1 2 9 0

UNIVERSIDADE D
COIMBRA

João Miguel Magalhães Santos

Symplectic keys and Demazure atoms in type C

Tese no âmbito do Programa Interuniversitário de Doutoramento em Matemática, orientada pela Professora Doutora Olga Maria da Silva Azenhas e apresentada ao Departamento de Matemática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra.

Dezembro de 2021

Symplectic keys and Demazure atoms in type C

João Miguel Magalhães Santos

UC|UP Joint PhD Program in Mathematics

Programa Interuniversitário de Doutoramento em Matemática

PhD Thesis | Tese de Doutoramento

May 2022

Acknowledgements

I would like to thank the Departments of Mathematics of both University of Porto and University of Coimbra, for providing me a workplace. I also want to thank all the support from the Centre for Mathematics of the University of Coimbra–UIDB/00324/2020, funded by the Portuguese Government through FCT/MCTES, and all the support from FCT, through the grant PD/BD/142954/2018, under POCH funds, co-financed by the European Social Fund and Portuguese National Funds from MCTES.

I want to thank my advisor Olga Azenhas, for all the guidance, and patience, during the last three years. I also want to thank C. Lenart, for some remarks about the alcove path model, P. Alexandersson, C. Lecouvey and T. Scrimshaw, the latter three for interesting remarks on the generalization of our work to other combinatorial models or Cartan types.

Finally, I want thank my family and friends for all the emotional support. In particular, I want to thank Beatriz, Bina, Carlos, Joana, Palmira and Tone.

Abstract

The type C Kashiwara-Nakashima tableaux, a variation of De Concini tableaux, provide a combinatorial model for crystals associated to finite-dimensional irreducible representations of the symplectic Lie algebra. Some of these tableaux, called key tableaux, yield a tableau criterion for the Bruhat order on the hyperoctahedral group, the type C Weyl group, and a tableau criterion for the Bruhat order induced on the left cosets defined by parabolic subgroups of the hyperoctahedral group. In the type A crystal of semistandard Young tableaux, using the *jeu de taquin*, Lascoux-Schützenberger presented an algorithm to compute type A right and left key maps, that return key tableaux, for semistandard Young tableaux. Using the Sheats-Lecouvey symplectic *jeu de taquin*, we adapt Lascoux-Schutzenberger's algorithm in order to be able to compute right and left keys for type C Kashiwara-Nakashima tableaux. In fact, we can compute symplectic keys, right or left, without the use of the *jeu de taquin* and, motivated by Willis' direct way of computing right and left keys of semistandard Young tableaux, we also give a way of computing symplectic, right or left, keys without the use of *jeu de taquin*. In type C_n, the symplectic right and left key maps give a description of Demazure atoms and opposite Demazure atoms, respectively, and consequently of Demazure and opposite Demazure characters. The symplectic right and left key maps, and consequently Demazure atoms and opposite Demazure atoms, are related through the Lusztig involution. A type C Schützenberger evacuation is defined to realize that involution.

Resumo

Os Kashiwara-Nakashima tableaux do tipo C, uma variante dos tableaux de De Concini, são um modelo combinatório para os cristais associados a representações irredutíveis de dimensão finita de álgebras de Lie simplécticas. Alguns destes tableaux, chamados de key tableaux, formam critério para a ordem de Bruhat dos elementos do grupo hiperoctaedral, o grupo de Weyl do tipo C, e formam também um critério para a Bruhat order induzida nas suas classes laterais esquerdas relativamente aos subgrupos parabólicos. No cristal do tipo A formado pelos semistandard Young tableaux, utilizando o *jeu de taquin*, Lascoux e Schützenberger apresentaram um algoritmo para calcular, no tipo A, duas funções que dado um semistandard Young tableau devolvem um key tableau, right key e left key. Utilizando o *jeu de taquin* simpléctico de Sheats e Lecouvey, nós adaptamos o algoritmo do Lascoux e do Schutzenberger para poder calcular right e left keys de Kashiwara-Nakashima tableaux do tipo C. Na realidade, conseguimos calcular keys de tableaux do tipo C, right ou left, sem utilizar o *jeu de taquin* e, motivados pela maneira directa de Willis' de calcular right e left keys para semistandard Young tableaux, apresentamos uma maneira de calcular keys simplécticas, right ou left, que não utiliza o *jeu de taquin*. As funções para calcular para calcular keys simplécticas, right ou left, servem como uma descrição dos Demazure atoms e dos opposite Demazure atoms, respectivamente, e consequentemente também descrevem os carácteres de Demazure e as suas versões opposite. As funções que calculam keys simplécticas, right ou left, e consequentemente os Demazure atoms e os opposite Demazure atoms, estão relacionadas pela Lusztig involution. Definimos ainda uma Schützenberger evacuation no tipo C para realizar a Lusztig involution.

Table of contents

Chapter 1

Introduction

Symplectic tableaux are a combinatorial tool to study finite-dimensional representations of the symplectic Lie algebra $sp(2n,C)$ and its Weyl group B_n, the *hyperoctahedral group* (or signed symmetric group). The irreducible symplectic character, or symplectic Schur function, indexed by a partition λ, can be seen as a sum on symplectic tableaux of shape λ. King has proved this using a family of tableaux [21], nowadays known as King tableaux, and later De Concini found another family of symplectic tableaux [11], known as De Concini tableaux. Although quite distinct, Sheats created a weight and shape preserving bijection between both families of tableaux [39]. Here, we will work with another family of tableaux, *Kashiwara-Nakashima tableaux* [20] (for short, KN tableaux), which is a small variation of the symplectic tableaux defined by De Concini. These KN tableaux are endowed with a type C_n crystal structure that naturally contains the type A_{n-1} crystal of semistandard Young tableaux (SSYT's). In fact, the SSYT's are a particular case of KN tableaux. These KN tableaux can be seen as a folding, with some additional restrictions, of the SSYT's in A_{2n-1}, to include negative entries. The type C_n Knuth relations, or plactic relations, which include the type A_{n-1} Knuth relations, allow us to define a type C_n plactic monoid, studied by Lecouvey in [25, 26], compatible with an insertion algorithm, known as Baker-Lecouvey insertion [5, 25], and with sliding algorithms, known as symplectic *jeu de taquin* (SJDT), due to Sheats [39] and further developed by Lecouvey in [25]. Lecouvey have interpreted plactic relations in terms of crystal isomorphisms.

The type C_n Demazure characters κ_v are indexed by vectors $v \in \mathbb{Z}^n$ in $B_n\lambda$, the B_n-orbit of the partition λ, and can be seen as "partial" characters. Kashiwara [18] and Littelmann [29] have shown that they can be obtained by summing monomial weights over certain subsets $\mathfrak{B}_v$, called Demazure crystals, of the crystal $\mathfrak{B}^\lambda$, with highest weight λ. The crystal $\mathfrak{B}^\lambda$ can be partitioned into Demazure crystal atoms, $\widehat{\mathfrak{B}}_u$, where $u \in B_n\lambda$, in such a way that, for all $v \in B_n\lambda$, the Demazure crystal $\mathfrak{B}_v$ is a union of Demazure crystal atoms $\widehat{\mathfrak{B}}_u$ over the Bruhat interval $\lambda \leq u \leq v$.

Lascoux and Schützenberger, in [24], were the first ones to introduce the concept of Demazure crystal atoms, originally called standard bases, in type A_{n-1}. They proved that each Demazure crystal atom contains exactly one key tableau, a tableau whose columns form a nested set. Using the *jeu de taquin*, Lascoux and Schützenberger also defined a map, called *right key map*, that given a tableau returns the key tableau that identifies the atom that contains the given tableau. There is a dual definition of these Demazure atoms and Demazure crystal atoms, called opposite Demazure atoms and opposite Demazure crystal atoms. As expected, each opposite Demazure crystal atoms also contains exactly one key tableau, and the map that, given a tableau returns the key tableau that identifies the opposite Demazure atom that contains it, is called the *left key map*. Lascoux-Schützenberger method of computing right keys via *jeu de taquin* can easily be adapted to compute left keys. More recently, Willis, in [42], found a way of computing these right and left keys without the use of the *jeu de taquin*. There are other methods and models where type A_{n-1} keys are computed, such as the alcove path model [28], semi skyline augmented fillings [31], and coloured vertex models [7], for instance.

In type C_n, in a presentation by Azenhas in The 69th Séminaire Lotharingien de Combinatoire [4], Azenhas and Mamede identified the KN tableaux with nested columns and without symmetric entries as type C_n key tableaux, and raised questions about the existence of a right key map that does the same job as Lascoux-Schützenberger's right key map, and consequently, would provide a description for the Demazure crystal that does not require to build the crystal $\mathfrak{B}^\lambda$. This was the motivation for our main theorem, Theorem 5.2.6, which provides a description of a Demazure crystal atom in type C using the right key map defined in Theorem 5.1.5, via SJDT. During our work, we were informed by Jacon and Lecouvey that they have found a way to compute key maps of KN tableaux [17]. Although their approach is different from ours, their algorithm to compute the key maps is effectively the same as ours. Also in [17], Jacon and Lecouvey also suggested that Willis' direct way [42] of computing right and left should have a generalization for type C_n KN tableaux. So, motivated by Willis' direct way of computing keys without the use of *jeu de taquin*, we provide algorithms, in Theorem 5.3.3 and in Theorem 5.3.9, of computing symplectic left and right keys that do not require the SJDT. Finally, we relate left and right keys through the Lusztig involution and realize it by a type C_n Schützenberger evacuation. There are also keys computed in the type C_n alcove path model [27, 28], which is a crystal isomorphic to the type C_n model of KN tableaux, and in the coloured five vertex model [8], whose coincidence with type C_n model of KN tableaux is conjectured.

This thesis is organized as follows:

- In Chapter 2 we introduce KN tableaux, as well as type C_n plactic monoid, together with the Baker-Lecouvey insertion, the Sheats-Lecouvey symplectic *jeu de taquin* and the type C_n Robinson-Schensted correspondence. We finish this chapter with type C_n crystals and their relation with the plactic monoid in terms of crystal isomorphisms.

- In Chapter 3 we recall the type C_n Weyl group, the hyperoctahedral group, its Bruhat order, and give a tableau criterion for the Bruhat order of this group, and a tableau criterion for the Bruhat order induced on the left cosets defined by the parabolic subgroups of the hyperoctahedral group. This tableau criterion uses only KN key tableaux.

- In Chapter 4 we define the Demazure crystal and the opposite Demazure crystal. Then, we embed every KN tableau in a type A cocrystal, isomorphic to the crystal of SSYT's with conjugated shape. This cocrystal is motivated by Lascoux' double crystal graph in type A [22], and by Heo-Kwon work in [16], where Schützenberger *jeu de taquin* slides are used as crystal operators for $\mathfrak{sl}_2$. This serves as an appetizer for the next chapter.

- In Chapter 5, we compute symplectic right and left key maps. First, we do this computation via SJDT, mimicking Lascoux-Schützenberger approach in type A_{n-1}, and prove, in our main theorem, that these maps describe Demazure crystal atoms and opposite Demazure crystal atoms. Later in this chapter, motivated by Willis' algorithms for SSYT's [42], we compute right and left keys of a type C_n KN tableau without using the *jeu de taquin*.

- In Chapter 6 we introduce the type C_n Lusztig involution [30] and we adapt the type A_{n-1} Schützenberger evacuation [38] for SSYT's to type C_n KN tableaux. The Lusztig involution relates the right and left key maps, and consequently, it relates Demazure crystals and opposite Demazure crystals.

- In our final chapter, Chapter 7, we discuss some unfinished or undone work related to symplectic key tableaux, focusing mainly on a combinatorial approach for the type C_n Fu-Lascoux non-symmetric Cauchy kernel. The two last sections address questions on the generalizations of our results to other combinatorial models.

Related to this thesis we published one paper, [36], and one preprint, [35], submitted to a journal. Also, for each one of these publications, an extended abstract was accepted in the proceedings of The 32nd and The 33rd Conference on Formal Power Series and Algebraic Combinatorics, [34] and [37], respectively.

Chapter 2

Type C_n Kashiwara-Nakashima tableaux, symplectic plactic monoid and type C_n crystal graphs

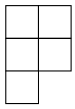
In this chapter we present type C_n Kashiwara-Nakashima tableaux, as well as the Baker-Lecouvey insertion, Lecouvey-Sheats symplectic *jeu de taquin*, type C_n Robinson-Schensted-Knuth correspondence and the type C_n plactic monoid. We finish this chapter with Kashiwara crystals and coplactic equivalence, and their relation with the plactic monoid. Our main references for this chapter are [25], [5] and [9].

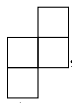
2.1 Kashiwara-Nakashima tableaux

The symplectic tableaux studied here were introduced by Kashiwara and Nakashima to label the vertices of the type C_n crystal graphs [20].

Fix $n \in \mathbb{N}_{>0}$ and define the sets $[n] = \{1,\dots,n\}$ and $[\pm n] = \{1,\dots,n,\overline{n},\dots,\overline{1}\}$ where $\overline{i}$ is just another way of writing $-i$, hence $\overline{\overline{i}} = i$. In the second set we will consider the following order of its elements: $1 < \cdots < n < \overline{n} < \cdots < \overline{1}$ instead of the usual order.

A vector $\lambda = (\lambda_1,\dots,\lambda_n) \in \mathbb{Z}^n$ is a partition of $|\lambda| = \sum_{i=1}^{n} \lambda_i$ if $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$. The *Young diagram* of shape λ, in English notation, is an array of boxes (or cells), left justified, in which the i-th row, from top to bottom, has λ_i boxes. We identify a partition with its Young diagram.

For example, the Young diagram of shape $\lambda = (2,2,1)$ is . We define $\Delta^n = (n, n-1, \dots, 1)$ to be the staircase partition in $\mathbb{Z}^n$.

Given μ and ν two partitions with $\nu \leq \mu$ entrywise, we write $\nu \subseteq \mu$. The Young diagram of shape μ/ν is obtained after removing the boxes of the Young diagram of ν from the Young diagram of μ. For example, the Young diagram of shape $\mu/\nu = (2,2,1)/(1,0,0)$ is ,.

Let $\nu \subseteq \mu$ be two partitions and A a completely ordered alphabet. A *semistandard Young tableau* (SSYT) of skew shape μ/ν, on the alphabet A, is a filling of the diagram μ/ν with letters from A, such that the entries are strictly increasing, from top to bottom, in each column and weakly increasing, from left to right, in each row. When $|\nu| = 0$ then we obtain a semistandard Young tableau of straight shape μ. Denote by $\mathcal{SSYT}(\mu/\nu, A)$ the set of all skew SSYT's T of shape μ/ν, with entries in A. In particular, when $|\nu| = 0$ we write $\mathcal{SSYT}(\mu, A)$ and when $A = [n]$ we write $\mathcal{SSYT}(\mu/\nu, n)$.

When considering tableaux with entries in $[\pm n]$, it is usual to have some extra conditions besides being semistandard. We will use a family of tableaux known as *Kashiwara-Nakashima* tableaux. From now on we consider tableaux on the alphabet $[\pm n]$.

A *column* is a strictly increasing sequence of numbers (or letters) in $[\pm n]$ and it is usually displayed vertically. The height of a column is the number of letters in it. A column is said to be *admissible* if the following *one column condition* (1CC) holds for that column:

Definition 2.1.1 (1CC)**.** *Let C be a column. The 1CC holds for C if for all pairs i and $\bar{i}$ in C, where i is in the a-th row counting from the top of the column, and $\bar{i}$ in the b-th row counting from the bottom, we have $a+b \le i$. Equivalently, for all pairs i and $\bar{i}$ in C, the number $N(i)$ of letters x in C such that $x \le i$ or $x \ge \bar{i}$ satisfies $N(i) \le i$.*

If a column C satisfies the $1CC$ then C has at most n letters. If $1CC$ does not hold for C we say that C *breaks the* $1CC$ *at* z, where z is the minimal positive integer such that z and $\bar{z}$ exist in C and there are more than z numbers in C with absolute value less or equal than z.

Example 2.1.2. *The column* $\begin{array}{|c|}\hline 1\\\hline 2\\\hline \bar{1}\\\hline\end{array}$ *breaks the* $1CC$ *at* 1*, and* $\begin{array}{|c|}\hline 2\\\hline 3\\\hline \bar{3}\\\hline\end{array}$ *is an admissible column.*

The following definition states conditions to when C can be *split*:

Definition 2.1.3. *Let C be a column and let $I = \{z_1 > \cdots > z_r\}$ be the set of unbarred letters z such that the pair $(z, \bar{z})$ occurs in C. The column C can be split when there exists a set of r unbarred letters $J = \{t_1 > \cdots > t_r\} \subseteq [n]$ such that:*

1. *t_1 is the greatest letter of $[n]$ satisfying $t_1 < z_1$, $t_1 \notin C$, and $\overline{t_1} \notin C$,*
2. *for $i = 2, \ldots, r$, we have that t_i is the greatest letter of $[n]$ satisfying $t_i < \min(t_{i-1}, z_i)$, $t_i \notin C$, and $\overline{t_i} \notin C$.*

The $1CC$ holds for a column C (or C is admissible) if and only if C can be split [39, Lemma 3.1]. If C can be split then we define *right column* of C, rC, and the *left column* of C, ℓC, as follows:

1. rC is the column obtained by changing in C, $\overline{z_i}$ into $\overline{t_i}$ for each letter $z_i \in I$ and by reordering if necessary,
2. ℓC is the column obtained by changing in C, z_i into t_i for each letter $z_i \in I$ and by reordering if necessary.

If C is admissible then $\ell C \le C \le rC$ by entrywise comparison, where ℓC has the same barred part as C and rC the same unbarred part. If C does not have symmetric entries, then C is admissible and $\ell C = C = rC$. In the next definition we give conditions for a column C to be *coadmissible*.

Definition 2.1.4. *We say that a column C is coadmissible if for every pair i and $\bar{i}$ on C, where i is on the a-th row counting from the top of the column, and $\bar{i}$ on the b-th row counting from the top, then $b-a \leq n-i$. Equivalently, for every pair i and $\bar{i}$ on C, the number $N^*(i)$ of letters x in C such that $i \leq x \leq \bar{i}$ satisfies $N^*(i) \leq n-i+1$.*

Unlike in Definition 2.1.1, in the last definition b is counted from the top of the column.

Definition 2.1.5. *Let C be a column and let $I = \{z_1 > \cdots > z_r\}$ be the set of unbarred letters z such that the pair $(z,\bar{z})$ occurs in C. The column C is coadmissible if and only if there exists a set of r unbarred letters $H = \{h_1 > \cdots > h_r\} \subseteq [n]$ such that:*

1. *h_r is the smallest letter of $[n]$ satisfying $h_r > z_r$, $h_r \notin C$, and $\overline{h_r} \notin C$,*
2. *for $i = r-1,\ldots,1$, we have that h_i is the smallest letter of $[n]$ satisfying $h_i > \max(h_{i+1}, z_i)$, $h_i \notin C$, and $\overline{h_i} \notin C$.*

Given an admissible column C, consider the map

$$\Phi : C \mapsto C^*$$

that sends C to the column C^* of the same size in which the unbarred entries are taken from ℓC and the barred entries are taken from rC.

Lemma 2.1.6. *Let C be an admissible column on the alphabet $[\pm n]$, and I and J the sets in Definition 2.1.3. The entries x (barred or unbarred) of $\Phi(C)$ are such that*

1. *$x \in \Phi(C)$ and $\bar{x} \notin \Phi(C)$ if and only if $x \in C$ and $\bar{x} \notin C$.*
2. *$x, \bar{x} \in \Phi(C)$ if and only if $x \in J$ or $\bar{x} \in J$.*

Equivalently, the set of entries in $\Phi(C)$ is $(J \cup \bar{J} \cup C) \setminus (I \cup \bar{I})$.

Henceforth, $\Phi(C) = C$ if and only if $I = \emptyset$ (hence $J = \emptyset$), that is, C does not have symmetric entries.

The *column* $\Phi(C)$ is a coadmissible column and the algorithm to form $\Phi(C)$ from C is reversible [25, Section 2.2]. In particular, every column on the alphabet $[n]$ is simultaneously admissible and coadmissible. The map Φ is a bijection between admissible and coadmissible columns of the same height on the alphabet $[\pm n]$.

Example 2.1.7. *Let $C = \begin{array}{|c|}\hline 2 \\ \hline 4 \\ \hline \bar{2} \\ \hline \end{array}$ be an admissible column, so it can be split. Then $\ell C = \begin{array}{|c|}\hline 1 \\ \hline 4 \\ \hline \bar{2} \\ \hline \end{array}$ and $rC = \begin{array}{|c|}\hline 2 \\ \hline 4 \\ \hline \bar{1} \\ \hline \end{array}$. So $\Phi(C) = \begin{array}{|c|}\hline 1 \\ \hline 4 \\ \hline \bar{1} \\ \hline \end{array}$ is coadmissible. C is also coadmissible and $\Phi^{-1}(C) = \begin{array}{|c|}\hline 3 \\ \hline 4 \\ \hline \bar{3} \\ \hline \end{array}$.*

Let T be a skew tableau with all of its columns admissible. The *split form* of a skew tableau T, $spl(T)$, is the skew tableau obtained after replacing each column C of T by the two columns $\ell C\, rC$. The tableau $spl(T)$ has double the amount of columns of T.

A semistandard skew tableau T is a *Kashiwara-Nakashima (KN) skew tableau* if its split form is a semistandard skew tableau in type A_{2n-1} (because it uses $2n$ letters). We define $\mathcal{KN}(\mu/\nu,n)$ to be the set of all KN tableaux of shape μ/ν in the alphabet $[\pm n]$. When $\nu=0$, we obtain $\mathcal{KN}(\mu,n)$. The weight of T is a vector whose i-th entry is the number of i's minus the number of $\bar{i}$.

