf{0}\}$, where $S \subseteq \mathcal{B}_n$. Then we have an analogue of Theorem 2.6:

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 minimal. 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 the equation contains the other side.

- Take $T \in \mathrm{SVT}(\alpha)$. By Lemma 4.36, 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 4.36, the source of its double i-string is in $\mathrm{SVT}(\alpha)$. Thus, T is in the right hand side.

□

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. Lascoux atoms

In this section, we extend our rule to another set of polynomials called Lascoux atoms.

Definition 6.1. Following [Las01], define the operator $\overline{\pi}_i^{(\beta)}$ on $\mathbb{Z}[\beta][x_1, x_2, \dots]$ by

$$\overline{\pi}_i^{(\beta)}(f) := \pi_i^{(\beta)}(f) - f.$$

Definition 6.2. Let α be a weak composition. Similar to Lascoux polynomials, a *Lascoux atom* $\overline{\mathfrak{L}}_\alpha^{(\beta)}$ is defined as

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

The relationship between Lascoux polynomials and Lascoux atoms can be described as follows.

Definition 6.3. Let α be a weak composition. Following [Mon16], define $w(\alpha)$ as the shortest permutation such that $\alpha = w(\alpha)\alpha^+$.

Lemma 6.4 ([Mon16, Theorem 5.1])**.** *Let α be a weak composition. Then*

$$\mathfrak{L}_\alpha^{(\beta)} = \sum_\gamma \overline{\mathfrak{L}}_\gamma^{(\beta)},$$

where the sum is over all weak composition γ such that $\gamma^+ = \alpha^+$ and $w(\gamma) \leq w(\alpha)$ in Bruhat order.

It is well-known that the condition on γ is the previous lemma can be phrased as a condition on $\mathrm{key}(\gamma)$.

Lemma 6.5 ([LS90, Equation (2.13)])**.** *Let α and γ be weak compositions such that $\gamma^+ = \alpha^+$. The following are equivalent.*

- *$w(\gamma) \leq w(\alpha)$ in Bruhat order.*
- *$\mathrm{key}(\gamma) \leq \mathrm{key}(\alpha)$ entry-wise.*

Finally, we are ready to extend our rule to Lascoux atoms.

Definition 6.6. Let $\overline{\mathrm{SVT}}(\alpha)$ be the set of all SVT T such that $K_+(T) = \mathrm{key}(\alpha)$.

Remark 6.7. By Lemma 6.5, we have

$$\mathrm{SVT}(\alpha) = \bigsqcup_\gamma \overline{\mathrm{SVT}}(\gamma),$$

where γ is any weak composition such that $\gamma^+ = \alpha^+$ and $w(\gamma) \leq w(\alpha)$.

Corollary 6.8. *We have*

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

Proof. Prove by induction on the Bruhat order. If α is a partition, then our result is immediate.

Now assume this rule holds for all γ such that $\gamma^+ = \alpha^+$ and $w(\gamma) < w(\alpha)$. By Lemma 6.4, we have

$$\overline{\mathfrak{L}}_\alpha^{(\beta)} = \mathfrak{L}_\alpha^{(\beta)} - \sum_\gamma \overline{\mathfrak{L}}_\gamma^{(\beta)},$$

where the sum is over all $\gamma \neq \alpha$ such that $\gamma^+ = \alpha^+$ and $w(\gamma) \leq w(\alpha)$. By our main result and the inductive hypothesis, we have

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

Then the inductive step is finished by the remark above. □

7. Future directions

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

7.1. **Finding a bijective proof of Theorem 3.16.** 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:

$$\kappa_\alpha = \sum_{K_-(T)\leq \mathrm{key}^R(\alpha)} x^{\mathrm{wt}(T)}, \tag{7.1}$$

where T is a RSSYT.

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

$$\mathfrak{L}_\alpha^{(\beta)} = \sum_{K_-(L(T))\leq \mathrm{key}^R(\alpha)} x^{\mathrm{wt}(T)}\beta^{\mathrm{ex}(T)}, \tag{7.2}$$

where T is a RSVT.

One can prove Theorem 3.16 by building an appropriate bijection between the SVT appeared in (3.3) and the RSVT in (7.2). 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 $\mathrm{key}^R(\alpha)$.*

7.2. **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.*

8. Acknowledgments

We are especially grateful to Travis Scrimshaw for carefully reading an earlier version of this paper and giving many useful comments. We also thank two anonymous referees for thorough reports. We are grateful to Jianping Pan, Brendon Rhoades, Mark Shimozono, and Alexander Yong 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.

[Kas90] Masaki Kashiwara. Crystalizing theq-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.

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

[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.

[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