If T is a skew tableau, the *column reading* of T, $cr(T)$, is the word read in T in the Japanese way, column reading top to bottom and right to left. The *length* of a word w is the total number of letters in w. The weight of a w in the alphabet $[\pm n]$ is the vector $\mathrm{wt}\, w=(t_1-t_{\bar{1}}, t_2-t_{\bar{2}},\dots,t_n-t_{\bar{n}})\in\mathbb{Z}^n$, where t_α is the number of α's in w, with $\alpha\in[\pm n]$. Note that the weight of a tableau and of its column reading coincide.

Example 2.1.8. *The split of the tableau* $T=\begin{array}{|c|c|}\hline 2&2\\\hline 3&3\\\hline \bar{3}\\\cline{1-1}\end{array}$ *is the tableau* $spl(T)=\begin{array}{|c|c|c|c|}\hline 1&2&2&2\\\hline 2&3&3&3\\\hline \bar{3}&\bar{1}\\\cline{1-2}\end{array}$. *Hence* $T\in\mathcal{KN}((2,2,1),3)$, *weight* $wtT=(0,2,1)$ *and* $cr(T)=2323\bar{1}$.

If T is a tableau without symmetric entries in any of its columns, i.e., for all $i\in[n]$ and for all columns C in T, i and $\bar{i}$ do not appear simultaneously in the entries of C, then in order to check whether T is a KN tableau it is enough to check whether T is semistandard in the alphabet $[\pm n]$. In particular $\mathcal{SSYT}(\mu/\nu,n)\subseteq\mathcal{KN}(\mu/\nu,n)$.

2.2 Symplectic *jeu de taquin*

Lecouvey-Sheats symplectic *jeu de taquin* (SJDT) [25, 39] is a procedure on KN skew tableaux, compatible with *Knuth equivalence* (or plactic equivalence on words over the alphabet $[\pm n]$) [25], that allows us to change the shape of a tableau and to rectify it. To explain how the SJDT behaves, we need to look how it works on 2-column KN skew tableaux C_1C_2. A skew tableau is *punctured* if one of its box contains the symbol $*$ called the *puncture*. A punctured column is admissible if the column is admissible when ignoring the puncture. A punctured skew tableau is admissible if its columns are admissible and the rows of its split form are weakly increasing, ignoring the puncture. Let T be a punctured skew tableau with two columns C_1 and C_2 with the puncture in C_1. In that case, the puncture splits into two punctures in $spl(T)$, and ignoring the punctures, $spl(T)$ must be semistandard. Let α be the entry under the puncture of rC_1, and β the entry to the right of the puncture of rC_1.

$$spl(T)=\ell C_1 rC_1 \ell C_2 rC_2=\begin{array}{|c|c|c|c|}\hline \cdots&\cdots&\cdots&\cdots\\\hline *&*&\beta&\cdots\\\hline \cdots&\alpha&\cdots&\cdots\\\hline \cdots&\cdots\\\cline{1-2}\end{array},$$

where α or β may not exist. The elementary steps of SJDT are the following:

A. If $\alpha\leq\beta$ or β does not exist, then the puncture of T will change its position with the cell beneath it. This is a vertical slide.

B. If the slide is not vertical, then it is horizontal. So we have $\alpha>\beta$ or α does not exist. Let C_1' and C_2' be the columns obtained after the slide. We have two subcases, depending on the sign of β:

1. If β is barred, we are moving a barred letter, β, from ℓC_2 to the punctured box of rC_1, and the puncture will occupy β's place in ℓC_2. Note that ℓC_2 has the same barred part as C_2 and that rC_1 has the same barred part as $\Phi(C_1)$. Looking at T, we will have an horizontal slide of the puncture, getting $C_2' = C_2 \setminus \{\beta\} \sqcup \{*\}$ and $C_1' = \Phi^{-1}(\Phi(C_1) \setminus \{*\} \sqcup \{\beta\})$. In a sense, β went from C_2 to $\Phi(C_1)$.

2. If β is unbarred, we have a similar case, but this time β will go from $\Phi(C_2)$ to C_1, hence $C_1' = C_1 \setminus \{*\} \cup \{\beta\}$ and $C_2' = \Phi^{-1}(\Phi(C_2) \setminus \{\beta\} \sqcup \{*\})$. Although in this case it may happen that C_1' is no longer admissible. In this situation, if the 1CC breaks at i, we erase both i and $\bar{i}$ from the column and remove a cell from the bottom and from the top column, and place all the remaining cells orderly with respect to their entries.

Applying successively elementary SJDT slides, eventually, the puncture will be a cell such that α and β do not exist. In this case we redefine the shape to not include this cell and the *jeu de taquin* ends.

Given an admissible tableau T of shape μ/ν, a box of the diagram of shape ν such that boxes under it and to the right are not in that shape is called an inner corner of μ/ν. An outside corner is a box of μ such that boxes under it and to the right are not in the shape μ. The rectification of T consists in playing the SJDT until we get a tableau of shape λ, for some partition λ. More precisely, apply successively elementary SJDT steps to T until each cell of ν becomes an outside corner. At the end, we obtain a KN tableau for some shape λ. The rectification is independent of the order in which the inner corners of ν are filled [25, Corollary 6.3.9].

Example 2.2.1. *Consider the KN skew tableau* $T = \begin{array}{|c|c|} \hline & 2 \\ \hline 1 & 3 \\ \hline 2 & \bar{1} \\ \hline \end{array}$. *Let* C_1 *and* C_2 *be the first and second columns of* T. *To rectify* T *via SJDT, one creates a puncture in the inner corner of* T *and, by splitting, one obtains* $\begin{array}{|c|c|c|c|} \hline * & * & 2 & 2 \\ \hline 1 & 1 & 3 & 3 \\ \hline 2 & 2 & \bar{1} & \bar{1} \\ \hline \end{array}$. *So, the first two slides are vertical, obtaining* $\begin{array}{|c|c|c|c|} \hline 1 & 1 & 2 & 2 \\ \hline 2 & 2 & 3 & 3 \\ \hline * & * & \bar{1} & \bar{1} \\ \hline \end{array}$. *Finally, we do an horizontal slide, of type* **B.1**, *in which we take* $\bar{1}$ *from* C_2, *and add it to the coadmissible column* $\Phi(C_1)$. *That is,* $C_2' = (C_2 \cup \{*\}) \setminus \bar{1}$ *and* $C_1' = \Phi^{-1}((\Phi(C_1) \setminus \{*\}) \cup \bar{1})$, *obtaining the tableau* $\begin{array}{|c|c|} \hline 2 & 2 \\ \hline 3 & 3 \\ \hline \bar{3} \\ \cline{1-1} \end{array}$.

Let T be a skew tableau of shape μ/ν. Consider a punctured box that can be added to μ, so that $\mu \cup \{*\}$ is a valid shape. The SJDT is reversible, meaning that we can move $*$, the empty cell outside of μ, to the inner shape ν of the skew tableau T, simultaneously increasing both the inner and outer shapes of T by one cell. The slides work similarly to the previous case: the vertical slide means that an empty cell is going up and an horizontal slide means that an entry goes from $\Phi(C_1)$ to C_2 or from C_1 to $\Phi(C_2)$, depending on whether the slid entry is barred or not, respectively. We will also call the *reverse jeu de taquin* as SJDT. In the next sections we will be mostly dealing with the *reverse jeu de taquin*. Consider the following examples, each containing a tableau and a punctured box that will be slid to its inner shape: $\begin{array}{|c|c|c|} \hline & & * \\ \hline 1 & \bar{1} \\ \cline{1-2} 2 \\ \cline{1-1} \end{array} \mapsto \begin{array}{|c|c|c|} \hline & & \\ \hline 1 & \bar{1} \\ \cline{1-2} 2 \\ \cline{1-1} \end{array}$; $\begin{array}{|c|c|} \hline & \\ \hline 1 & \bar{1} \\ \hline 2 & * \\ \hline \end{array} \mapsto \begin{array}{|c|c|} \hline & \\ \hline & 2 \\ \hline 2 & \bar{2} \\ \hline \end{array}$.

If a tableau with columns C_1 and C_2 does not have symmetric entries then the SJDT applied to C_1C_2 coincides with the *jeu de taquin* known for SSYT's. In sections 5.3.1 and 5.3.3, we use SJDT to swap lengths of consecutive columns in a skew tableau, to obtain skew tableaux Knuth related to a straight tableau, which is minimal for the number of cells within its Knuth class. Recall that in the elementary step $B.2$ it is possible to lose cells. If we do a reverse elementary step $B.2$ that results in having two more cells in the skew tableau, we would have to start by adding two symmetric entries to an admissible column, making it non admissible [25, Lemma 3.2.3], and then slide an unbarred cell to the column to its right. For instance, consider the following reverse elementary step $B.2$ ($\equiv$ denotes type C_n Knuth equivalence [25, Definition 3.2.1]):

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

.

The first and last skew tableaux are Knuth equivalent, but the middle tableau is not a KN skew tableau. The three semistandard tableaux are Knuth equivalent column words, via the contractor/dilator Knuth relation [25, Definition 3.2.1].

Hence, a reverse elementary step $B.2$ that results in having more cells in the skew tableau has to be forced, since we have to start by forcing the existence of a non admissible column. This means that if we start with a minimal skew tableau, that is, a skew-tableau with the number of cells of its rectification, we can play SJDT, or its reverse, without ever incur in a loss/gain of boxes.

2.3 Baker-Lecouvey insertion

Let's start by recalling the column insertion for SSYT's. Let $T \in \mathcal{SSYT}(\lambda, n)$ be a tableau with column decomposition $T = C_1C_2\cdots C_k$. Given a column reading word, we can recover the original tableau via *column insertion*: Let $w = w_1 \cdots w_\ell$. We start with $i := 1$, $T = \emptyset$, the empty tableau, and $p = 1$.

1. If w_i is bigger than all entries of C_p, Just add a cell to the column C_p with entry w_i. Else find $\alpha \in C_p$ the smallest entry of C_p bigger or equal than w_i. Then replace α by w_i in C_p and redefine $w_i := \alpha$, p:=p+1 and go to (1) (this is called a *bumping*).

2. If $i \neq \ell$, then $i := i+1$, $p := 1$ and go to (1). Else the algorithm ends.

The Baker-Lecouvey insertion [5, 25] is a bumping algorithm that given a word in the alphabet $[\pm n]$ returns a KN tableau. Let w be a word in the alphabet $[\pm n]$, we call $P(w)$ to the tableau obtained after inserting w. This insertion is similar to the column insertion for SSYT's. In fact both have the same behaviour unless one the following three cases happens:

Suppose that we are inserting the letter α in the column C of the KN tableau and

(I) $\overline{y} \in C$ is the smallest letter bigger or equal then α and $y \in C$, for some $y \in [n]$: there is in C a maximal string of consecutive decreasing integers $y, y-1, \ldots, u+1$ starting in the entry y in the column C. Then the bump consists of replacing the entry $\overline{y}$ with α and subtracting 1 to the entries $y, y-1, \ldots, u+1$. The entry $\overline{u}$ is then inserted in the next column to the right. This is known as the *Type I special bump*.

(II) if $\alpha = x$ and $\bar{x} \in C$, for some $x \in [n]$: there is a maximal string of consecutive decreasing entries $\bar{x}, \overline{x+1}, \ldots, \overline{v-1}$ starting in the entry $\bar{x}$ in C. Let β be the next entry above $\overline{v-1}$. Then we have two subcases:

(a) If $v \leq \beta \leq \overline{v+1}$ then suppose δ is the smallest entry in C which is bigger or equal than v. Then this bump consists of deleting the entry $\bar{x}$, shifting the entries $\overline{x+1}, \ldots, \overline{v-1}$ down one position, inserting $\bar{v}$ where $\overline{v-1}$ was, and replacing δ with v. The entry δ is then bumped into the next column. This is known as the *Type IIa special bump*.

(b) If $\beta \leq v-1$ or β does not exist then there is a maximal string (possibly empty) of consecutive integers $v-1, \ldots, u+1$ above the entry $\overline{v-1}$. The string is not empty only when $\beta = v-1$, or else the string is empty and $u = v-1$. The bump consists of deleting the entry $\bar{x}$, shifting the entries $\overline{x+1}, \ldots, u+1$ down one position, and inserting an entry u where $u+1$ (or $\overline{v-1}$, if $\beta \neq v-1$) was. The entry $\bar{u}$ is then bumped into the next column. This is known as the *Type IIb special bump*.

(III) after adding α in the bottom of the column C, the 1CC breaks at i: then we will slide out the cells that contain $\bar{i}$ and i via symplectic *jeu de taquin*.

In the case III of the Baker-Lecouvey insertion we will be removing a cell from the tableau instead of adding. The length of $cr(P(w))$ might be less than the length of w and the weight is preserved during Baker-Lecouvey insertion, $\mathrm{wt}(w) = \mathrm{wt}(P(w))$.

Remark 2.3.1. *The Baker-Lecouvey insertion is different from what we would have if we use the SSYT column insertion. However, if the word w does not have symmetric letters, then the insertion works just like the column insertion for SSYT's. Apart from this case, if we were to use SSYT column insertion, the final tableau may not even be a KN tableau. For instance, consider the word $w = 2\bar{1}1$. The Baker-Lecouvey insertion of w creates the sequence of tableaux* $\begin{array}{|c|}\hline 2\\\hline\end{array}$ $\begin{array}{|c|}\hline 2\\\hline \bar{1}\\\hline\end{array}$ $\begin{array}{|c|c|}\hline 2 & 2\\\hline \bar{2} \\\cline{1-1}\end{array}$ $= P(2\bar{1}1)$. *The SSYT column insertion of w results in the tableau* $\begin{array}{|c|c|}\hline 1 & 2\\\hline \bar{1} \\\cline{1-1}\end{array}$*, which is not a KN tableau because the first column is not admissible.*

Example 2.3.2. *Consider the word $w = 23\overline{23}1$. We now insert all five letters of w, obtaining the following sequence of tableaux:* $\begin{array}{|c|}\hline 2\\\hline\end{array}$ $\begin{array}{|c|}\hline 2\\\hline 3\\\hline\end{array}$ $\begin{array}{|c|}\hline 2\\\hline 3\\\hline \bar{2}\\\hline\end{array}$ $\begin{array}{|c|c|}\hline 1 & \bar{1}\\\hline 3 \\\cline{1-1} \bar{3} \\\cline{1-1}\end{array}$ $\begin{array}{|c|c|c|}\hline 1 & 1 & \bar{1}\\\hline 3 \\\cline{1-1} \bar{3} \\\cline{1-1}\end{array}$ $= P(w)$. *Note that the insertion of the fourth letter, $\bar{3}$, causes a type I special bump on the first column and the insertion of the fifth letter,* 1, *causes a type IIb special bump on the second column.*

Proposition 2.3.3. *[25, Corollary 6.3.9] Let $T \in \mathcal{KN}(\mu/\nu, n)$. Then the tableau obtained after rectifying T via symplectic jeu de taquin coincides with $P(cr(T))$. Moreover, the insertion of $w = w_1 \ldots w_k$, $P(w)$, is the rectification of the tableau with diagonal shape Δ^n/Δ^{n-1} and column reading w.*

In particular we have that if we insert $cr(T)$ we obtain T again. This implies that during the insertion of $cr(T)$ the case III of the Baker-Lecouvey insertion cannot happen. In Example 2.3.2, we may conclude that $P(23\overline{23}1) = P(cr(P(23\overline{23}1))) = P(\bar{1}113\bar{3})$.

2.4 Robinson-Schensted type C_n correspondence and plactic equivalence

Let $[\pm n]^*$ be the free monoid on the alphabet $[\pm n]$. The type C_n Robinson-Schensted correspondence [25, Theorem 5.2.2] is a combinatorial bijection between words $w \in [\pm n]^*$ and pairs $(P(w), Q)$ where $P(w)$ is a KN tableau and Q is an oscillating tableau, a sequence of Young diagrams that record, by order, the shapes of the tableaux obtained while inserting w, whose final shape is the same as $P(w)$. Every two consecutive shapes of the oscillating tableau differ in exactly one cell and its length is the same of w. Words with the same oscillating tableau identify the coplactic classes in the Robinson-Schensted correspondence. These words are connected by crystal operators [25, Proposition 5.2.1], that we present in the next section.

Since both the SJDT and the Baker-Lecouvey insertion are reversible [5, 25], we have that every pair (P, Q), with the same final shape, is originated by exactly one word. The type C_n Robinson-Schensted correspondence is the following map:

$$\begin{aligned} [\pm n]^* &\to \bigsqcup_{\lambda} \mathcal{KN}(\lambda, n) \times \mathcal{O}(\lambda, n) : \\ w &\mapsto (P(w), Q(w)) \end{aligned}$$

where the union is over all partitions λ with at most n parts, and $\mathcal{O}(\lambda, n)$ is the set of all oscillating tableaux with final shape λ and all shapes of the sequence have at most n rows.

Example 2.4.1. *In Example 2.3.2, the word $w = 23\overline{23}1$ is associated to the pair*

$$\left(\begin{ytableau} 1 & 1 & \bar{1} \\ 3 \\ \bar{3} \end{ytableau}\ ,\ \ydiagram{1}\ \ydiagram{1,1}\ \ydiagram{1,1,1}\ \ydiagram{2,1,1}\ \ydiagram{3,1,1} \right).$$

Given $w_1, w_2 \in [\pm n]^*$, the relation $w_1 \sim w_2 \Leftrightarrow P(w_1) = P(w_2)$ defines an equivalence relation on $[\pm n]^*$ known as *Knuth equivalence* (or plactic equivalence). The type C_n plactic monoid is the quotient $[\pm n]^*/\sim$ where each Knuth (plactic) class is uniquely identified with a KN tableau [23, 25]. The quotient $[\pm n]^*/\sim$ can also be described as the quotient of $[\pm n]^*$ by the *elementary Knuth relations*:

K1: $\gamma\beta\alpha \sim \beta\gamma\alpha$, where $\gamma < \alpha \leq \beta$ and $(\beta, \gamma) \neq (\bar{x}, x)$ for all $x \in [n]$.

K2: $\alpha\beta\gamma \sim \alpha\gamma\beta$, where $\gamma \leq \alpha < \beta$ and $(\beta, \gamma) \neq (\bar{x}, x)$ for all $x \in [n]$.

K3: $y+1\overline{y+1}\beta \sim \bar{y}y\beta$, where $y < \beta < \bar{y}$ and $y \in [n-1]$.

K4: $\beta\bar{y}y \sim \beta y+1\overline{y+1}$, where $y < \beta < \bar{y}$ and $y \in [n-1]$.

K5: $w \sim w \setminus \{z, \bar{z}\}$, where $w \in [\pm n]^*$ and $z \in [n]$ are such that w is a non-admissible column that the $1CC$ breaks at z, and any proper factor of w is an admissible column.

Remark 2.4.2. *It can be proved that given a word $w \in [\pm n]^*$, any proper factor is admissible if and only if any proper prefix of w is admissible. Thus, in order to be able to apply the Knuth relation $K5$ to a subword w' of w, we only need to check that all proper prefixes of w' are admissible, instead of all proper factors.*

When Knuth relations are applied to subwords of a word, the weight is preserved while the length may not. Knuth relations can be seen as *jeu de taquin* moves on words or a diagonally shaped tableau, and each SJDT slide preserves the Knuth class of the reading word of a tableau [25, Theorem 6.3.8]. In Example 2.3.2 the words $23\overline{2}\overline{3}1$ and $\overline{1}113\overline{3}$ are Knuth related: $\overline{1}113\overline{3} \overset{K2}{\sim} \overline{1}131\overline{3} \overset{K2}{\sim} \overline{1}13\overline{3}1 \overset{K3}{\sim} 2\overline{2}3\overline{3}1 \overset{K1}{\sim} 23\overline{2}\overline{3}1$.

The next proposition states that in order to be able to apply the Knuth relation $K5$ to a subcolumn w' of the word w we only need to check that the biggest proper prefix of w' is admissible, instead of all proper factors.

Proposition 2.4.3. *Given a column w on the alphabet $[\pm n]$, we have that any proper factor of w is admissible if and only if the biggest proper prefix of w is admissible.*

Proof. Part "only if": The biggest proper prefix is a proper factor.

Part "if": Any proper factor is contained either in the biggest proper prefix of w or contains the last letter of w, being a proper suffix. Assuming that the statement is false, there is a proper suffix w' of w that breaks the $1CC$ at y. If $\bar{y}$ is not its last letter, if one considers the factor obtained from w' after adding the last letter of w not in w' and removing the last of w', we will have a non-admissible proper factor of w contained in the biggest proper prefix of w, that is admissible, hence we have a contradiction. So $\bar{y}$ is the last letter of w' (and w). In order to break the $1CC$ at y, since there are no letters bigger than $\bar{y}$ in w', we have that $\{1,2,\dots,y\} \in w'$. This implies that there are no letters to the left of 1 in w, because this is the minimal letter of the alphabet. So $w' = w$, which is another contradiction. So w' must be admissible. ☐

2.5 Kashiwara crystal and A_{n-1} and C_n crystals

Let V be an Euclidean space with inner product $\langle\cdot,\cdot\rangle$. Fix a root system Φ with simple roots $\{\alpha_i \mid i \in I\}$ where I is an indexing set and a weight lattice $\Lambda \supseteq \mathbb{Z}\text{-span}\{\alpha_i \mid i \in I\}$. A *Kashiwara crystal* of type Φ is a non-empty set $\mathfrak{B}$ together with maps [9]:

$$e_i, f_i : \mathfrak{B} \to \mathfrak{B} \sqcup \{0\} \quad \varepsilon_i, \varphi_i : \mathfrak{B} \to \mathbb{Z} \sqcup \{-\infty\} \quad \mathrm{wt} : \mathfrak{B} \to \Lambda$$

where $i \in I$ and $0 \notin \mathfrak{B}$ is an auxiliary element, satisfying the following conditions:

1. if $a, b \in \mathfrak{B}$ then $e_i(a) = b \Leftrightarrow f_i(b) = a$. In this case, we also have $\mathrm{wt}(b) = \mathrm{wt}(a) + \alpha_i$, $\varepsilon_i(b) = \varepsilon_i(a) - 1$ and $\varphi_i(b) = \varphi_i(a) + 1$;

2. for all $a \in \mathfrak{B}$, we have $\varphi_i(a) = \langle \mathrm{wt}(a), \frac{2\alpha_i}{\langle \alpha_i, \alpha_i \rangle} \rangle + \varepsilon_i(a)$.

The crystals we deal with are seminormal [9], i.e., $\varphi_i(a) = \max\{k \in \mathbb{Z} \geq 0 \mid f_i^k(a) \neq 0\}$ and $\varepsilon_i(a) = \max\{k \in \mathbb{Z} \geq 0 \mid e_i^k(a) \neq 0\}$. An element $u \in \mathfrak{B}$ such that $e_i(u) = 0$ for all $i \in I$ is called a *highest weight element*. A *lowest weight element* is an element $u \in \mathfrak{B}$ such that $f_i(u) = 0$ for all $i \in I$. We associate with $\mathfrak{B}$ a coloured oriented graph with vertices in $\mathfrak{B}$ and edges labelled by $i \in I$: $b \xrightarrow{i} b'$ if and only if $b' = f_i(b)$, $i \in I$, $b, b' \in \mathfrak{B}$. This is the *crystal graph* of $\mathfrak{B}$.

A *morphism* $\psi : \mathfrak{B} \to \mathfrak{B}'$ of crystal graphs is a map that preserves coloured directed edges and weights. More precisely, a morphism is a map $\psi : \mathfrak{B} \to \mathfrak{B}'$ that satisfies

$$\psi(fi(b)) = fi(\psi(b))$$
$$\psi(ei(b)) = ei(\psi(b))$$
$$\mathrm{wt}(\psi(b)) = \mathrm{wt}(b).$$

where $\psi(\emptyset) = \emptyset$ by convention. Note that composing morphisms yields a morphism. An *isomorphism* of crystal graphs is a bijective morphism of crystal graphs whose inverse function is also a morphism of crystal graphs.

If $\mathfrak{B}$ and $\mathfrak{C}$ are two seminormal crystals associated to the same root system, the *tensor product* $\mathfrak{B} \otimes \mathfrak{C}$ is also a seminormal crystal. As a set, we will have the Cartesian product $\mathfrak{B} \times \mathfrak{C}$, where its elements are denoted by $b \otimes c$, $b \in \mathfrak{B}$ and $c \in \mathfrak{C}$, with $\mathrm{wt}(b \otimes c) = \mathrm{wt}(b) + \mathrm{wt}(c)$,

$$f_i(b \otimes c) = \begin{cases} f_i(b) \otimes c \text{ if } \varphi_i(c) \leq \varepsilon_i(b) \\ b \otimes f_i(c) \text{ if } \varphi_i(c) > \varepsilon_i(b) \end{cases}, \; e_i(b \otimes c) = \begin{cases} e_i(b) \otimes c \text{ if } \varphi_i(c) < \varepsilon_i(b) \\ b \otimes e_i(c) \text{ if } \varphi_i(c) \geq \varepsilon_i(b) \end{cases}.$$

If $\mathfrak{B}$ and $\mathfrak{C}$ are finite, $\varphi_i(b \otimes c) = \varphi_i(b) + \max(0, \varphi_i(c) - \varepsilon_i(b))$ and $\varepsilon_i(b \otimes c) = \varepsilon_i(b) + \max(0, \varepsilon_i(b) - \varphi_i(c))$.

In type A_{n-1}, we consider $\{e_i\}_{i=1}^n$ the canonical basis of $\mathbb{R}^n$. The root system is $\Phi_A = \{\pm e_i \pm e_j \mid i < j\}$ and the simple roots are $\alpha_i = e_i - e_{i+1}$, for $i \in [n-1]$. In type C_n, we consider the same the canonical basis of $\mathbb{R}^n$. The root system is $\Phi_C = \{\pm e_i \pm e_j \mid i < j\} \cup \{\pm 2e_i\}$ and the simple roots are $\alpha_i = e_i - e_{i+1}$, if $i \in [n-1]$, $\alpha_n = 2e_n$. For both types, the weight lattice $\mathbb{Z}^n$ has dominant weights $\lambda = (\lambda_1 \geq \cdots \geq \lambda_n \geq 0)$.

In type C_n, the standard crystal is seminormal and has the following crystal graph:

$$1 \xrightarrow{1} 2 \xrightarrow{2} \dots \xrightarrow{n-1} n \xrightarrow{n} \overline{n} \xrightarrow{n-1} \dots \xrightarrow{1} \overline{1}$$

with set $\mathfrak{B} = [\pm n]$, $\mathrm{wt}(\boxed{i}) = \mathbf{e_i}$, $\mathrm{wt}(\boxed{\bar{i}}) = -\mathbf{e_i}$. The highest weight element is the word 1, and the highest weight $\mathbf{e_1}$. We denote the crystal by $\mathfrak{B}^{\mathbf{e_1}}$. The type A_{n-1} standard crystal uses only the first n vertices of the type C_n standard crystal, hence it has the same highest weight element.

The crystal $\mathfrak{B}^{\mathbf{e_1}}$ is the crystal on the words of $[\pm n]^*$ of a sole letter. The tensor product of crystals allows us to define the crystal $G_n = \bigoplus_{k \geq 0} (\mathfrak{B}^{\mathbf{e_1}})^{\otimes k}$ of all words in $[\pm n]^*$, where the vertex $w_1 \otimes \cdots \otimes w_k$ is identified with the word $w_1 \dots w_k \in [\pm n]^*$. The action of the operators e_i and f_i is easily given by the signature rule [9, 20, 25]. We substitute each letter w_j by $+$ if $w_j \in \{i, \overline{i+1}\}$ or by $-$ if $w_j \in \{i+1, \bar{i}\}$, and erase it in any other case. Then successively erase any pair $+-$ until all the remaining letters form a word that looks like $-^a+^b$. Then $\varphi_i(w) = b$ and $\varepsilon_i(w) = a$, e_i acts on the letter associated to the rightmost unbracketed $-$ (i.e., not erased), whereas f_i acts on the letter w_j associated to the leftmost unbracketed $+$,

$$f_i(w_j) = \begin{cases} i+1 \text{ if } w_j = i \text{ and } i \neq n \\ \bar{i} \text{ if } w_j = \overline{i+1} \\ \overline{n} \text{ if } w_j = i \text{ and } i = n \end{cases},$$

and the other letters of w are unchanged, and e_i is the inverse map. If $b=0$ then $f_i(w)=0$ and if $a=0$ then $e_i(w)=0$. This description of the signature rule also works in type A_{n-1}, with the only difference being the fact that all letters of w are positive.

Example 2.5.1. *Consider $w=\overline{2}31\overline{2}2\overline{1}$ and $i=1$. Using the signature rule we rewrite w as $++++--$. Now we erase pairs $+-$ as many times as possible, obtaining only $+$, that came from the first $\overline{2}$ in w.*

Given that $f_1(\overline{2})=\overline{1}$, we have that $f_1(w)=\overline{1}31\overline{2}2\overline{1}$. Also, since there are no $-$ after eliminating all $+-$ pairs, we have that $e_1(w)=0$.

The crystal G_n, as a graph, is the union of connected components where each component has a unique highest weight word. Two connected components are isomorphic if and only if they have the same highest weight [19]. Two words in $[\pm n]^*$ are said to be crystal connected or coplactic equivalent if and only if they belong to the same connected component of G_n. This means that both words are obtained from the same highest weight word, through a sequence of crystal operators f_i, or one is obtained from another by some sequence of crystal operators f_i and e_j, $i,j\in[n]$.

The connected components of G_n are the coplactic classes in the Robinson-Schensted correspondence that identify words with the same oscillating tableau [25, Proposition 5.2.1]. Also, two words $w_1,w_2\in[\pm n]^*$ are Knuth equivalent if and only if they occur in the same place in two isomorphic connected components of G_n, that is, they are obtained from two highest words with the same weight through a same sequence of crystal operators [25]. Crystal operators are coplactic and commute with the *jeu de taquin*. The next proposition identifies all highest weight words of G_n.

Proposition 2.5.2. *Let w be a word in the alphabet $[\pm n]$. Then w is a highest weight word if and only if the weight of all its prefixes (including itself) is a partition. In this case, one has that $P(w)=K(\lambda)$, the tableau of shape and weight λ, also known as Yamanouchi tableau.*

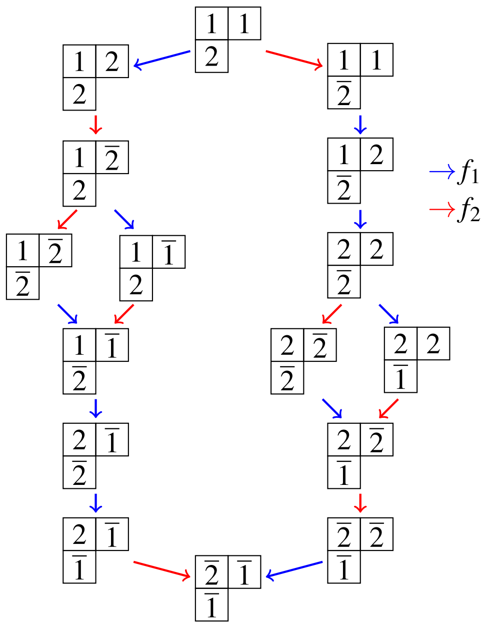
Proof. Part "if": We will prove the contraposition of the statement. There is a i such that $e_i(w)\neq 0$. Let k be the position of the leftmost $-$ of the signature rule of w, and consider the prefix w_k with the first k letters. Since the k-th letter of w had an unbracketed $-$ in the signature rule then the last letter of w_k will also be an unbracketed $-$. Hence there are more $-$ than $+$ in the signature rule of w_k. Let t_α be the number of α in w_k. We have that $t_i+t_{\overline{i+1}}<t_{i+1}+t_{\overline{i}}\Leftrightarrow t_i-t_{\overline{i}}<t_{i+1}-t_{\overline{i+1}}$, hence the weight of w_k is not a partition.

Part "only if": We will once again prove the contrapositive of the statement. Let w_k be a prefix such that its weight is not a partition. Hence there is $i\in[n]$ such that $t_i-t_{\overline{i}}<t_{i+1}-t_{\overline{i+1}}\Leftrightarrow t_i+t_{\overline{i+1}}<t_{i+1}+t_{\overline{i}}$, hence for this i there will be more $-$ than $+$ in the signature rule of w_k. So in the first k letters of w there will be more $-$ than $+$, so there is an unbracketed $-$ in w, hence $e_i(w)\neq 0$. Note that the argument works even if $i=n$. In this case we need to assume $t_{n+1}=t_{\overline{n+i}}=0$.

It follows from [25, Proposition 3.2.6] that the insertion of the highest word w of weight λ is $K(\lambda)$. □

Choose a word $w\in[\pm n]^*$ such that the shape of $P(w)$ is λ. If we replace every word of its coplactic class with its insertion tableau we obtain the crystal of tableaux $\mathfrak{B}^\lambda$ that has all KN tableaux of shape λ on the alphabet $[\pm n]$. The crystal $\mathfrak{B}^\lambda$ does not depend on the initial choice of word w, as long as $P(w)$ has shape λ. [25, Theorem 6.3.8].

Example 2.5.3. *Here we have the type C_2 crystal graph $\mathcal{KN}((2,1),2)$ containing the A_1 crystal $\mathcal{SSYT}((2,1),2)$:*



Chapter 3

Weyl group of type C_n, Bruhat order and symplectic key tableaux

In this chapter we will present the group B_n, known as hyperoctahedral group, which is the type C_n Weyl group. The main result of this section is a tableau criterion for the Bruhat order on the elements of B_n and on the coset space of B_n define by a parabolic subgroup, using *symplectic key tableaux*. This shows that the combinatorics of the type C_n crystal graphs are strongly connected with the Bruhat order of B_n.

3.1 Weyl group of type C_n

Consider the group B_n, with generators s_i, $1 \leq i \leq n$, having the following presentation, regarding the relations among the generators,

$$B_n := \langle s_1, \dots, s_n \,| s_i^2 = 1,\, 1 \leq i \leq n;\; (s_i s_{i+1})^3 = 1, 1 \leq i \leq n-2;\; (s_{n-1} s_n)^4 = 1;\\ (s_i s_j)^2 = 1,\, 1 \leq i < j \leq n,\, |i-j| > 1 \rangle,$$

known as hyperoctahedral group or signed symmetric group. This group is a Coxeter group [6].

The elements of B_n can be seen as odd bijective maps from $[\pm n]$ to itself, i.e., for all $\sigma \in B_n$ we have $\sigma(i) = \overline{\sigma(\overline{i})}$, $i \in [\pm n]$. The subgroup with the generators $s_1, \dots, s_{n-1}$ is the symmetric group $\mathfrak{S}_n$ and its elements can be seen as bijections from $[n]$ to itself. Both groups can also be seen as groups of $n \times n$ matrices. The elements of the symmetric group can be identified with the permutation matrices, and if we allow the non-zero entries to be either 1 or -1, we have the elements of B_n. Hence B_n has $2^n n!$ elements. The groups $\mathfrak{S}_n$ and B_n are the Weyl groups for the root systems of types A_{n-1} and C_n, respectively.

Let $\sigma \in B_n$. We denote $[a_1\, a_2 \dots a_n]$, where $a_i = \sigma(i)$ for $i \in [n]$, the window notation of σ, and write $\sigma = [a_1\, a_2 \dots a_n]$. The elements of B_n, or $\mathfrak{S}_n$, act on vectors in $\mathbb{Z}^n$ on the left. Given a vector $v \in \mathbb{Z}^n$, we have that s_i, with $i \in [n-1]$, acts on v swapping the i-th and the $(i+1)$-th entries, and s_n acts on v, $s_n v$, changing the sign of the last entry. Note that the window notation of σs_i is obtained after applying s_i to the window notation of σ, if we see it as a vector. Ignoring signs, $\sigma v = (v_{\sigma^{-1}(1)}, \dots, v_{\sigma^{-1}(n)})$, with $v = (v_1, \dots, v_n)$. The i-th letter of σv changes its sign if and only if $\bar{i}$

appears in σ. Hence $\sigma v = (sgn(\sigma^{-1}(1))v_{|\sigma^{-1}(1)|}, \ldots, sgn(\sigma^{-1}(n))v_{|\sigma^{-1}(n)|})$, where $sgn(x) = 1$ if x is positive and -1 if x is negative, for $x \in [\pm n]$.

Example 3.1.1. *Consider* $v = (1,2,3) \in \mathbb{Z}^3$ *and* $\sigma = [2\,\overline{3}\,1] = [s_1s_3s_2(1), s_1s_3s_2(2), s_1s_3s_2(3)] = s_1s_3s_2 \in B_3$. *So*

$$\begin{aligned}\sigma(1,2,3) &= s_1s_3s_2(1,2,3) = s_1s_3(1,3,2) = s_1(1,3,\overline{2}) = (3,1,\overline{2})\\ &= (sgn(\sigma^{-1}(1))v_{|\sigma^{-1}(1)|}, sgn(\sigma^{-1}(2))v_{|\sigma^{-1}(2)|}, sgn(\sigma^{-1}(3))v_{|\sigma^{-1}(3)|})\\ &= (1\cdot 3, 1\cdot 1, -1\cdot 2).\end{aligned}$$

3.2 Bruhat order on B_n

The *length* of $\sigma \in B_n$, $\ell(\sigma)$, is the least number of generators of B_n needed to go from $[1\,2\,\ldots\,n]$, the identity map, to σ. Any expression of σ as a product of $\ell(\sigma)$ generators of B_n is called reduced. We say that two letters of the window notation of σ form an inversion if the bigger letter appears first. The next proposition gives a way to compute $\ell(\sigma)$ that only requires to look at the window notation of σ. This is a variation of the length formula presented on [6, Proposition 8.1.1], where the authors consider the usual ordering of the alphabet $[\pm n]$ and the generator that changes the sign of an entry of the window notation acts on the first entry instead of the last one.

Proposition 3.2.1. *Consider* $\sigma \in B_n$. *Then*

$$\ell(\sigma) = \#\{\textit{inversions of } \sigma\} + \sum_{\bar{i} \textit{ appears in } \sigma} (n+1-i).$$

The (signed) permutation $\sigma = [2\,\overline{3}\,1]$ has two inversions: 2, 1 and $\overline{3}$, 1 and $\ell(\sigma) = 3$.

Remark 3.2.2.

- *If* $\bar{i}$ *does not appear in the window presentation of* σ, *for all* $i \in [n]$, *we may identify* σ, *in one-line notation, with* $\sigma(1)\ldots\sigma(n) \in \mathfrak{S}_n$ *and* $\ell(\sigma) = \#\{\textit{inversions of } \sigma\}$ *[6, Proposition 1.5.2].*

- *We have that* $\ell(\sigma s_i) > \ell(\sigma)$ *if* $i = n$ *and* $\sigma(n)$ *is positive, or,* $i \neq n$ *and* $\sigma(i) < \sigma(i+1)$.

The Bruhat order on the set of the elements of B_n can be defined in the following way:

Definition 3.2.3. *[6] Let* $w = \sigma_1 \ldots \sigma_{\ell(w)}$, *where* $\sigma_i \in \{s_1, \ldots, s_n\}$ *are generators of* B_n, *and* u *be two elements in* B_n. *Then* $u \leq w$ *in the Bruhat order if*

$$\exists 1 \leq i_1 < i_2 \cdots < i_{\ell(u)} \leq \ell(w) \textit{ such that } u = \sigma_{i_1}\sigma_{i_2}\ldots\sigma_{i_{\ell(u)}}.$$

By definition, if $u \leq w$ then $\ell(u) \leq \ell(w)$, but the reverse is not true. If $\sigma(n)$ is positive and $i = n$, or, $\sigma(i) < \sigma(i+1)$ and $i \neq n$, we can also say that $\sigma s_i > \sigma$.

3.3 Symplectic key tableaux in type C_n and the Bruhat order on B_n

Definition 3.3.1. *A* key tableau *of shape λ, in type C_n, is a KN tableau in $\mathcal{KN}(\lambda, n)$, in which the set of elements of each column, left to right, is contained in the set of elements of the previous column, if any, and the letters i and $\bar{i}$ do not appear simultaneously as entries, for any $i \in [n]$. Equivalently, the key tableaux in type C_n are the KN tableaux of shape λ whose weight is in $B_n\lambda$, the B_n-orbit of λ. For each element of $B_n\lambda$ there is exactly one key tableau of shape λ with that weight (see Proposition 3.3.3).*

Example 3.3.2. *The KN tableau* $T = \begin{array}{|c|c|}\hline 2 & 2 \\ \hline 3 & \bar{1} \\ \hline \bar{1} & \\ \cline{1-1} \end{array}$ *is a key tableau.*

The set of key tableaux in type A_{n-1} is the subset of the key tableaux in type C_n consisting of the tableaux having only positive entries, hence they are SSYT's for the alphabet $[n]$.

Every vector v of $\mathbb{Z}^n$ is in the B_n-orbit of exactly one partition, λ_v, which is the one obtained by sorting the absolute values of all entries of v. Given a partition $\lambda \in \mathbb{Z}^n$, the *B_n-orbit of λ* is the set $B_n\lambda := \{\sigma\lambda \mid \sigma \in B_n\}$. For instance, the vector $v = (1, \bar{3}, 0, 3, \bar{2})$ is in the B_5-orbit of $\lambda = (3, 3, 2, 1, 0)$.

Proposition 3.3.3. *Let λ be a partition and $v \in B_n\lambda$. There is exactly one key tableau $K(v)$ whose weight is v. In addition, the shape of the key tableau $K(v)$ is λ. When $v = \lambda$, $K(\lambda)$ is the only KN tableau of weight and shape λ, also called Yamanouchi tableau of shape λ.*

Proof. Existence: Given $v = (v_1, \ldots, v_n) \in \mathbb{Z}^n$ there exists a key tableau K of weight v by putting in the first $|v_i|$ columns the letter i if $v_i \geq 0$ or $\bar{i}$ if $v_i \leq 0$, and then sorting the columns properly. Clearly the columns of K are nested and it is a KN tableau without symmetric entries, hence it is a key tableau. Also, its shape is $\lambda_v = \lambda$.

Uniqueness: Since the key tableaux do not have symmetric entries then, for all $i \in [n]$, we have that in K the letter $sgn(v_i)i$ appears $|v_i|$ times in its entries. In order to the columns of K be nested we have that these $|v_i|$ entries appear in the first $|v_i|$ columns, hence we have determined exactly which letters appear in each column of K and now we just have to order them correctly. So the key tableau K with weight v is unique. When $v = \lambda$, $K(\lambda)$ has only i's in the row i, for $i \in [n]$. □

Example 3.3.4. *Let $v = (1, \bar{3}, 0, 3, \bar{2})$. Then* $K(v) = \begin{array}{|c|c|c|}\hline 1 & 4 & 4 \\ \hline 4 & \bar{5} & \bar{2} \\ \hline \bar{5} & \bar{2} & \\ \cline{1-2} \bar{2} & & \\ \cline{1-1} \end{array}$.

Hence there is a bijection between vectors in $B_n\lambda$ and the key tableaux in type C_n on the alphabet $[\pm n]$ with shape λ, given by the map $v \mapsto K(v)$.

Remark 3.3.5. *The type C_n key tableaux in $\mathcal{KN}(\lambda, n)$ are characterized by their weight $\alpha\lambda$, for all $\alpha \in B_n$, and thereby denoted $K(\alpha\lambda)$. The orbit of $K(\lambda)$, the highest weight element of $\mathfrak{B}_\lambda$, under the action of the Weyl group B_n, is defined to be $O(\lambda) = \{K(\alpha\lambda) : \alpha \in B_n\}$. In particular, $K(w_0\lambda) = K(-\lambda)$, with w_0 the longest element of B_n, is the lowest weight element of the type C_n crystal $\mathfrak{B}^\lambda$.*

If $\sigma \in B_n$ we put $K(\sigma) := K(\sigma\Delta^n)$, where Δ^n is the staircase partition. One has a natural bijection between B_n and the B_n-orbit of Δ^n.

Proposition 3.3.6. *If $\sigma \in B_n$ has the letter α in the j-th position then α appears in the first $n+1-j$ columns of the corresponding key tableau, $K(\sigma)$.*

Proof. Put $\Delta := \Delta^n$. Remember that, ignoring signs, $\sigma\Delta = (\Delta_{\sigma^{-1}(1)}, \dots, \Delta_{\sigma^{-1}(n)})$. The i-th letter of $\sigma\Delta$ has negative sign if and only if $\bar{i}$ appears in σ. If α is positive, then in the position α of $\sigma\Delta$ will appear $\Delta_j = n+1-j$. If α is negative, then in the position $-\alpha$ will appear $\overline{\Delta_j} = \overline{n+1-j}$. $\square$

We now append 0 to the alphabet $[\pm n]$, obtaining $[\pm n] \cup \{0\}$, where $n < 0 < \bar{n}$, and, for all $\sigma \in B_n$, we put $\sigma(0) := 0$. Given an element $\sigma \in B_n$ consider the map

$$[\pm n] \cup \{0\} \times [\pm n] \cup \{0\} \to \mathbb{N}_0$$
$$(i,j) \mapsto |\{a \leq i : \sigma(a) \geq j\}| := \sigma[i,j].$$

This map $\sigma[\cdot,\cdot]$, originally defined in [6], produces a table which is related to key tableaux in type C_n. See example below:

Example 3.3.7. *Let $\sigma = [\bar{3}\,\bar{1}\,2\,4]$. Then $\sigma(4,3,2,1) = (\bar{3},2,\bar{4},1)$ and* $K(\sigma) = \begin{array}{|c|c|c|c|} \hline 2 & 2 & \bar{3} & \bar{3} \\ \hline 4 & \bar{3} & \bar{1} \\ \cline{1-3} \bar{3} & \bar{1} \\ \cline{1-2} \bar{1} \\ \cline{1-1} \end{array}$

The family of numbers $\sigma[i,j]$ where $(i,j) \in [\pm n] \cup \{0\} \times [\pm n] \cup \{0\}$ originates the following table, where i indexes the columns, left to right, and j indexes the rows, top to bottom. We add a row of zeros at the bottom for convenience:

	1	**2**	**3**	**4**	**0**	$\bar{\mathbf{4}}$	$\bar{\mathbf{3}}$	$\bar{\mathbf{2}}$	$\bar{\mathbf{1}}$
1	1	2	3	4	5	6	7	8	9
2	1	2	3	4	5	6	7	7	8
3	1	2	2	3	4	5	6	6	7
4	1	2	2	3	4	5	6	6	6
0	1	2	2	2	3	4	5	5	5
$\bar{\mathbf{4}}$	1	2	2	2	2	3	4	4	4
$\bar{\mathbf{3}}$	1	2	2	2	2	2	3	3	3
$\bar{\mathbf{2}}$	0	1	1	1	1	1	2	2	2
$\bar{\mathbf{1}}$	0	1	1	1	1	1	1	1	1
	0	0	0	0	0	0	0	0	0

To go from the table to the key tableau note that, for $i \in [n]$, the i-th column of the table encodes the $(n+1-i)$-th column of the tableau, in the sense that if we look to the the i-th column of the table, from bottom to top, if the entry of the table increases in one unity then the index of the row associated to that entry exists in the $(n+1-i)$-th column of the tableau. Knowing the entries in a column of a tableau, its ordering is unique. The columns of the tableau constructed this way are nested because the indexes in which the column i increases are $\sigma(j)$, for $j \leq i$. So the tableau taken from the table is the key tableau $K(\sigma)$. It is also possible to construct the table from the key tableau and then we only need the first n columns of the table.

We then have the following result:

Proposition 3.3.8. *Consider $\sigma, \rho \in B_n$. $K(\sigma) \geq K(\rho)$ entrywise if and only if $\sigma[i,j] \geq \rho[i,j]$, where $i \in [n]$, and $j \in [\pm n]$.*

In [6, Theorem 8.1.8] it is proved that, for $\sigma, \rho \in B_n$, $\sigma \leq \rho$ in the Bruhat order if and only if $\sigma[i,j] \leq \rho[i,j]$ for all $i,j \in [\pm n]$. But the result in [6, Theorem 8.1.7] implies that we only need to compare $\sigma[i,j]$ and $\rho[i,j]$ for $i \in [n]$. Henceforth, we have the following criterion for the Bruhat order on B_n:

Theorem 3.3.9. *Consider $\sigma, \rho \in B_n$. $K(\sigma) \geq K(\rho)$ by entrywise comparison if and only if $\sigma \geq \rho$ in the Bruhat order.*

Remark 3.3.10. *In [6, Chapter 8.1] the authors use the same alphabet as here, but with the usual ordering on the integers. So, to translate the results from there to here, it is needed to apply the ordering isomorphism defined by: $i \mapsto \overline{n-i+1}$ if $i \in [n]$; $i \mapsto n+i+1$ if $i \in -[n]$; $0 \mapsto 0$. Using the usual ordering, the authors give a tableau criterion for the Bruhat order in Exercise* 6*, pp.* 287–288*, which is effectively the transpose version of the tableau criterion presented here. Also note that the generators used in [6, Chapter 8.1] are the same used here, although with different indexation. Our generator s_i corresponds to the generator s_{n-i} in [6, Chapter 8.1], for all $i \in [n]$.*

3.4 The Bruhat order on cosets of B_n defined by parabolic subgroups

A *parabolic subgroup* of B_n is a group generated by some of the generators $s_1, s_2, \ldots, s_n$ of B_n. Consider a partition $\lambda \in \mathbb{N}^n$. Let $W_\lambda = \{\rho \in B_n \mid \rho\lambda = \lambda\}$ be the stabilizer of λ, under the action of B_n. Since λ is a partition, W_λ is a *parabolic subgroup* of B_n, because it is generated by some of the generators of B_n. Also note that given a parabolic subgroup G of B_n, there is a partition λ such that $G = W_\lambda$. Let $J \subseteq [n]$ be the set of the indices of the generators of W_λ, i.e. $W_\lambda = \langle s_j, j \in J \rangle$, and J^c the complement of this set in $[n]$. Let $B_n/W_\lambda = \{\sigma W_\lambda : \sigma \in B_n\}$ be the set of left cosets of B_n determined by the subgroup W_λ. Each coset σW_λ returns a unique vector v when acting on λ, and has a unique minimal length element σ_v, such that $v = \sigma_v \lambda$. Reciprocally, given a vector $v \in B_n\lambda$, there is a unique minimal length element $\sigma_v \in B_n$ such that $v = \sigma_v\lambda$. We have then a bijection between the vectors in $B_n\lambda$ and the left cosets of B_n, determined by the subgroup W_λ, given by $v \mapsto \sigma_v W_\lambda$. The set J^c detects the minimal length coset representatives of B_n/W_λ: σ is a minimal coset representative of B_n/W_λ if and only if all its reduced decompositions end with a generator $s_i \in J^c$ [6]. However key tableaux, $K(v)$, $v \in B_n\lambda$, may be used to explicitly construct the minimal length coset representatives of B_n/W_λ. Given a vector $v \in B_n\lambda$, we show that there is a unique minimal length element $\sigma_v \in B_n$ such that $v = \sigma_v\lambda$ and we show how to obtain σ_v explicitly. The next proposition is a generalization of what Lascoux does in [22] for vectors in $\mathbb{N}^n$ (hence $\sigma_v \in \mathfrak{S}_n$).

Proposition 3.4.1. *Let $v \in B_n\lambda$ and T the tableau obtained after adding the column $C = \begin{array}{|c|}\hline 1 \\ \hline 2 \\ \hline \vdots \\ \hline n \\ \hline \end{array}$ to the left of $K(v)$. The minimal length element $\sigma \in B_n$, modulo W_λ, is given by the reading word of T where entries with the same absolute value are only read on their first appearance.*

Proof. Consider $\lambda = (\lambda_1, \dots, \lambda_n)$. Let a_i be the multiplicity of i in λ, for $0 \le i \le \lambda_1$. In this proof we will write λ as $(\lambda_1^{a_{\lambda_1}}, (\lambda_1 - 1)^{a_{\lambda_1 - 1}}, \dots, 1^{a_1}, 0^{a_0})$. Note that $\sum_{i=0}^{\lambda_1} a_i = n$.

Let $\sigma = [\alpha_1 \dots \alpha_n] \in B_n$ read from T. Let's prove that α_j appears λ_j times in $K(v)$: If $j = 1$ then α_1 appears in all columns of $K(v)$, because it was the first letter read and the columns are nested. Hence it appears λ_1 times. Also, the $|\alpha_1|$-th entry of $\lambda\sigma$ is $sgn(\alpha_1)\lambda_1$ which is the weight of $|\alpha_1|$ in $K(v)$. For $j > 1$, proceeding inductively, we have that α_j appears in all columns of $K(v)$ not fully occupied by α_i, with $i < j$, hence it appears λ_j times. Also, the $|\alpha_j|$-th entry of $\lambda\sigma$ is $sgn(\alpha_j)\lambda_j$, which is the weight of $|\alpha_j|$ in $K(v)$. This makes sense even if $\lambda_j = 0$. So we have that $\sigma\lambda = v$.

We only have to see that σ is the minimal length element of the set $\{\rho \in B_n \mid \rho\lambda = v\}$. The subset of elements B_n that applied to λ returns v is the coset σW_λ. Looking at σ, this allows us to swap α_i and α_j in σ if $\lambda_i = \lambda_j$ and to change the sign of α_i if $\lambda_i = 0$. Since for each column the reading to obtain σ is ordered from the least to the biggest, we have that between these elements of B_n, σ has minimal number of inversions and the letter α_j is unbarred if $\lambda_j = 0$ because α_j will only be added to σ when reading the column C. Hence, by Proposition 3.2.1, σ is the minimal length element of σW_λ. □

Given a partition $\lambda \in \mathbb{Z}^n$ we identify each coset σW_λ with its minimal length representative σ_v, where $v = \sigma\lambda \in B_n\lambda$. Under this identification, we now induce the Bruhat order in the B_n-orbit of λ and in the coset space of B_n/W_λ.

Definition 3.4.2. *Consider the vectors $v, w \in B_n\lambda$, where λ is a partition. We say that $v \le w$, in the Bruhat order, if $\sigma_v \le \sigma_w$.*

Let $v \in B_n\lambda$. If $K := K(v)$ is the key tableau with weight v, consider the tableau $\widetilde{K}$ obtained from K after erasing the minimal number of columns in order to have a tableau with no duplicated columns. Let $\widetilde{v}$ and $\widetilde{\lambda}$ be the weight and the shape of $\widetilde{K}$, respectively. If K and K' are two key tableaux with shape λ, we have that $K \ge K'$ (by entrywise comparison) if and only if $\widetilde{K} \ge \widetilde{K}'$. Note that to recover K from $\widetilde{K}$ we just have to know λ, and that $\widetilde{K} = K(\widetilde{v})$.

It is also possible to obtain $\widetilde{v}$ from v without having to look to key tableaux. If i is positive, i and $\bar{i}$ do not appear in v but $i+1$ or $\overline{i+1}$ appears then change all appearances of $i+1$ and $\overline{i+1}$ to i and $\bar{i}$, respectively, and repeat this as many times as possible, obtaining the vector $\widetilde{v}$. The set of the absolute values of its entries is a set of consecutive integers starting either in 0 or 1. Hence the key tableau associated to it does not have repeated columns.

Due to Proposition 3.4.1 we have that $\sigma_{\widetilde{v}} = \sigma_v$ and $\widetilde{v} = \sigma_{\widetilde{v}}\widetilde{\lambda}_v = \widetilde{\sigma_v\lambda_v}$.

Example 3.4.3. *Consider $v = (1, 0, \bar{3}, 3, \bar{5}) \in B_5(5,3,3,1,0)$. Hence* $K(v) = \begin{array}{|c|c|c|c|c|} \hline 1 & 4 & 4 & \bar{5} & \bar{5} \\ \hline 4 & \bar{5} & \bar{5} & & \\ \cline{1-3} \bar{5} & \bar{3} & \bar{3} & & \\ \cline{1-3} \bar{3} & & & & \\ \cline{1-1} \end{array}$ *has shape $\lambda = (5,3,3,1,0)$, weight v and $\sigma_v = [\bar{5}4\bar{3}12]$. Now note that $\widetilde{v} = (1,0,\bar{2},2,\bar{3})$, hence* $K(\widetilde{v}) = \begin{array}{|c|c|c|} \hline 1 & 4 & \bar{5} \\ \hline 4 & \bar{5} & \\ \cline{1-2} \bar{5} & \bar{3} & \\ \cline{1-2} \bar{3} & & \\ \cline{1-1} \end{array}$ $= \widetilde{K(v)}$ *has shape $(3,2,2,1,0) = \widetilde{\lambda}$ and $\sigma_{\widetilde{v}} = [\bar{5}4\bar{3}12] = \sigma_v$.*

Recall J and J^c defined above. Note that the set J is the same for λ and $\widetilde{\lambda}$. If $i \in J^c$ and $i = n$ then all entries of λ are different from 0, which implies $K(v)$ (and $\widetilde{K(v)}$) having columns of length n; if $i \in J^c$ and $i < n$ then $\lambda_i > \lambda_{i+1}$, hence $K(v)$ will have exactly i rows with length greater then λ_{i+1}, hence $K(v)$ (and $\widetilde{K(v)}$) will have columns of length i. Since $\widetilde{K(v)}$ does not have repeated columns, J^c have exactly the information of what column lengths exist in $\widetilde{K(v)}$. Theorem 3BC of Proctor's Ph.D. thesis [33] states that given a partition λ there is a poset isomorphism between the poset formed by the key tableaux of shape $\widetilde{\lambda}$ (ordered by entrywise comparison) and the poset formed by the Bruhat order in the vectors of the orbit $B_n\tilde{\lambda} = \{\sigma\tilde{\lambda} : \sigma \in B_n\}$.

The following theorem gives a tableau criterion for the Bruhat order on vectors in the same B_n-orbit and for the corresponding B_n-coset space.

Theorem 3.4.4. *Let $v, u \in B_n\lambda$. Then $\sigma_v \leq \sigma_u$ if and only if $K(v) \leq K(u)$.*

Proof. We have that

$$\sigma_v \leq \sigma_u \overset{(1)}{\Leftrightarrow} v \leq u \overset{(2)}{\Leftrightarrow} \widetilde{v} \leq \widetilde{u} \overset{(3)}{\Leftrightarrow} K(\widetilde{v}) \leq K(\widetilde{u}) \Leftrightarrow \widetilde{K(v)} \leq \widetilde{K(u)} \overset{(4)}{\Leftrightarrow} K(v) \leq K(u),$$

where (1) holds by Definition 3.4.2. Note that in (2) we also need to record λ, because it is needed in (4) to recover the shape of $K(v)$ from the shape $\widetilde{K(v)}$. Finally the equivalence (3) is an application of Theorem 3BC of Proctor's Ph.D. thesis [33]. □

The following example illustrates Theorem 3.4.4.

Example 3.4.5. *Here we have two vectors with the respective key tableaux, ordered by entrywise comparison. The corresponding minimal coset representatives, calculated using Proposition 3.4.1, preserve this order.*

$K(3,\overline{3},0,0,\overline{2}) = \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline \overline{5} & \overline{5} & \overline{2} \\ \hline \overline{2} & \overline{2} \\ \cline{1-2} \end{array} \leq K(\overline{3},2,0,\overline{3},0) = \begin{array}{|c|c|c|} \hline 2 & 2 & \overline{4} \\ \hline \overline{4} & \overline{4} & \overline{1} \\ \hline \overline{1} & \overline{1} \\ \cline{1-2} \end{array}$ *and* $\sigma_v = [1\overline{2}\overline{5}34] \leq \sigma_u = [\overline{4}\overline{1}235]$, *because σ_v has the reduced expression $s_5s_4s_3s_2s_3s_4s_5s_4s_3s_2$, which is a subword of the following reduced expression of σ_u: $s_4s_5s_4s_3s_2s_1s_2s_3s_4s_5s_4s_3s_2$.*

Chapter 4

Type C_n Demazure crystals, their opposite and cocrystals

In this chapter, given a partition λ with at most n parts, the type C_n crystal $\mathfrak{B}^\lambda$ of KN tableaux will be partitioned in two different ways: one into Demazure crystal atoms and the other into opposite Demazure crystal atoms. Motivated by Lascoux's double crystal graph construction in type A [22], and by Heo-Kwon work in [16] where Schützenberger *jeu de taquin* slides are used as crystal operators for $\mathfrak{sl}_2$, we define the cocrystal associated to a fixed KN tableau in the type C_n crystal $\mathfrak{B}^\lambda$. The main result of this section, Proposition 4.4.3, shows that given a KN tableau T, the cocrystal associated to T is isomorphic to a type A crystal of SSYT's with conjugated shape. This cocrystal is a type A crystal whose elements are type C objects, more precisely KN tableaux, and the crystal operators are described by SJDT slides. This cocrystal is useful in the next chapter, where SJDT is used to define right and left key maps.

4.1 Demazure crystal

Let λ be a partition and $v \in B_n\lambda$. Given a subset X of $\mathfrak{B}^\lambda$, consider the operator $\mathfrak{D}_i$ on X, with $i \in [n]$ defined by $\mathfrak{D}_i X = \{x \in \mathfrak{B}^\lambda \mid e_i^k(x) \in X \text{ for some } k \geq 0\}$ [9]. If $v = \sigma\lambda$ where $\sigma = s_{i_\ell} \cdots s_{i_1} \in B_n$ is a reduced word, we define the *Demazure crystal* $\mathfrak{B}_v$ to be

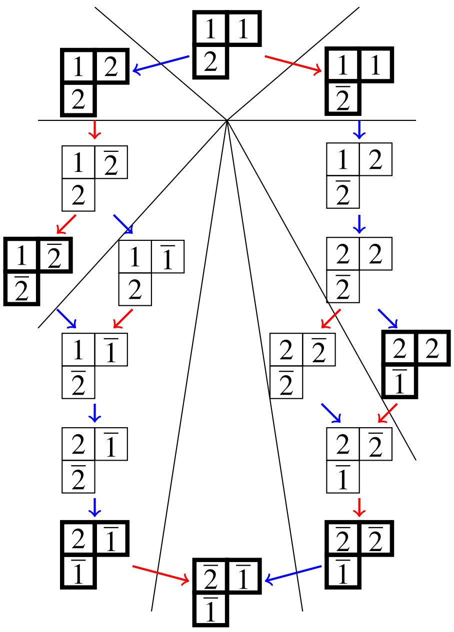
$$\mathfrak{B}_v = \mathfrak{D}_{i_\ell} \cdots \mathfrak{D}_{i_1} \{K(\lambda)\}.$$

This definition is independent of the reduced word for σ [9, Theorem 13.5]. In particular, when σ is the longest element fo B_n we recover B^λ. Also this definition is independent of the coset representative of σW_λ, that is, $\mathfrak{B}_{\sigma\lambda} = \mathfrak{B}_{\sigma_v\lambda}$. From [6, Proposition 2.4.4], σ uniquely factorizes as $\sigma_v\sigma'$ where $\sigma' \in W_\lambda$ and $\ell(\sigma) = \ell(\sigma_v) + \ell(\sigma')$. If $\sigma' = s_{j_{\ell(\sigma')}} \ldots s_{j_1} \in W_\lambda$ is a reduced word, then $\mathfrak{B}_{\sigma'\lambda} = \mathfrak{B}_\lambda = \mathfrak{D}_{j_{\ell(\sigma')}} \ldots \mathfrak{D}_{j_1}\{K(\lambda)\} = \{K(\lambda)\}$ and we may write $\mathfrak{B}_{\sigma\lambda} = \mathfrak{B}_v$.

From [6, Proposition 2.5.1], if $\rho \leq \sigma$ for the Bruhat order of B_n, then $u = \rho\lambda \leq v$. Since $e_i^0(x) = x$, if $\rho \leq \sigma$ then $\mathfrak{B}_u \subseteq \mathfrak{B}_v$. Thus we define the *Demazure crystal atom* $\widehat{\mathfrak{B}}_v$ to be

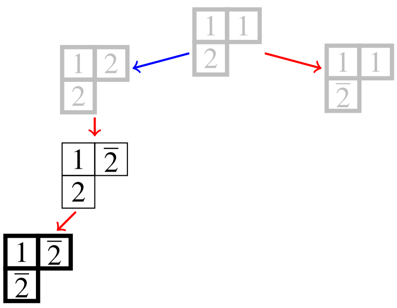
$$\widehat{\mathfrak{B}}_v = \mathfrak{B}_v \setminus \bigcup_{u<v} \mathfrak{B}_u = \mathfrak{B}_v \setminus \bigcup_{K(u)<K(v)} \mathfrak{B}_u.$$

Example 4.1.1. *Recall the type C_2 crystal graph from Example 2.5.3, associated to the partition $\lambda = (2,1)$:*



The crystal is split into $|B_2(2,1)| = 8$ parts, the number of elements of the B_2-orbit of $(2,1)$. Each part is a Demazure crystal atom and contains exactly one symplectic key tableau in $O(\lambda)$, the set of key tableaux with shape λ, drawn with a thick line, so we can identify each part with the weight of that key tableau, which is a vector in the B_2-orbit of $(2,1)$.

Now we see how to compute the Demazure crystal atom $\widehat{\mathfrak{B}}_{(1,\bar{2})}$:



To compute this Demazure crystal atom we start by computing the Demazure crystal $\mathfrak{B}_{(1,\bar{2})}$, which is formed by all tableaux on the left. Then, since $(2,1)$, $(1,2)$ and $(2,\bar{1})$ are smaller than $(1,\bar{2})$, we remove from the whole Demazure crystal three Demazure crystals contained in it: $\mathfrak{B}_{(2,1)}$, $\mathfrak{B}_{(1,2)}$ and $\mathfrak{B}_{(2,\bar{1})}$. The union of this sets is the greyed out section on the left.

Like in Example 4.1.1, every Demazure crystal atom contains exactly one key tableau (see Corollary 5.2.3). Hence we can define the right key map, a map sends each tableau of $\mathfrak{B}^\lambda$ to the unique key tableau living in the Demazure crystal atom that contains the given tableau. The right key of a tableau T is a key tableau of the same shape as T, entrywise "slightly" bigger than T. This is revisited in Chapter 5, where we define the right key map using the SJDT and see that both right key maps have the same output.

4.2 Opposite Demazure crystal

Let λ be a partition. Analogously to the previous case, we start by creating an opposing operator $\mathfrak{D}_i^{op}$ on X, with $i \in [n]$ defined by $\mathfrak{D}_i^{op} X = \{x \in \mathfrak{B}^\lambda \mid f_i^k(x) \in X \text{ for some } k \geq 0\}$. If $v = \sigma\lambda$ where

$\sigma = s_{i_\ell} \cdots s_{i_1} \in B_n$ is a reduced word, we define the *opposite Demazure crystal* $\mathfrak{B}^{op}_{-v}$ to be

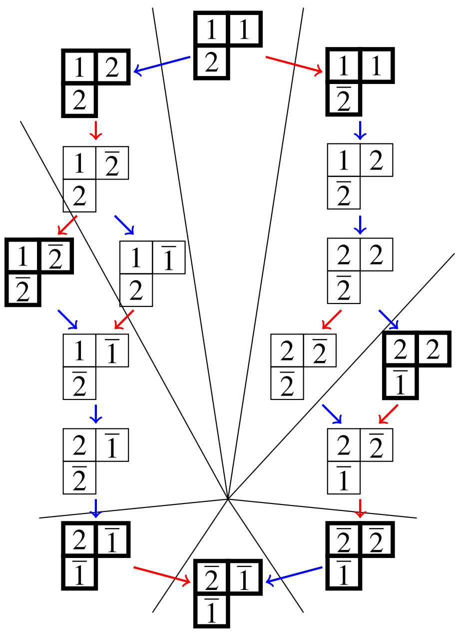
$$\mathfrak{B}^{op}_{-v} = \mathfrak{D}^{op}_{i_\ell} \cdots \mathfrak{D}^{op}_{i_1} \{K(-\lambda)\}.$$

We define the *opposite Demazure crystal atom* $\widehat{\mathfrak{B}}_v$ to be

$$\widehat{\mathfrak{B}}^{op}_{-v} = \mathfrak{B}^{op}_{-v} \setminus \bigcup_{-u>-v} \mathfrak{B}^{op}_{-u} = \mathfrak{B}^{op}_{-v} \setminus \bigcup_{K(-u)>K(-v)} \mathfrak{B}^{op}_{-u}.$$

The opposite Demazure crystal $\mathfrak{B}^{op}_{-v}$ is the image of $\mathfrak{B}_v$ by the Schützenberger-Lusztig involution (see Chapter 6. In particular, the tableau weights in $\mathfrak{B}_v$ and in $\mathfrak{B}^{op}_{-v}$ are symmetric.

Example 4.2.1. *The C_2 crystal graph $\mathfrak{B}^{(2,1)}$ can also be split into opposite Demazure crystal atoms:*



Similar to what we had in the previous section, every opposite Demazure crystal atom contains exactly one key tableau, as we shall see in the next chapter. So we can define the left key map, a map sends each tableau to the key tableau present in the opposite Demazure crystal atom that contains the given tableau. The left key of a tableau T is a key tableau of the same shape as T, entrywise "slightly" smaller than T. In Chapter 5 we will compute the left key map using SJDT.

4.3 Demazure characters and opposite Demazure characters

The Demazure character, or key polinomial, κ_v for $v \in B_n\lambda$, is the character of the Demazure crystal $\mathfrak{B}_v$:

$$\kappa_v(x_1, \ldots, x_n) = \sum_{T \in \mathfrak{B}_v} x^{\mathrm{wt}T}.$$

We also define the Demazure atom $\widehat{\kappa}_v$ as the character of the Demazure crystal atom $\widehat{\mathfrak{B}}_v$:

$$\widehat{\kappa}_v(x_1,\dots,x_n) = \sum_{T\in\widehat{\mathfrak{B}}_v} x^{\mathrm{wt}T}.$$

Analogously, we can define opposite Demazure characters and opposite Demazure atoms:

$$\kappa^{op}_{-v}(x_1,\dots,x_n) = \sum_{T\in\mathfrak{B}^{op}_{-v}} x^{\mathrm{wt}T};\ \widehat{\kappa}^{op}_{-v}(x_1,\dots,x_n) = \sum_{T\in\widehat{\mathfrak{B}}^{op}_{-v}} x^{\mathrm{wt}T}.$$

Since the tableau weights in $\mathfrak{B}_v$ and in $\mathfrak{B}^{op}_{-v}$ are symmetric, we have the following result:

Corollary 4.3.1.

$$\kappa_v(x_1,\dots,x_n) = \kappa^{op}_{-v}(x_1^{-1},\dots,x_n^{-1})$$

As a consequence, for instance, the type C_n Fu-Lascoux non-symmetric Cauchy kernel, given in [12], can be written as:

$$\begin{aligned}\frac{\prod_{1\le i<j\le n}(1-x_ix_j)}{\prod_{i,j=1}^n(1-x_iy_j)\prod_{i,j=1}^n(1-x_i/y_j)} &= \sum_{v\in\mathbb{N}^n}\widehat{\kappa}_v(x_1,\dots,x_n)\kappa_{-v}(y_1,\dots,y_n)\\ &= \sum_{v\in\mathbb{N}^n}\widehat{\kappa}_v(x_1,\dots,x_n)\kappa^{op}_v(y_1^{-1},\dots,y_n^{-1})\end{aligned}$$

Remark 4.3.2. *In [12], Fu-Lascoux also presented, and proved algebraically, type A_{n-1} Fu-Lascoux non-symmetric Cauchy kernel. For this identity there are three combinatorial knowns proofs: one by Lascoux, in [22], a second one by Azenhas-Emami, in [3], and the most recent one, by Choi-Kwon, in [10]. In [10], Choi-Kwon, working in the Lakshmibai-Seshadri paths, started by manipulating the identity using opposite Demazure characters.*

4.4 Cocrystals

In this section we start working with SSYT's and type A crystals, and we only address KN tableaux in the last subsection.

4.4.1 Dual RSK correspondence

Let T be a $T\in\mathcal{SSYT}(\lambda,n)$ with column decomposition $T=C_1C_2\cdots C_k$, and recall the column insertion for SSYT's from Section 2.3.

Given $r\ge 1$, let E^r_n be the set of biwords without repeated biletters, in lexicographic order, $\begin{pmatrix}u\\v\end{pmatrix}\le\begin{pmatrix}u'\\v'\end{pmatrix}$ if $u<u'$ or if $u=u'$ and $v\le v'$, with the bottom word on the alphabet $[n]$, and the top word on the alphabet $[r]$. The set E^r_n can also be thought as the set of sequences of r columns, possibly some of them empty, on the alphabet $[n]$, where each pair of consecutive columns has maximum overlapping, and, in the case of two non-empty columns whose intermediate columns are empty, the top edge of the left column and the bottom edge of the right column are aligned. In particular, E^r_n has

a subset identified with $\mathcal{SSYT}(\lambda, n)$, such that $\ell(\lambda') \leq r$, where $\ell(\lambda')$ is the length of λ', the conjugate partition of λ'. Given a tableau $T \in \mathcal{SSYT}(\lambda, n)$, we create a biword, without repeated biletters, whose bottom word is $cr(T)$ and in the top word we register in which column of T, counted from the right, was each letter of cr(T) read. Each biword will be an element of E_n^r, where $\ell(\lambda') \leq r$. For instance, if $T = \begin{array}{|c|c|c|} \hline 1 & 2 & 2 \\ \hline 2 & 3 \\ \cline{1-2} 4 & 4 \\ \cline{1-2} \end{array} \in \mathcal{SSYT}((3,2,2,0),4)$, the biword of T is

$$w = \begin{pmatrix} 1 & 2 & 2 & 2 & 3 & 3 & 3 \\ 2 & 2 & 3 & 4 & 1 & 2 & 4 \end{pmatrix} \in E_4^3, \text{ with } \ell(\lambda') = 3$$

.

The (type A) dual RSK, RSK^*, is a bijection [13, Section A.4.3] between E_n^r and pairs of SSYT's of conjugate shapes and lengths $\leq n$ and $\leq r$, respectively:

$$\begin{aligned} RSK^* : E_n^r &\to \bigsqcup_{\substack{\ell(\lambda) \leq n \\ \ell(\lambda') \leq r}} \mathcal{SSYT}(\lambda, n) \times \mathcal{SSYT}(\lambda', r) = \bigsqcup_{\substack{\ell(\lambda) \leq n \\ \ell(\lambda') \leq r \\ P \in \mathcal{SSYT}(\lambda, n)}} \{P\} \times \mathcal{SSYT}(\lambda', r) \\ w &\mapsto (P, Q). \end{aligned}$$

The bijection can be calculated in the following way:

Let $w = \begin{pmatrix} x_1 & x_2 & \dots & x_m \\ y_1 & y_2 & \dots & y_m \end{pmatrix}$. Then start with $i = 1$, $P = Q$ are empty tableaux.

1. Column insert y_i into P.
2. Add one cell to Q whose entry is x_i, in a position such that P and Q, with this new cell, have conjugate shapes.
3. If $i \neq m$, then $i := i + 1$ and return to (1). Else the algorithm is finished.

Given a biword w, the first and second components of $RSK^*(w)$ are the P- symbol and the Q-symbol of w.

For instance, the biword $w = \begin{pmatrix} 1 & 2 & 2 & 2 & 3 & 3 & 3 \\ 2 & 2 & 3 & 4 & 1 & 2 & 4 \end{pmatrix}$ of $T = \begin{array}{|c|c|c|} \hline 1 & 2 & 2 \\ \hline 2 & 3 \\ \cline{1-2} 4 & 4 \\ \cline{1-2} \end{array}$, is mapped to the pair

$$\left(T = \begin{array}{|c|c|c|} \hline 1 & 2 & 2 \\ \hline 2 & 3 \\ \cline{1-2} 4 & 4 \\ \cline{1-2} \end{array}, K(rev((3,2,2)')) = \begin{array}{|c|c|c|} \hline 1 & 2 & 2 \\ \hline 2 & 3 & 3 \\ \hline 3 \\ \cline{1-1} \end{array} \right).$$

More generally, given $T \in \mathcal{SSYT}(\lambda, n)$ with $\ell(\lambda') \leq r$, the dual RSK maps its biword, w, to the pair $(T, K(rev(\lambda')))$, where $rev(\lambda')$ is the vector λ' written backwards. Note that the weight of $K(\lambda')$ registers the column lengths of T, from right to left.

We also can compute RSK^* of a biword obtained from a skew SSYT. For instance, let $\widetilde{T}$ be the skew SSYT $\begin{array}{|c|c|c|} \cline{3-3} \multicolumn{2}{c|}{} & 2 \\ \hline 1 & 2 & 3 \\ \hline 2 & 4 \\ \cline{1-2} 4 \\ \cline{1-1} \end{array}$. Its biword is

$$\widetilde{w} = \begin{pmatrix} 1 & 1 & 2 & 2 & 3 & 3 & 3 \\ 2 & 3 & 2 & 4 & 1 & 2 & 4 \end{pmatrix}.$$

Finally,

$$RSK^*(\widetilde{w}) = \left(T = \mathrm{rect}(\widetilde{T}), \widetilde{Q} = \begin{array}{|c|c|c|} \hline 1 & 1 & 2 \\ \hline 2 & 3 & 3 \\ \hline 3 \\ \cline{1-1} \end{array} \right),$$

where $\mathrm{rect}(\widetilde{T})$ is the rectification of $\widetilde{T}$ via SJDT. The weight of $\widetilde{Q}$ records the column lengths of $\widetilde{T}$ from right to left.

4.4.2 Cocrystal of SSYT's

Given $T \in \mathcal{SSYT}(\lambda, n)$ with $\ell(\lambda') \leq r$, we define the *cocrystal* of T, $\mathfrak{CB}^{\lambda'}(T)$, to be the $\mathfrak{gl}_r$-crystal,

$$\mathfrak{CB}^{\lambda'}(T) = (\mathrm{RSK}^*)^{-1}(\{T\} \times \mathcal{SSYT}(\lambda', r)), \tag{4.1}$$

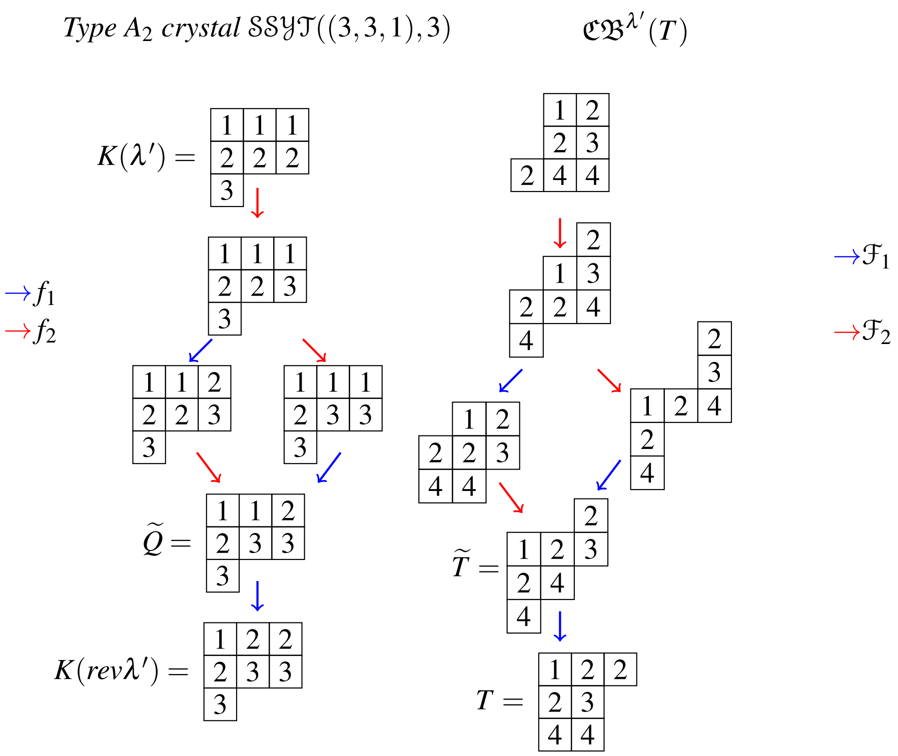
whose crystal operators, lowering $\mathcal{F}_i$ and raising $\mathcal{E}_i$, are SJDT slides on consecutive columns i, $i+1$ of T, for $i = 1, \dots, r-1$. More precisely, $\mathcal{F}_i$ sends a cell from the i-th column to the $i+1$-th column, counting from right to left. The lowest weight element of $\mathfrak{CB}^{\lambda'}(T)$ is T, and the highest weight element is the anti rectification of T, that is, the rectification is performed south-eastward. The type A_{r-1} crystals $\mathcal{SSYT}(\lambda', r)$ and $\mathfrak{CB}^{\lambda'}(T)$ are isomorphic. This isomorphism relies on the following proposition, a consequence of [16, Lemma 2.3, Lemma 2.4] by Heo-Kwon:

Proposition 4.4.1. *Let T be a skew SSYT. The Q-symbol of $\mathcal{F}_i(T)$ is the same as f_i applied to the Q-symbol of T, and the weight of the Q-symbol of T records the column lengths of T from right to left.*

Example 4.4.2. *Recall T and $\widetilde{T}$ from the previous subsection. Note that $T = \mathcal{F}_i(\widetilde{T})$ and that the Q-symbols obtained from both tableaux are connected via f_i, that is, $\widetilde{Q} = f_1(K(rev((3,2,2)')))$.*

This can be easily seen in the next crystal graphs. On the right, we have the cocrystal $\mathfrak{CB}^{\lambda'}(T)$, whose vertices are obtained by applying the elementary SJDT slides $\mathcal{E}_i$, for $i = 1, 2$, on T, the lowest weight element of the cocrystal $\mathfrak{CB}^{\lambda'}(T)$. Namely, $\mathcal{E}_1$ sends an entry from the second column to the first column, and $\mathcal{E}_2$ sends an entry from the third column to the second column, where we count columns starting from the right. $\mathcal{F}_1$ and $\mathcal{F}_2$ are the inverse operations.

On the left, we have the type A_2 crystal $\mathcal{SSYT}((3,3,1),3)$, formed by the Q-symbols of every skew tableau that exists in the type A_2 crystal $\mathfrak{CB}^{\lambda'}(T)$ on the right. The type A_2 crystal operators on the left are defined by the signature rule on the alphabet [3], whereas, on the right, $\mathcal{F}_1$ and $\mathcal{F}_2$ are type A_2 crystal operators defined by SJDT.



The type A_2 crystal operators f_1 and f_2 are given by the signature rule on the alphabet [3], *whereas $\mathcal{F}_1$ and $\mathcal{F}_2$, even though they are also type A_2 crystal operators, are defined by SJDT.*

4.4.3 Cocrystal of KN tableaux

Let $T \in \mathcal{SSYT}(\lambda, n)$. Note that T, the lowest weight of the cocrystal $\mathfrak{CB}^{\lambda'}(T)$, is also in the type C_n crystal $\mathfrak{B}^{\lambda}$ (recall that $\mathcal{SSYT}(\lambda, n)$ is a crystal contained in $\mathfrak{B}^{\lambda}$). Fixed an arbitrary tableau Y in the crystal $\mathfrak{B}^{\lambda}$, there is a sequence S, of type C_n crystal operators of $\mathfrak{B}^{\lambda}$, such that $S(T) = Y$. All elements of the cocrystal $\mathfrak{CB}^{\lambda'}(T)$ are SJDT related and we can apply this sequence S to all skew tableaux on the cocrystal, obtaining, for each skew tableau, a new skew tableau of the same shape. All these skew tableaux, obtained by application of the sequence S to each element of $\mathfrak{CB}^{\lambda'}(T)$, will be connected via SJDT, because the SJDT and the crystal operators of $\mathfrak{B}^{\lambda}$ commute [25, Theorem 6.3.8], hence they are the elements of a new cocrystal $\mathfrak{CB}^{\lambda'}(S(T))$ of type A_{r-1}, despite the possibility that its vertices are type C_n objects (i.e. KN skew tableaux). Recalling that the weight function of $\mathfrak{CB}^{\lambda'}(T)$ is given by the column lengths of each vertex, from right to left, which is preserved by any sequence S of crystal operators given by the C_n signature rule in $\mathfrak{B}^{\lambda}$, the following is a consequence of Proposition 4.4.1.

Proposition 4.4.3. *Given $T \in \mathcal{KN}(\lambda, n)$, with $\ell(\lambda') \geq r$, the cocrystal $\mathfrak{CB}^{\lambda'}(T)$ with lowest weight element T, obtained from T by successive application of elementary SJDT moves, is crystal isomorphic to the $\mathfrak{gl}_r$-crystal $\mathcal{SSYT}(\lambda, r)$.*

Fulton [13] has proved the following result for type A_{n-1} SSYT's.

Proposition 4.4.4. *[13, Proposition 7, Corollary 1, Appendix A.5] Given $T \in \mathcal{SSYT}(\lambda, n)$ and a skew shape whose column lengths are a permutation of λ, the column lengths of T, there is exactly one skew tableau with that shape that rectifies to T. Furthermore, the last and first columns only depend on their lengths.*

This means that given $T \in \mathcal{SSYT}(\lambda, n)$, the cocrystal $\mathfrak{CB}^{\lambda'}(T)$ attached to $T \in \mathcal{SSYT}(\lambda, n)$ has a distinguished set of skew tableaux whose column lengths are a permutation of λ', the column lengths of T. The skew shapes of these distinguished vertices are preserved by any sequence S of type C_n crystal operators of the crystal $\mathfrak{B}^{\lambda}$. Thus we obtain another proof of our Proposition 40 and Corollary 41 of [36] which is an extension of Proposition 4.4.4 to KN tableaux.

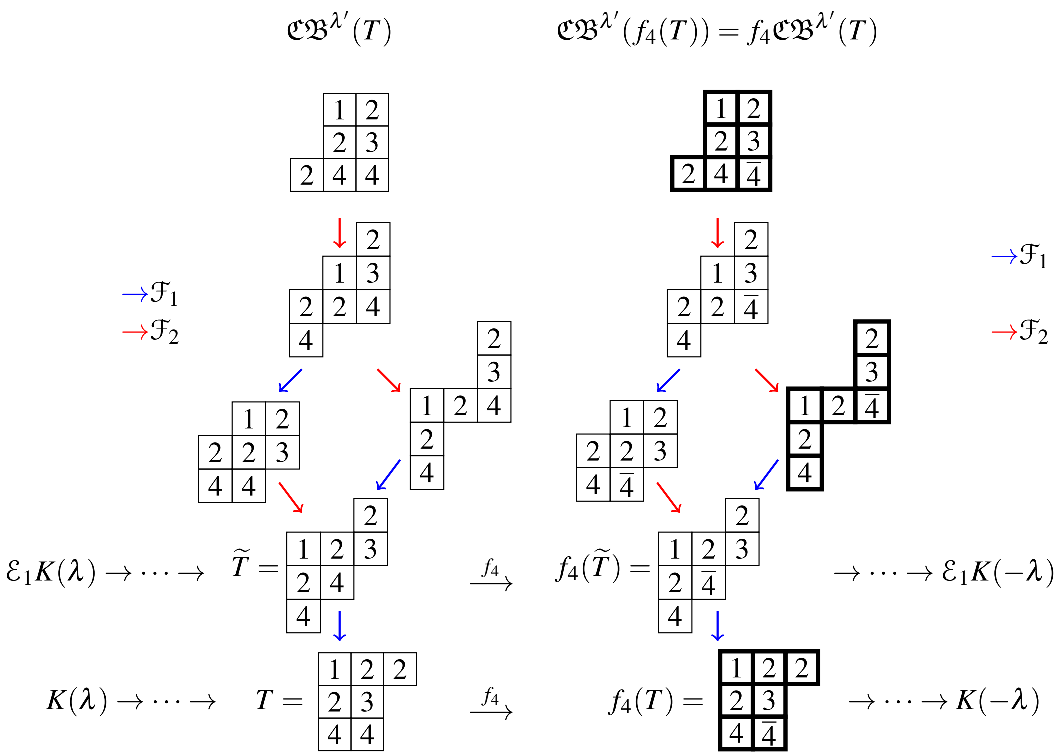
Proposition 4.4.5. *[34, 36, Proposition 40, Corollary 41] Given $T \in \mathcal{KN}(\lambda, n)$ and a skew shape whose column lengths are a permutation of the column lengths of T, there is exactly one skew tableau with that shape that rectifies to T. Furthermore, the last and first columns only depend on their lengths.*

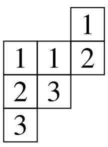
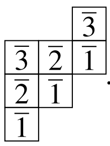
A key tableau in the type A_{r-1} crystal $\mathcal{SSYT}(\lambda', r)$ is a tableau of shape λ' whose weight is in $\mathfrak{S}_n\lambda'$, the $\mathfrak{S}_r$-orbit of λ'. For each element of $\mathfrak{S}_r\lambda'$ there is exactly one key tableau of shape λ' with that weight. More precisely the key tableaux in $\mathcal{SSYT}(\lambda', r)$ are distinguished vertices and define the set $\mathfrak{S}_r K(\lambda')$ where $s_i K(\lambda') = K(s_i\lambda')$ and s_i, for $i = 1, \ldots, r-1$, are the simple transpositions of $\mathfrak{S}_r$. Thereby it is natural to define keys in a cocrystal.

Definition 4.4.6. *Given $T \in \mathcal{KN}(\lambda, n)$, with $\ell(\lambda') \leq r$, and $X \in \mathfrak{CB}^{\lambda'}(T)$, X is said to be a key (skew tableau) of $\mathfrak{CB}^{\lambda'}(T)$ if its weight as an element of the said cocrystal, the sequence column lengths of X, from right to left, is a permutation of the weight of T as an element of the same cocrystal.*

In other words, the keys of $\mathfrak{CB}^{\lambda'}(T)$ are the image of the keys in $\mathcal{SSYT}(\lambda', r)$ via the crystal isomorphism between both crystals. We then have an action of $\mathfrak{S}_r$ on set of keys of $\mathfrak{CB}^{\lambda'}(T)$.

Example 4.4.7. *Recall the right hand side crystal from Example 4.4.2. T is in the type C_4 crystal $\mathfrak{B}^{(3,2,2,0)}$. Hence we can apply to each vertex of $\mathfrak{CB}^{\lambda'}(T)$ the sequence of crystal operators $S = f_4$, obtaining a new cocrystal, on the right, whose vertices are KN skew tableaux connected via SJDT. This cocrystal $\mathfrak{CB}^{\lambda'}(f_4(T))$ is a type A_2 crystal.*



The bold skew tableaux in the cocrystal on the right are its key (skew) tableaux. The KN tableaux T and $f_4(T)$ are contained in a type C_4 crystal with highest weight element $K(\lambda)$ and lowest weight element $K(-\lambda)$. The KN skew tableaux in a same position of the cocrystal define a type C_4 crystal isomorphic to the crystal $\mathfrak{B}^{\lambda}$. In fact, their highest weight are the Littlewood-Richardson tableaux [13] of weight λ, defining the cocrystal attached to $K(\lambda)$, the Yamanouchi tableau of weight and shape λ. For instance, the type C_4 crystal containing $\widetilde{T}$ and $f_4(\widetilde{T})$ has highest weight element the Littlewood-Richardson tableau $\mathcal{E}_1(K(\lambda)) =$  *and lowest weight element its symmetric, in the sense of Lusztig involution (see Chapter 6), $\mathcal{E}_1(K(-\lambda)) =$* .

In the next chapter, we compute the right and left keys of a tableau using the SJDT. The cocrystal $\mathfrak{CB}^{\lambda'}(T)$ contains all the needed information to compute both keys.

Chapter 5

Right and left keys and Demazure atoms in type C_n

Frank words were introduced in type A by Lascoux and Schützenberger in [24]. In this chapter, we define type C_n frank words on the alphabet $[\pm n]$ and use them to create the right and left key maps, that send KN tableaux to key tableaux in type C_n. Lascoux and Schützenberger constructed a right key map, via *jeu de taquin*, that provided a description for type A_{n-1} Demazure crystal atoms [24, Theorem 3.8]. The main result of this section, Theorem 5.2.6, shows that the right key map defined in this chapter, via SJDT, provides a description for type C_n Demazure crystal atoms. We finish this chapter with a way of computing right and left keys that does not require the use of SJDT.

5.1 Frank words in type C_n

We start by defining frank words in the alphabet $[\pm n]$. Given a ordered alphabet and a word on that alphabet, a column of the word is a maximal factor whose letters are strictly increasing. Hence, we can decompose a word into columns, and such decomposition is unique.

Definition 5.1.1. *Let w be word on the alphabet $[\pm n]$. We say that w is a type C_n* frank word *if the lengths of its columns form a multiset equal to the multiset formed by the lengths of the columns of the tableau $P(w)$, the Baker-Lecouvey insertion of w.*

Example 5.1.2. *In Example 2.3.2 we have that* $P(23\bar{2}\bar{3}1) = P(\bar{1}113\bar{3}) = \begin{array}{|c|c|c|}\hline 1 & 1 & \bar{1} \\ \hline 3 \\ \cline{1-1} \bar{3} \\ \cline{1-1}\end{array}$ *. Since* $23\bar{2}\bar{3}1$ *and* $\bar{1}113\bar{3}$ *have one column of length* 3 *and two columns of length* 1*, they are frank words.*

Given a frank word w, the number of letters of w is the same as the number of cells of $P(w)$, hence the case 3 of the Baker-Lecouvey insertion does not happen.

Proposition 5.1.3. *Let w be frank word on the alphabet $[\pm n]$. All columns of w are admissible.*

Proof. Suppose that the statement is false. So there is a factor of w that is a non-admissible column with all of its proper factors admissible. Hence we can apply the Knuth relation $K5$, meaning that

w is Knuth related to a smaller word w'. But in this case, the number of letters of w' is less then the number of cells of $P(w) = P(w')$, which is a contradiction. □

Fixed a KN tableau T, consider the set of all possible last columns taken from skew tableaux SJDT connected to T and with same number of columns of each length as T. Proposition 4.4.5 implies that for each permutation of the columns lengths of T, there is a skew tableau with that sequence of column lengths that rectifies to T. In particular, it implies that for every column length of T, there is a skew tableau SJDT connected to T with last column having that length. Proposition 4.4.5 also implies that this set of last columns has exactly one element for each distinct column length of T. For every column C in this set, consider the columns rC, its right column. The next proposition implies that this set of right columns is nested, if we see each column as the set formed by its elements.

Proposition 5.1.4. *Consider T a two-column KN skew tableau C_1C_2 with a puncture in the first column. Slide that cell once via SJDT, obtaining a punctured two-column KN skew tableau $C_1'C_2'$. Then $rC_2' \subseteq rC_2$.*

Proof. If the sliding was vertical then $C_2' = C_2$, hence $rC_2' = rC_2$. If the sliding was horizontal, Let β be the number on the cell right of the puncture on $spl(T)$. Recall Φ, the function that takes an admissible column to the associated coadmissible column.

If $\beta = b$ is unbarred then $C_2' = \Phi^{-1}(\Phi(C_2) \setminus \{b\} \sqcup \{*\})$. In this case $\Phi(C_2') = \Phi(C_2) \setminus \{b\} \sqcup \{*\}$, hence rC_2 and rC_2' have the same barred part. Consider $z_1 < \cdots < z_\ell$ the unbarred letters that appear on C_2 and not on $\Phi(C_2)$. When we take b from $\Phi(C_2)$, if $\overline{b} \in \Phi(C_2)$ our set of letters $z_1 < \cdots < z_\ell$ will lose an element, giving the inclusion of the unbarred part of C_2' in C_2; if $\overline{b} \notin \Phi(C_2)$, then $b \in C_2$ and in C_2' the least $z_i > b$ may reduce to b, and subsequent z_j may reduce to z_{j-1}. Hence we have the inclusion of the unbarred part of C_2' in C_2.

If $\beta = \overline{b}$ is barred then $C_2' = C_2 \setminus \{\overline{b}\} \sqcup \{*\}$. In this case rC_2 and rC_2' have the same unbarred part. Consider $\overline{t_1} > \cdots > \overline{t_\ell}$ the barred letters that appear on $\Phi(C_2)$ and not on C_2. When we take $\overline{b}$ from C_2, if $b \in C_2$ our set of $\overline{t_1} > \cdots > \overline{t_\ell}$ letters will lose an element, giving the inclusion of the barred part of rC_2' in rC_2; if $b \notin C_2$, then $\overline{b} \in \Phi(C_2)$ and in C_2' the least $\overline{z_i} > \overline{b}$ may reduce to $\overline{b}$, and subsequent bigger $\overline{z_j}$'s may reduce to $\overline{z_{j+1}}$. Hence we have the inclusion of the barred part of $\Phi(C_2')$ in $\Phi(C_2)$. □

This proposition allows us to define a map that sends a KN tableau to a key tableau in type C_n, called the (symplectic) right key map of a given KN tableau. We shall see in the next chapter that this right key map has the same output as the right key map defined in the previous chapter.

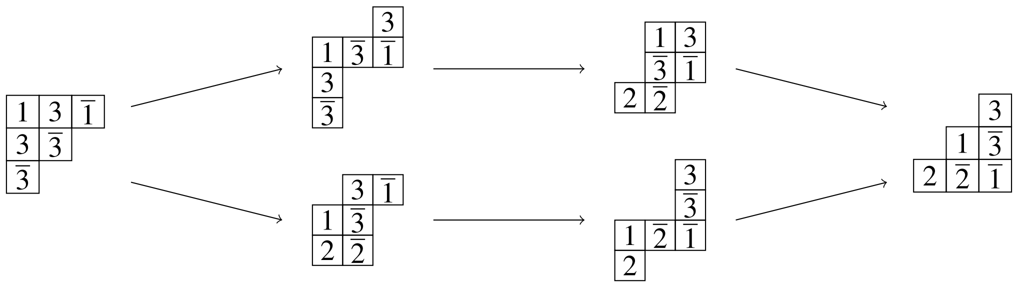
Theorem 5.1.5 (Right key map)**.** *Given a KN tableau T, we can replace each column with a column of the same size taken from the right columns of the last columns of all skew tableaux associated to it. The tableau obtained is a key tableau. We call this tableau the right key tableau of T and denote it by $K_+(T)$.*

Proof. The previous proposition implies that the columns of $K_+(T)$ are nested and do not have symmetric entries. So, it is indeed a KN key tableau. □

Remark 5.1.6. *Recall the set up of Proposition 4.4.5. If the shape of S, μ/ν, is such that every two consecutive columns have at least one cell in the same row, then each column of S is a column of the* *word $cr(S)$, hence $cr(S)$ is a frank word. Moreover, the columns of S appear in reverse order in $cr(S)$. Therefore, given a KN tableau T, the columns of $K_+(T)$ can be also found as the right columns of the first columns of frank words associated to T.*

If T is a SSYT then this right key map coincides with the one defined by Lascoux and Schützenberger in [24].

Example 5.1.7. *The tableau $T = \begin{array}{|c|c|c|}\hline 1 & 3 & \bar{1} \\ \hline 3 & \bar{3} \\ \cline{1-2} \bar{3} \\ \cline{1-1}\end{array}$ gives rise to the cocrystal $\mathfrak{CB}^{\lambda'}(T)$, with $\lambda = (3,2,1)$. The following are the vertices of $\mathfrak{CB}^{\lambda'}(T)$ consisting of the six KN skew tableaux with the same number of columns of each length as T, each one corresponding to a permutation of its column lengths.*



The right key tableau associated to T has as columns $r\begin{array}{|c|}\hline 3 \\ \hline \bar{3} \\ \hline \bar{1} \\ \hline\end{array}$, $r\begin{array}{|c|}\hline 3 \\ \hline \bar{1} \\ \hline\end{array}$ and $r\begin{array}{|c|}\hline \bar{1} \\ \hline\end{array}$. Hence $K_+(T) = \begin{array}{|c|c|c|}\hline 3 & 3 & \bar{1} \\ \hline \bar{2} & \bar{1} \\ \cline{1-2} \bar{1} \\ \cline{1-1}\end{array}$.

Remark 5.1.8. *Proposition 4.4.5 shows that the action defined by the SJDT on two consecutive columns of a straight KN tableau T of shape λ gives rise to a permutohedron where the vertices are all the KN skew tableaux in the Knuth class of T whose column length sequence is a permutation of the column length sequence of T [24]. For instance, in Example 5.1.7 we have a permutohedron (hexagon) for $\mathfrak{S}_3$. In fact, these vertices are the key (skew) tableaux of the cocrystal $\mathfrak{CB}^{(3,2,1)'}(T)$. Hence the cocrystal $\mathfrak{CB}^{(3,2,1)'}(T)$ contains all the needed information to compute the right key of T.*

In the same spirit of the right key, we define the left key of a KN tableau. Just like in Proposition 5.1.4, we can prove that the slides of the SJDT are effectively adding an entry to ℓC_1, i.e. $\ell C_1 \subseteq \ell C_1'$, hence the left columns of the first columns of all skew tableaux with the same number of columns of each length as T will be nested.

So, if we replace each column of T with a column of the same size taken from the left columns of the first columns of all skew tableaux associated to it we obtain the left key $K_-(T)$.

Example 5.1.9. *In Example 5.1.7 we have that the left key of $T = \begin{array}{|c|c|c|}\hline 1 & 3 & \bar{1} \\ \hline 3 & \bar{3} \\ \cline{1-2} \bar{3} \\ \cline{1-1}\end{array}$ has as columns $\ell\begin{array}{|c|}\hline 1 \\ \hline 3 \\ \hline \bar{3} \\ \hline\end{array}$, $\ell\begin{array}{|c|}\hline 1 \\ \hline 2 \\ \hline\end{array}$ and $\ell\begin{array}{|c|}\hline 2 \\ \hline\end{array}$. Hence $K_-(T) = \begin{array}{|c|c|c|}\hline 1 & 1 & 2 \\ \hline 2 & 2 \\ \cline{1-2} \bar{3} \\ \cline{1-1}\end{array}$. So, the cocrystal $\mathfrak{CB}^{(3,2,1)'}(T)$ also contains all the needed information to compute the left key of T.*

5.2 Demazure crystals and right key tableaux

Let $\lambda \in \mathbb{Z}^n$ be a partition and $v \in B_n\lambda$. We define

$$\mathfrak{U}(v) = \{T \in \mathcal{KN}(\lambda, n) \mid K_+(T) = K(v)\}$$

the set of KN tableaux of B^λ with right key $K(v)$.

In the following lemma we identify how does the crystal operators affect the weight of a column:

Lemma 5.2.1. *Let $\sigma = s_i$ be a generator of B_n and C an admissible column such that $f_i(C) \neq 0$. Then* $\mathrm{wt}(rC) = \mathrm{wt}(r(f_i(C)))$ *or* $\mathrm{wt}(rC) = \sigma(\mathrm{wt}(r(f_i(C))))$.

Proof. Let $i = n$. We can apply f_i to C if and only $n \in C$ and $\overline{n} \notin C$. In this case $n \in rC$ and after applying f_i we have $n \notin C$ and $\overline{n} \in C$, hence $\overline{n} \in rC$. So $\mathrm{wt}(rC) = s_n(\mathrm{wt}(r(f_n(C))))$.

Let $i < n$. We can apply f_i to C, so we have 6 cases to study:

1. $i \in C, i+1, \overline{i+1}, \overline{i} \notin C$: In this case we have that $i+1 \in f_i(C), i, \overline{i+1}, \overline{i} \notin f_i(C)$. Note that $\overline{i} \notin rC$ and $\overline{i+1} \notin r(f_i(C))$. If $\overline{i+1} \notin rC$ then $\overline{i} \notin r(f_i(C))$, hence f_i swaps the weight of i and $i+1$ from $(1,0)$ to $(0,1)$, respectively. If $\overline{i+1} \in rC$ then $\overline{i} \in r(f_i(C))$, hence f_i swaps the weight of i and $i+1$ from $(1,-1)$ to $(-1,1)$.

2. $i, \overline{i+1} \in C$, $i+1, \overline{i} \notin C$: In this case we have that $i+1, \overline{i+1} \in f_i(C)$, $i, \overline{i} \notin f_i(C)$. Note that $i, \overline{i+1} \in rC, i+1, , \overline{i} \notin rC$ and that $i+1, \overline{i} \in r(f_i(C)), i, \overline{i+1} \notin r(f_i(C))$, and all other appearances in rC are intact. Hence f_i swaps the weight of i and $i+1$ from $(1,-1)$ to $(-1,1)$.

3. $i+1, \overline{i+1} \in C$, $i, \overline{i} \notin C$: In this case we have that $i+1, \overline{i} \in f_i(C)$, $i, \overline{i+1} \notin f_i(C)$. Note that $i+1, \overline{i} \in rC, i, \overline{i+1} \notin rC$ and that $i+1, \overline{i} \in r(f_i(C)), i, \overline{i+1} \notin r(f_i(C))$, and all other appearances in rC are intact. Hence f_i did nothing to weight of rC.

4. $i, i+1, \overline{i+1} \in C$, $\overline{i} \notin C$: In this case we have that $i, i+1, \overline{i} \in f_i(C)$, $\overline{i+1} \notin f_i(C)$. Note that $i, i+1 \in rC, \overline{i+1}, \overline{i} \notin rC$ and that $i, i+1 \in r(f_i(C)), \overline{i+1}, \overline{i} \notin r(f_i(C))$, and all other appearances in rC are intact. Hence f_i did nothing to weight of rC.

5. $i, \overline{i+1}, \overline{i} \in C$, $i+1 \notin C$: In this case we have that $i+1, \overline{i+1}, \overline{i} \in f_i(C)$, $i \notin f_i(C)$. Note that $i, \overline{i+1} \in rC, i+1, \overline{i} \notin rC$ and that $i+1, \overline{i} \in r(f_i(C)), i, \overline{i+1} \notin r(f_i(C))$, and all other appearances in rC are intact. Hence f_i swaps the weight of i and $i+1$ from $(1,-1)$ to $(-1,1)$.

6. $\overline{i+1} \in C$, $i, i+1, \overline{i} \notin C$: In this case we have that $\overline{i} \in f_i(C)$, $i, i+1, \overline{i+1} \notin f_i(C)$. Note that $i, i+1 \notin rC$ and $\overline{i+1} \in rC$. If $\overline{i} \in rC$ then we have $i, i+1 \notin r(f_i(C))$ and $\overline{i+1}, \overline{i} \in r(f_i(C))$, so f_i did nothing to weight of rC. If $\overline{i} \notin rC$ then $\overline{i+1} \notin r(f_i(C))$ and $\overline{i} \in r(f_i(C))$, hence f_i swaps the weight of i and $i+1$ from $(0,-1)$ to $(-1,0)$.

□

Remark 5.2.2. *All the cases where the weight is preserved happen to have equal weight for i or $i+1$ in rC or we are in a column C in which we can also apply e_i. If the weights for i and $i+1$ in rC swap, then if rC the weight of i is bigger (in the usual ordering) then the weight of $i+1$.*

The following corollary shows that every Demazure crystal atom defined in the previous chapter has exactly one key tableau:

Corollary 5.2.3. *Let T be a KN tableau and $i \in [n]$. If $K_+(T) = K(v)$, for some $v = (v_1, \dots, v_n) \in \mathbb{Z}^n$, then $K_+(f_i(T)) = K(v)$ or $K_+(f_i(T)) = K(s_i v)$. Moreover, $K_+(f_i(T)) = K(s_1 v)$ only if $v_i > v_{i+1}$ (in the usual ordering of real numbers) and $1 \le i < n$, or, $v_i > 0$ and $i = n$.*

Proof. Consider a multiset of frank words F such that the multiset of length of their first columns is the same of the multiset of lengths of columns of T.

If $K_+(f_i(T)) = K_+(T)$ then we are done. Else there are two cases: $1 \le i < n$ and $i = n$.

Consider $1 \le i < n$. Since there is a change in the weight of the key tableau, we have that in at least one first column of words in F weight of i is bigger or equal than the weight of $i+1$. These first columns form a nested set without symmetric entries, hence in all first column of words in F weight of i is bigger or equal than the weight of $i+1$.

Let A be the subset of F such that the weight of i and $i+1$ in the right column of its first column is different and does not swap when we apply f_i to the frank word. Consider (a,b) the sum of weights of i and $i+1$, respectively, of all right columns of first columns of words in A, and (c,d) defined analogously to $F \setminus A$.

The weights of i and $i+1$ in $K_+(T)$ is $(a,b)+(c,d) = (a+c, b+d)$ and the weights of i and $i+1$ in $K_+(f_i(T))$ is $(a,b)+(d,c) = (a+d, b+c)$, and note that $(a+c, b+d) \in B_2(a+d, b+c)$, because f_i doesn't change any other weight (Lemma 5.2.1).

Since in all first columns of F weight of i is bigger or equal than the weight of $i+1$, $a \ge 0$ and $b \le 0$, and they are equal when $A = \emptyset$, so $(a+c, b+d) = s_1(a+d, b+c)$, hence $\mathrm{wt}(K_+(f_i(T))) = s_i v$. Hence we assume $a \ne b$. If $c = d$ we have $\mathrm{wt}(K_+(f_i(T))) = v$, hence $K_+(f_i(T)) = K(v) = K_+(T)$, which is a contradiction.

This implies that $(a+c, b+d) = \sigma(a+d, b+c)$ where $\sigma = \overline{12}$ or $\sigma = \overline{21}$. The first case implies that $a = \frac{-c-d}{2} = b$ and the second case implies $c = \frac{-a-b}{2} = d$, hence there are not more possibilities for the weight of $K_+(f_i(T))$.

The case $i = n$ is a simpler version of this one. □

Analogously, one can also prove that every opposite Demazure crystal atom has exactly one key tableau, using the following corollary of Lemma 5.2.1:

Corollary 5.2.4. *Let $\sigma = s_i$ be a generator of B_n and C an admissible column. Then $\mathrm{wt}(rC) = \mathrm{wt}(r(e_i(C)))$ or $\mathrm{wt}(rC) = \sigma(\mathrm{wt}(r(e_i(C))))$.*

Proof. Let $C' = e_i(C)$. By Lemma 5.2.1 we have that $\mathrm{wt}(rC') = \mathrm{wt}(r(f_i(C')))$ or $\mathrm{wt}(C') = \sigma(\mathrm{wt}(r(f_i(C'))))$. Hence $\mathrm{wt}(r(e_i(C))) = \mathrm{wt}(rC)$ or $\mathrm{wt}(e_i(C)) = \sigma(\mathrm{wt}(rC)) \Leftrightarrow \sigma(\mathrm{wt}(e_i(C))) = \mathrm{wt}(rC)$. □

The following lemma identifies when we can apply e_i to a column without taking it to 0.

Lemma 5.2.5. *Let $i \in [n]$ and C be an admissible column such that one of the following happens*

1. *$i < n$ and the weight of i in rC is less than the weight of $i+1$ in rC;*

2. *$i = n$ and weight of i is negative in rC,*

then we can apply e_i to C (in the sense $e_i(C) \neq 0$).

Proof. If $i = n$ then $-n$ appears on rC and n does not. Since n is the biggest unbarred letter of the alphabet we have that $-n$ also appears in C and n does not. Hence we can apply e_n to C.

If $i < n$ and the weight of i in rC is less than the weight of $i+1$ in rC then the weight of both can be one of the following three options: $(0,1)$, $(-1,1)$, $(-1,0)$. Note that rC does not have symmetric entries. So in the first two cases we have that $i+1$ exists in rC and i does not, hence $i+1$ exists in C and i does not, so we can apply e_i to C. In the last case, we have that $\bar{i}$ exists in rC and $i+1$ and $\overline{i+1}$ does not. Hence we have that $\bar{i}$ exists in C and i or $\overline{i+1}$ does not, so we can apply e_i to C. □

The next theorem gives a description of a Demazure crystal atom in type C using the right key map Theorem 5.1.5. Lascoux and Schützenberger, in [24, Theorem 3.8], proved the type A version of this theorem, which consists in considering the case when $v \in \mathbb{N}^n$ and, consequently, $\sigma_v \in \mathfrak{S}_n$. For inductive reasoning, used in what follows, we recall the chain property on the set of minimal length coset representatives modulo W_λ [6, Theorem 2.5.5].

Theorem 5.2.6 (Main theorem). *Let $v \in B_n\lambda$. Then $\mathfrak{U}(v) = \widehat{\mathfrak{B}}_v$.*

Proof. Let ρ be a minimal length coset representative modulo W_λ such that $v = \rho\lambda$. We will proceed by induction on $\ell(\rho)$. If $\ell(\rho) = 0$ then $\rho = id$ and $v = \lambda$. In this case we have that $\widehat{\mathfrak{B}}_\lambda = \{K(\lambda)\} = \mathfrak{U}(\lambda)$.

Let $\rho \geq 0$. Consider $\sigma = s_i$ a generator of B_n such that $\sigma\rho > \rho$ and $\sigma\rho\lambda \neq \rho\lambda = v$, i.e., $\rho^{-1}\sigma\rho \notin W_\lambda$. Recall e_i, ε_i, f_i and ϕ_i from the definition of the crystal $\mathfrak{B}^\lambda$. If $T \in \widehat{\mathfrak{B}}_{\sigma\rho\lambda}$ then T is obtained after applying f_i (maybe more than once) to a tableau in $\widehat{\mathfrak{B}}_{\rho\lambda}$, which by inductive hypothesis exists in $\mathfrak{U}(v)$. By Corollary 5.2.3, if $f_i(T) \notin \mathfrak{U}(v)$ then $f_i(T) \in \mathfrak{U}(\sigma v)$. So it is enough to prove that given a tableau $T \in \mathfrak{U}(v) \cup \mathfrak{U}(\sigma v)$ then $e_i^{\varepsilon_i(T)}(T) \in \mathfrak{U}(v)$.

We have two different cases to consider: $i = n$ and $i < n$.

If $T \in \mathfrak{U}(\sigma v)$ then, if $i < n$, there exists a frank word of T such that, if V_1 is its first column then rV_1 has less weight for i than for $i+1$ (less in the usual ordering of real numbers); if $i = n$, there exists a frank word of T such that, if V_1 is its first column then rV_1 has negative weight for i. Since we are in the column rV_1, if $i < n$, i and $i+1$ can have weights $(0,1)$, $(-1,1)$ or $(-1,0)$ and if $i = n$ then i has weight -1. Note that these are the exact conditions of Lemma 5.2.5. In either case, due to Lemma 5.2.5, we can applying e_i enough times to the frank word associated until this no longer happens. This is true because we only need to look to V_1 to see if it changes after applying e_i enough times to the frank word. In the signature rule we have that successive applications of e_i changes the letters of a word from the end to the beginning, so, from the remark after Lemma 5.2.1, the number of times that we need to apply e_i, in order to conditions of Lemma 5.2.5 do not hold for the first column, is $\varepsilon_i(T)$. So $K_+\left(e_i^{\varepsilon(T)}(T)\right) \neq K(\sigma v)$, hence, from Corollary 5.2.4, we have that $e_i^{\varepsilon_i(T)}(T) \in \mathfrak{U}(v)$.

If $T \in \mathfrak{U}(v)$ then $e_i^{\varepsilon_i(T)}(T) \in \mathfrak{U}(v)$ because if not, $e_i^{\varepsilon_i(T)}(T)$ will be in a Demazure crystal associated to $\rho' \in B_n$, with $\rho' < \rho$ such that $\sigma\rho' = \rho$. This cannot happen because in this case $\rho' = \sigma\rho < \rho$, which is a contradiction. □

5.2.1 Combinatorial description of type C_n Demazure characters and atoms

Recall the Demazure characters and Demazure atoms defined in 4.3. Theorem 5.2.6 detects the KN tableaux in $\mathfrak{B}^\lambda$ contributing to the Demazure atom $\widehat{\kappa}_v$, $\widehat{\kappa}_v = \sum_{\substack{K_+(T)=K(v)\\ T\in\mathfrak{B}^\lambda}} x^{\mathrm{wt}\,T}$.

Proposition 5.2.7. *Given $v \in B_n\lambda$, one has $\kappa_v = \sum_{u\le v} \widehat{\kappa}_u$.*

Proof. It is enough to prove that $\mathfrak{B}_v = \bigcup_{u\le v} \widehat{\mathfrak{B}}_u$, because κ_v and $\widehat{\kappa}_u$ are the generating functions of the tableau weights in $\mathfrak{B}_v$ and $\widehat{\mathfrak{B}}_u$, respectively. Since $v = \sigma\lambda$, where $\sigma := \sigma_v$, we can rewrite the identity as $\mathfrak{B}_{\sigma\lambda} = \bigcup_{\rho\le\sigma} \widehat{\mathfrak{B}}_{\rho\lambda}$.

We will proceed by induction on $\ell(\sigma)$. If $\ell(\sigma) = 0$ then the result follows because $\mathfrak{B}_\lambda = \widehat{\mathfrak{B}}_\lambda = \{K(\lambda)\}$. From the definition of Demazure crystal atom, we have $\widehat{\mathfrak{B}}_{\sigma\lambda} = \mathfrak{B}_{\sigma\lambda} \setminus \bigcup_{\rho<\sigma} \mathfrak{B}_{\rho\lambda}$, and by inductive hypothesis, we have that $\mathfrak{B}_{\rho\lambda} = \bigcup_{\rho'\le\rho} \widehat{\mathfrak{B}}_{\rho'\lambda}$. Hence:

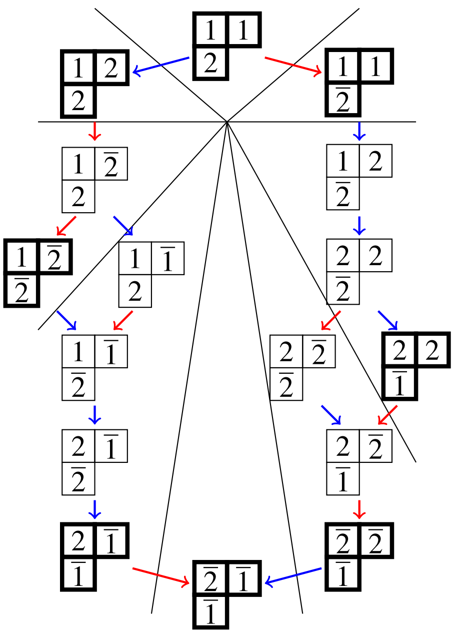
$$\widehat{\mathfrak{B}}_{\sigma\lambda} = \mathfrak{B}_{\sigma\lambda} \setminus \bigcup_{\rho<\sigma} \mathfrak{B}_{\rho\lambda} = \mathfrak{B}_{\sigma\lambda} \setminus \bigcup_{\rho<\sigma} \bigcup_{\rho'\le\rho} \widehat{\mathfrak{B}}_{\rho'\lambda} = \mathfrak{B}_{\sigma\lambda} \setminus \bigcup_{\rho'<\sigma} \widehat{\mathfrak{B}}_{\rho'\lambda}$$

□

Proposition 5.2.7, the equivalence $u \le v \Leftrightarrow K(u) \le K(v)$, and Theorem 5.2.6, allow us to detect the KN tableaux contributing to a key polynomial in type C:

$$\kappa_v = \sum_{u\le v} \widehat{\kappa}_u = \sum_{\substack{u\le v\\ T\in\mathfrak{U}(u)}} x^{\mathrm{wt}T} = \sum_{\substack{K(u)\le K(v)\\ T\in\mathfrak{U}(u)}} x^{\mathrm{wt}T} = \sum_{K_+(T)\le K(v)} x^{\mathrm{wt}T}.$$

Example 5.2.8. *Recall the type C_2 crystal $\mathfrak{B}^{(2,1)}$, partitioned into Demazure crystal atoms.*



One can check that, for example $\mathfrak{U}((1,\bar{2})) = \left\{ \begin{array}{|c|c|}\hline 1 & \bar{2} \\ \hline 2 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}, \begin{array}{|c|c|}\hline 1 & \bar{2} \\ \hline \bar{2} & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array} \right\} = \widehat{\mathfrak{B}}_{\lambda s_1 s_2}$.
Also,
$\mathfrak{B}_{(1,\bar{2})} = \{T \in \mathfrak{B}^\lambda \mid K_+(T) \le K((1,\bar{2}))\} = \left\{ \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 2 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}, \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 2 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}, \begin{array}{|c|c|}\hline 1 & 1 \\ \hline \bar{2} & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}, \begin{array}{|c|c|}\hline 1 & \bar{2} \\ \hline 2 & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array}, \begin{array}{|c|c|}\hline 1 & \bar{2} \\ \hline \bar{2} & \multicolumn{1}{c}{} \\ \cline{1-1} \end{array} \right\}$.

5.3 Symplectic right and left keys - direct way

In this section, we start by by introducing two maps, K_+^1 and K_-^1, that, given a tableau, return the leftmost column of the right key and the rightmost column of the left key, respectively. Given a tableau T, we can express its right key $K_+(T)$ in terms of K_+^1 applied to T and to some subtableaux of T, and analogously for left side. For each side, we start by introducing an algorithm, based on SJDT, to compute these maps K_+^1 and K_-^1. Motived by Willis' direct way of computing right and left keys of SSYT's [42], we introduce a way of computing these maps K_+^1 and K_-^1, and consequently symplectic right and left keys, without the use of SJDT. We end this section with an example of these direct algorithms.

5.3.1 The right key of a tableau - *Jeu de taquin* approach

Let $T = C_1C_2\cdots C_k$ be a straight KN tableau with columns $C_1, C_2, \ldots, C_k$. Note that, to compute which entries appear in the i-th column of $K_+(T)$ we do not need to look to the first $i-1$ columns of T. We only need the last column of a skew tableau obtained by applying the SJDT to the columns $C_i \cdots C_k$ of T, so that the last column has the length of C_i, because, by Proposition 4.4.5, all last columns of skew tableaux associated to T with the same length are equal. Let $K_+^1(T)$ be the map that given a tableau returns the first column of $K_+(T)$. This is noticeable in Example 5.1.7 where $K_+(T) = K_+^1(C1C2C3)K_+^1(C2C3)K_+^1(C3)$. In general, $K_+(T) = K_+^1(C_1\cdots C_k)K_+^1(C_2\cdots C_k)\cdots K_+^1(C_k)$. Based on this observation and Proposition 4.4.5, next algorithm summarizes our way to compute $K_1^+(T)$ using SJDT:

Algorithm 5.3.1. *Let T be a straight KN tableau:*

1. *Let $i = 2$.*
2. *If T has exactly one column, return the right column of T. Otherwise, let $T_i = T_2$ be the tableau formed by the first two columns of T.*
3. *If the length of the two columns of T_i is the same, put $T_i' := T_i$. Else, play the SJDT on T_i until both column lengths are swapped, obtaining T_i'.*
4. *If T has more than i columns, redefine $i := i+1$, and define T_i to be the two-columned tableau formed by the rightmost column of T_{i-1}' and the i-th column of T, and go back to 3.. Else, return the right column of the rightmost column of T_i'.*

This algorithm is illustrated on the bottom path of Example 5.1.7.

Corollary 5.3.2. *If T is a rectangular tableau, $K_+(T) = rC_k rC_k \cdots rC_k$ (k times).*

Next, we present a way of computing $K_+^1(T)$ that does not require the SJDT. Willis has done this when T is a SSYT [42]. It is a simplified version of the algorithm presented here.

5.3.2 Right key - a direct way

Let $T = C_1C_2$ be a straight KN two column tableau and $spl(T) = \ell C_1 rC_1 \ell C_2 rC_2$ a straight semistandard tableau. In particular, $rC_1\ell C_2$ is a semistandard tableau. The *matching between* rC_1 *and* ℓC_2 is defined as follows:

• Let $\beta_1 < \cdots < \beta_{m'}$ be the elements of ℓC_2. Let i go from m' to 1, match β_i with the biggest, not yet matched, element of rC_1 smaller or equal than β_i.

Theorem 5.3.3 (The direct way algorithm for the right key)**.** *Let T be a straight KN tableau with columns $C_1, C_2, \ldots, C_k$, and consider its split form $spl(T)$. For every right column $rC_2, \ldots, rC_k$, add empty cells to the bottom in order to have all columns with the same length as rC_1. We will fill all of these empty cells recursively, proceeding from left to right. The extra numbers that are written in the column rC_2 are found in the following way:*

• *match rC_1 and ℓC_2.*

• *Let $\alpha_1 < \cdots < \alpha_m$ be the elements of rC_1. Let i go from 1 to m. If α_i is not matched with any entry of ℓC_2, write in the new empty cells of rC_2 the smallest element bigger or equal than α_i such that neither it or its symmetric exist in rC_2 or in its new cells. Let C_2' be the column defined by rC_2 together with the filled extra cells, after ordering.*

To compute the filling of the extra cells of rC_3, we do the same thing, with C_2' and C_3. If we do this for all pairs of consecutive columns, we eventually obtain a column C_k', consisting of rC_k together with extra cells, with the same length as rC_1. We claim that $C_k' = K_+^1(T)$.

Example 5.3.4. *Let* $T = C_1C_2C_3 = \begin{array}{|c|c|c|}\hline 1 & 3 & \overline{1} \\ \hline 3 & \overline{3} & \\ \cline{1-2} \overline{3} & & \\ \cline{1-1}\end{array}$ *, with split form* $spl(T) = \begin{array}{|c|c|c|c|c|c|}\hline 1 & 1 & 2 & 3 & \overline{1} & \overline{1} \\ \hline 2 & 3 & \overline{3} & \overline{2} & & \\ \cline{1-4} \overline{3} & \overline{2} & & & & \\ \cline{1-2}\end{array}$ *. We match rC_1 and ℓC_2, as indicated by the letters a and b:* $\begin{array}{|c|c|c|c|c|c|}\hline 1 & 1^a & 2^a & 3 & \overline{1} & \overline{1} \\ \hline 2 & 3^b & \overline{3}^b & \overline{2} & & \\ \cline{1-4} \overline{3} & \overline{2} & & & & \\ \cline{1-2}\end{array}$ *. Hence $\overline{2}$ creates a $\overline{1}$ in rC_2, completing the right column rC_2:* $\begin{array}{|c|c|c|c|c|c|}\hline 1 & 1 & 2 & 3 & \overline{1}^a & \overline{1} \\ \hline 2 & 3 & \overline{3} & \overline{2} & & \\ \cline{1-4} \overline{3} & \overline{2} & & \overline{1}^a & & \\ \cline{1-4}\end{array}$ *. Now we match C_2' and ℓC_3, which is already done, and see what new cells 3 and $\overline{2}$ create in rC_3, obtaining* $\begin{array}{|c|c|c|c|c|c|}\hline 1 & 1 & 2 & 3 & \overline{1} & \overline{1} \\ \hline 2 & 3 & \overline{3} & \overline{2} & & 3 \\ \hline \overline{3} & \overline{2} & & \overline{1} & & \overline{2} \\ \hline\end{array}$*. Hence* $K_+^1(T) = \begin{array}{|c|}\hline 3 \\ \hline \overline{2} \\ \hline \overline{1} \\ \hline\end{array}$ *is obtained from C_3' after reordering its entries.*

The proof of Theorem 5.3.3

It is enough to prove that by the end of this algorithm, the entries in C_k' are the entries on the right column of the rightmost column of T_k' from Algorithm 5.3.1. In fact, it is enough to do this for $k = 2$. For bigger k note that the entries that are "slid" into C_k come from rC_{k-1}, so, to go to the next step on the SJDT algorithm we only need to know the previous right column, which is exactly what we claim to compute this way. The next lemma determines which number is added to rC_2 given that we know α, the entry that is horizontally slid:

Lemma 5.3.5. *Suppose that $T = C_1C_2$ is a non-rectangular two-column tableau (if the tableau is rectangular then we have nothing to do). Play the SJDT on this tableau which ends up moving one cell from the first column to the second (some entries may change their values). Then,*

- *Immediately before the horizontal slide of the SJDT, the entry α, on the left of the puncture, is an unmatched cell of rC_1.*
- *Call C_1' and C_2' to both columns after the horizontal slide on T. The new entry in rC_2', compared to rC_2, is the smallest element bigger or equal than α such that neither it or its symmetric exist in rC_2.*

Example 5.3.6. *Let* $T = \begin{array}{|c|c|}\hline 2 & 3 \\ \hline 3 & 4 \\ \hline 5 & \overline{5} \\ \hline \overline{5} \\ \cline{1-1} \overline{2} \\ \cline{1-1}\end{array}$. *After splitting, and just before the first horizontal slide, we have*

$T = \begin{array}{|c|c|c|c|}\hline 1 & 2 & 3 & 3 \\ \hline 3 & 3 & 4 & 4 \\ \hline 4 & 5 & \overline{5} & \overline{5} \\ \hline \overline{5} & \overline{4} & * & * \\ \hline \overline{2} & \overline{1} \\ \cline{1-2}\end{array}$. *The new entry in rC_2 is $\overline{2}$, as predicted by the lemma:* $\begin{array}{|c|c|c|c|}\hline 1 & 2 & 2 & 3 \\ \hline 3 & 3 & 3 & 4 \\ \hline 4 & 4 & \overline{5} & \overline{5} \\ \hline * & * & \overline{4} & \overline{2} \\ \hline \overline{2} & \overline{1} \\ \cline{1-2}\end{array}$.

Proof. **Case 1:** α is barred. Then $C_2' = C_2 \cup \{\alpha\}$. If $\overline{\alpha}$ does not exist neither in C_2 nor in $\Phi(C_2)$, then α will exist in both C_2' and $\Phi(C_2')$. If $\overline{\alpha}$ does exist in C_2, and consequently in $\Phi(C_2)$ (but $\alpha \notin \Phi(C_2)$), then α and $\overline{\alpha}$ will both exist in C_2'. Hence, in the construction of the barred part of $\Phi(C_2')$, compared to $\Phi(C_2)$, there will be a new barred number which is the smallest number bigger (or equal, but the equality can not happen) than α such that neither it nor its symmetric exist in the barred part of $\Phi(C_2)$ or the unbarred part of C_2 (i.e., rC_2). If α existed in $\Phi(C_2)$, then $\overline{\alpha}$ existed in $\Phi(C_2)$. That means that whatever number got sent to α in the construction of $\Phi(C_2)$ will be sent to the next available number, meaning that in rC_2 will appear a new number, the smallest number bigger (or equal, but the equality can not happen because α is already there) than α such that neither it nor its symmetric exist in rC_2.

Case 2: α is unbarred. Then $C_2' = \Phi^{-1}(\Phi(C_2) \cup \{\alpha\})$. If $\overline{\alpha}$ does not exist in C_2 nor in $\Phi(C_2)$, then α will exist in both C_2' and $\Phi(C_2')$. If $\overline{\alpha}$ existed in $\Phi(C_2)$, and consequently in C_2, then both α and $\overline{\alpha}$ will exist in $\Phi(C_2')$, hence, if we start in the coadmissible column, in the construction of the unbarred part of C_2', compared to C_2, there will be a new unbarred number which is the smallest number bigger than α such that neither it nor its symmetric exist in rC_2. Finally, if α existed in C_2, then $\overline{\alpha}$ also existed in C_2. That means that whatever number got sent to α in the construction of C_2, from $\Phi(C_2)$, will be sent to the next available number, meaning that in rC_2 will appear a new number, the smallest number bigger than α such that neither it nor its symmetric exist in rC_2. □

Proof of Theorem 5.3.3: Each SJDT in T, a two-column skew tableau, moves a cell from the first to the second column. We will prove that if we apply the direct way algorithm after each SJDT, the output C_2' does not change. The cells on ℓC_2 without cells to its left do not get to be matched. When we slide horizontally, the columns rC_1 and ℓC_2 may change more than the adding/removal of α, the horizontally slid entry. Since the horizontal slides happen from top to bottom, we only need to see what changes happen to bigger entries than the one slid. All entries above α are matched to the entry in the same row in ℓC_2.

If α is barred then, the remaining barred entries of rC_1 and ℓC_2 remain unchanged, and since all entries above α, including the unbarred ones, are matched to the entry directly on their right, there is no noteworthy change and everything runs as expected.

If α is unbarred then, the remaining unbarred entries of rC_1 and ℓC_2 remain unchanged. In the barred part of rC_1 either nothing happens, or there is an entry bigger than $\overline{\alpha}$, $\overline{x}$, that gets replaced by $\overline{\alpha}$. Note that $\overline{x}$ must be such that for every number between $\overline{x}$ and $\overline{\alpha}$, either it or its symmetric existed in rC_1. In the barred part of ℓC_2, if $\overline{\alpha} \in \ell C_2$, then $\overline{\alpha}$ gets replaced by $\overline{y}$, smaller than $\overline{\alpha}$, such that for every number between $\overline{y}$ and $\overline{\alpha}$, either it or its symmetric existed in ℓC_2, and both y and $\overline{y}$ do not exist in ℓC_2.

Let's look to ℓC_2. Let $\alpha < p_1 < p_2 < \cdots < p_m = y$ be the numbers between α and y that does not exist in ℓC_2, right before the horizontal slide. Then, their symmetric exist in ℓC_2. For all numbers in rC_2 between α and y, there exists, in the same row in rC_1, a number between α and y. Let $\alpha < p'_1 < p'_2 < \cdots < p'_m = y$ be the missing numbers between α and y in rC_1, then $p_i \leq p'_i$. Note that $\overline{p_1} > \overline{p_2} > \cdots > \overline{p_m} = \overline{y}$ exist in ℓC_2 after the horizontal slide and that the biggest numbers between $\overline{\alpha}$ and $\overline{y}$ (not including $\overline{\alpha}$) that can exist in rC_1 are $\overline{p'_1} > \overline{p'_2} > \cdots > \overline{p'_m}$, and since $\overline{p_i} \geq \overline{p'_i}$, the matching holds for this interval after swapping $\overline{\alpha}$ by $\overline{y}$ in ℓC_2.

Now let's look to rC_1. Before the slide, call $\overline{x'}$ to the biggest unmatched number of rC_1 smaller or equal than $\overline{x}$ and bigger than $\overline{\alpha}$. If there is no such $\overline{x'}$, then everything in rC_1 between $\overline{\alpha}$ and $\overline{x}$ is matched, hence swapping $\overline{x}$ by $\overline{\alpha}$ will keep all of them matched, meaning that the algorithm works in this scenario. Let $x' < q_1 < q_2 < \cdots < q_m < \alpha$ be the numbers between x' and α that does not exist in rC_1, right before the horizontal slide. Then, their symmetric exist in rC_1. For all numbers in rC_1 between x' and α, there exists, in the same row in ℓC_2, a number between x' and α, because α is unmatched. Let $x' < q'_1 < q'_2 < \cdots < q'_m < \alpha$ be the missing numbers between x' and α in ℓC_2, then $q_i \geq q'_i$. Note that $\overline{q_1} > \overline{q_2} > \cdots > \overline{q_m} > \overline{\alpha}$ exist in rC_1 after the horizontal slide and the numbers between $\overline{x'}$ and $\overline{\alpha}$ that can exist in ℓC_2 are $\overline{q'_1} > \overline{q'_2} > \cdots > \overline{q'_m}$, and since $\overline{q_i} \leq \overline{q'_i}$, these numbers are matching a number bigger or equal then $\overline{q_i}$ in rC_1, meaning that $\overline{\alpha}$ is unmatched in rC_1 after the slide. Ignoring signs, the numbers that appear in either rC_2 or ℓC_2 are the same. So before playing the SJDT, applying the direct way algorithm we have that the unmatched numbers in rC_1 are sent to the not used numbers of $\overline{q'_1} > \overline{q'_2} > \cdots > \overline{q'_m}$ in ℓC_2 (this is a bijection), and $\overline{x'}$ is sent to the smallest available number, bigger or equal than $\overline{x'}$. Now consider rC_1 and ℓC_2 after the slide. In rC_1 we replace $\overline{x'}$ by $\overline{\alpha}$ and remove α and in ℓC_2 there is α and not $\overline{\alpha}$. In the direct algorithm, all unmatched numbers of $\overline{q_1} > \overline{q_2} > \cdots > \overline{q_m} > \overline{\alpha}$ are sent to the not used numbers of $\overline{q'_1} > \overline{q'_2} > \cdots > \overline{q'_m}$ in ℓC_2, but now we have more numbers in the first set than in the second, meaning that $\overline{\alpha}$ will bump the image of the smallest unmatched number, which will bump the image of the second smallest unmatched number, and so on, meaning that the image of biggest unmatched will be out of this set. This image will be the smallest number available, which was the image of $\overline{x'}$ before the horizontal slide.

Hence, the outcome of the direct way does not change due to the changes to the columns when we play the SJDT, meaning that the outcome is what we intend. □

5.3.3 The left key of a tableau - *Jeu de taquin* approach

Now we present a way to compute the leftmost column of the left key of a tableau, via SJDT, and in Section 5.3.4 we present a way of doing that does not require the use of SJDT.

Analogously to the symplectic right key map, let $T = C_1 C_2 \cdots C_k$ be a KN tableau and note that, to compute which entries appear in the i-th column of $K_-(T)$, we only need to look to the first i columns of T. We need the first column of a skew tableau obtained by applying the SJDT to the columns $C_1 \cdots C_i$ of T, so that the first column has the length of C_i. Let $K_-^1(T)$ be the map that given a tableau returns the last column of $K_-(T)$. Then, $K_-(T) = K_-^1(C_1) \cdots K_-^1(C_1 \cdots C_{k-1}) K_-^1(C_1 \cdots C_k)$. In Example 5.1.9 we have $K_-(T) = K_-^1(C1) K_-^1(C1C2) K_-^1(C1C2C3)$.

Next we present how we compute $K_-^1(T)$ using SJDT:

Algorithm 5.3.7. *Let k be the number of columns of T and $i = k-1$.*

1. *If T has exactly one column, return the left column of T. Otherwise, let $T_i := T_{k-1}$ be the tableau formed by the last two columns of T.*

2. *If the length of the two columns of T_i is the same, put $T_i' := T_i$. Else, play the SJDT on T_i until both column lengths are swapped, obtaining T_i'.*

3. *If $i \neq 1$, redefine $i := i-1$, and define T_i as the two-columned tableau formed with the leftmost column of T_{i+1}' and the i-th column of T, and go back to (1). Else, return the left column of the leftmost column of T_i'.*

This algorithm is exemplified on the top path of Example 5.1.7.

Corollary 5.3.8. *If T is a rectangular tableau, $K_-(T) = \ell C_1 \ell C_1 \cdots \ell C_1$ (k times).*

Next, we present a way of computing $K_-^1(T)$ that does not require the use of SJDT. In [42], this is done when T is a SSYT. It is simplified version of the algorithm presented here.

5.3.4 Left key - a direct way

Theorem 5.3.9. *Let T be a KN tableau with columns $C_1, C_2, \ldots, C_k$, and consider its split form $spl(T)$.*

We will now delete entries from the left columns, proceeding from right to left, in such a way that in the end every left column has as many entries as C_k. The entries deleted from ℓC_{k-1} are found in the following way:

We start by creating a matching between rC_{k-1} and ℓC_k. Let $\beta_1 < \cdots < \beta_m$ be the unmatched elements of rC_{k-1}. For i between 1 and m, let α_i be the entry on ℓC_{k-1} next to β_i. Let i go from 1 to m. Starting at α_i and going up, delete the first entry of ℓC_{k-1} bigger than the entry directly Northeast of it. If there is no entry in this conditions, delete the top entry of ℓC_{k-1}. Also delete b_i from rC_{k-1}. By the end of this procedure we obtain $\ell C_{k-1}'$ with the same number of cells as C_k.

To continue the algorithm, we do the same thing with C_{k-2} and $\ell C_{k-1}'$. If we do this for all pairs of consecutive columns, we eventually obtain a column $\ell C_1'$, consisting of ℓC_1 with some entries deleted, with the same length as C_k. We claim that $\ell C_1' = K_-^1(T)$.

Example 5.3.10. *Consider* $T = \begin{array}{ccc} 2 & 3 & \overline{3} \\ 3 & \overline{3} & \\ \overline{3} & & \end{array}$ *, whose split form is* $spl(T) = \begin{array}{cccccc} 1 & 2 & 2 & 3 & \overline{3} & \overline{3} \\ 2 & 3 & \overline{3} & \overline{2} & & \\ \overline{3} & \overline{1} & & & & \end{array}$ *. We match* rC_2 *and* ℓC_1*, obtaining:* $\begin{array}{cccccc} 1 & 2 & 2 & 3^a & \overline{3}^a & \overline{3} \\ 2 & 3 & \overline{3} & \overline{2} & & \\ \overline{3} & \overline{1} & & & & \end{array}$ *. Hence* $\overline{2}$ *is unmatched in* rC_2*. So it will get deleted, alongside the* $\overline{3}$ *in* ℓC_2*. Thus we have* $\begin{array}{cccccc} 1 & 2^a & 2^a & 3 & \overline{3} & \overline{3} \\ 2 & 3 & {\color{gray}\overline{3}} & {\color{gray}\overline{2}} & & \\ \overline{3} & \overline{1} & & & & \end{array}$ *(the deleted entries are greyed out).*

Now we have to create the match between $\ell C_2'$ *and* rC_1*, which is already done. The entries* 3 *and* $\overline{1}$ *are unmatched in* rC_1*, hence they will be removed alongside the entries* 1 *and* $\overline{3}$ *in* ℓC_1*, obtaining* $\begin{array}{cccccc} {\color{gray}1} & 2 & 2 & 3 & \overline{3} & \overline{3} \\ 2 & {\color{gray}3} & {\color{gray}\overline{3}} & {\color{gray}\overline{2}} & & \\ {\color{gray}\overline{3}} & {\color{gray}\overline{1}} & & & & \end{array}$ *. Hence* $K_-^1(T) = \boxed{2}$ *.*

Proof of Theorem 5.3.9

It is enough to prove that by the end of this algorithm, the entries in $\ell C_j'$ are the entries on the left column of the leftmost column of T_j' from Algorithm 5.3.7. Just like in the right key case, it is enough to do this for $j = k-1$. For smaller j note that we only need to know what remains in the left column $\ell C_j'$, which is exactly what we claim to compute this way.

So only need to prove this when T is a two-column tableaux.

Lemma 5.3.11. *Suppose that* T *is a non-rectangular two-column tableau (if the tableau is rectangular then we have nothing to do). Play the SJDT on this tableau, which ends up moving one cell from the first column to the second (some entries may change its value). Immediately before the horizontal slide of the SJDT, the entry* β*, on the left of the puncture, is an unmatched cell of* rC_1*. Call* C_1' *and* C_2' *to both columns after the slide.*

Then $\ell C_1'$ *will lose an entry, compared to* ℓC_1*, which is the biggest entry of* ℓC_1*, in a row not under the row that contains* β*, bigger than the entry directly Northeast of it.*

Example 5.3.12. *Consider the tableau* $T = \begin{array}{cc} 2 & 3 \\ 4 & 4 \\ 5 & \overline{2} \\ \overline{5} & \\ \overline{2} & \end{array}$*. After split, and just before the horizontal slide, we have* $T = \begin{array}{cccc} 1 & 2 & 3 & 3 \\ 3 & 4 & 4 & 4 \\ 4 & 5 & * & * \\ \overline{5} & \overline{3} & \overline{2} & \overline{2} \\ \overline{2} & \overline{1} & & \end{array}$*. So* 5 *slides from* rC_1 *to* ℓC_2*, obtaining the tableau* $\begin{array}{cc} 2 & 3 \\ 4 & 4 \\ * & 5 \\ \overline{5} & \overline{2} \\ \overline{2} & \end{array}$*, whose split is* $\begin{array}{cccc} 1 & 2 & 3 & 3 \\ 4 & 4 & 4 & 4 \\ * & * & 5 & 5 \\ \overline{5} & \overline{5} & \overline{2} & \overline{2} \\ \overline{2} & \overline{1} & & \end{array}$*. The entry removed from* ℓC_1 *is* 3*, as predicted by the lemma.*

Proof. If β is unbarred then look at all numbers $\beta \leq i \leq n$, and count, in C_1, count how many of them exist together with its symmetric and it is not matched to a number with bigger than β in the coadmissible column. Let k be that count. Now let i go from $\beta - 1$ to 1. If i and $\bar{i}$ exist in C_1 then $k := k+1$, and if neither exist then $k := k-1$. Since C_1 is admissible, eventually $k = 0$ and this is the i removed from ℓC_1. So, the columns ℓC_1 and rC_1 have same number of entries with absolute value bigger or equal than i, hence the entry i of ℓC_1 is bigger than the entry directly Northeast of it.

If β is barred then look at all numbers $\beta \leq i \leq \bar{1}$, and count, in C_1, count how many of them exist together with its symmetric and it is not matched to a number bigger than β in the coadmissible column. Let k be that count. Now let i go from $\beta - 1$ to $\bar{n}$. If i and $\bar{i}$ exist in C_1 then $k := k+1$, and if neither exist then $k := k-1$. Since $\Phi(C_1)$ is coadmissible, eventually $k = 0$ and this is the i removed from ℓC_1. The columns ℓC_1 and rC_1 have same number of entries with absolute value smaller or equal than $\bar{i}$, hence the entry i of ℓC_1 is bigger than the entry directly Northeast of it (remember that i is negative). □

Proof of Theorem 5.3.9: Hence we have determined which entry is removed from ℓC_1 given that we know β, the entry of the cell that is horizontally slid. The SJDT on T may change the entries or the matching in rC_1. We need to prove that, even with these eventual changes, the entries removed from ℓC_1 are the ones that we calculated in the beginning, before doing any SJDT slide.

If β is barred, since we run the unmatched entries of rC_1 from smallest to biggest, when removing β from rC_1 the unbarred part of rC_1 remains the same, hence, the remaining entries and matched entries do not change, hence the outcome will be the one predicted.

If β is unbarred then the remaining unbarred entries of rC_1 remain unchanged. In the barred part of rC_1 either nothing happens, or there is an entry bigger than $\overline{\beta}$, $\bar{x}$, that gets replaced by $\overline{\beta}$. Note that $\bar{x}$ must be such that for every number between $\bar{x}$ and $\overline{\beta}$, either it or its symmetric existed in rC_1. This can only happen if k, from the proof of Lemma 5.3.11 starts being bigger than 0.

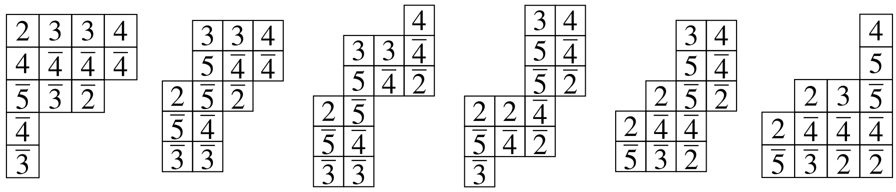
Since for all numbers between $\bar{x}$ and $\overline{\beta}$ either it or its symmetric exist in rC_1, all unmatched entries here will remove from ℓC_1 an entry smaller or equal than $\bar{x}$. In fact, the way of constructing $\bar{x}$ and i, from the proof of Lemma 5.3.11, is effectively the same. Since, after the slide of β, we may have different matches in the numbers between $\bar{x}$ and $\overline{\beta}$, and the number of unmatched entries remains the same after the slide. Since all unmatched entries in here will remove something smaller or equal than $\overline{\beta}$ from ℓC_1, the outcome of the algorithm is the same as if we apply it to ℓC_1, rC_1 before or after the horizontal slide. Hence we do not need to do any SJDT in order to know the entries of ℓC_1 after the SJDT. □

5.3.5 Example

In this section we present a KN tableau and compute its right and left keys via SJDT and using the direct way.

Let T be the KN tableau $\begin{array}{|c|c|c|c|}\hline 2 & 3 & 3 & 4 \\ \hline 4 & \bar{4} & \bar{4} & \bar{4} \\ \hline \bar{5} & \bar{3} & \bar{2} \\ \cline{1-3} \bar{4} \\ \cline{1-1} \bar{3} \\ \cline{1-1} \end{array}$ with split form $\begin{array}{|c|c|c|c|c|c|c|c|}\hline 1 & 2 & 2 & 3 & 3 & 3 & 3 & 4 \\ \hline 3 & 4 & \bar{4} & \bar{4} & \bar{4} & \bar{4} & \bar{4} & \bar{3} \\ \hline \bar{5} & \bar{5} & \bar{3} & \bar{2} & \bar{2} & \bar{2} \\ \cline{1-6} \bar{4} & \bar{3} \\ \cline{1-2} \bar{3} & \bar{1} \\ \cline{1-2} \end{array}$.

In order to find the right (resp. left) key of T, we play the SJDT to swap heights of consecutive columns, and find skew tableaux, Knuth related to T, such that for every column height there is a skew tableau whose last column (resp. first) has that height.



Each tableau is obtained from the previous after playing SJDT in two consecutive columns, swapping their heights.

If we compute the right (resp. left) columns of all last (resp. first) columns of these tableaux, we find the columns of the right (resp. left) key associated to T:

$$K_+(T) = \begin{array}{|c|c|c|c|}\hline 4 & 4 & 4 & 4 \\ \hline 5 & \bar{3} & \bar{3} & \bar{3} \\ \hline \bar{3} & \bar{2} & \bar{2} & \\ \cline{1-3} \bar{2} & & & \\ \cline{1-1} \bar{1} & & & \\ \cline{1-1}\end{array} \quad \text{and } K_-(T) = \begin{array}{|c|c|c|c|}\hline 1 & 2 & 2 & 2 \\ \hline 2 & \bar{5} & \bar{5} & \bar{5} \\ \hline \bar{5} & \bar{3} & \bar{3} & \\ \cline{1-3} \bar{4} & & & \\ \cline{1-1} \bar{3} & & & \\ \cline{1-1}\end{array} .$$

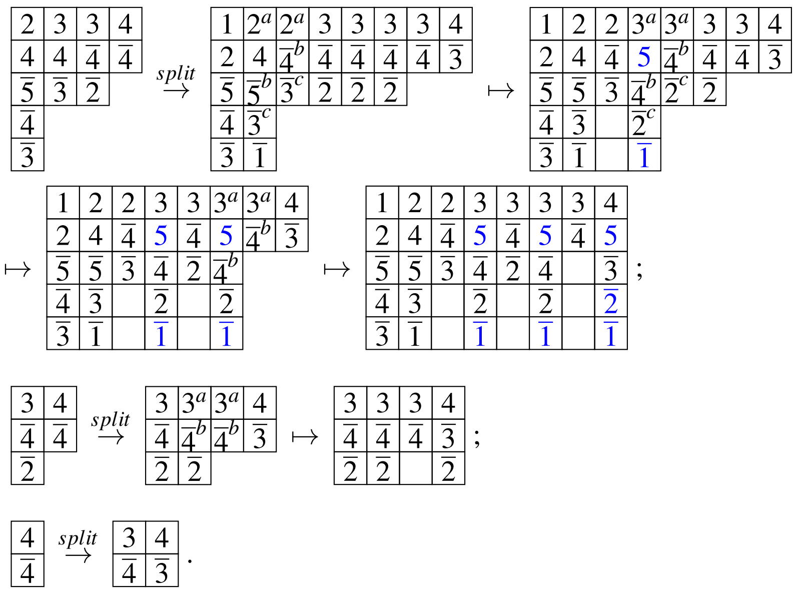
Note that we have 9 horizontal slides in our sequence of tableaux, and for each horizontal slide we have to apply the map Φ, or its inverse, two times. This means that we are effectively computing the split form of 9 skew tableaux, even though we only need 3 tableaux (the first, the third and the last one) to have all column heights in each end of the tableau.

Now we compute both keys using the direct way. In here we only need to compute one split form, and make some calculations on it, and on subtableaux of the split form.

To compute the right key, via direct way, we need to compute the columns $K_+^1\left(\begin{array}{|c|c|c|c|}\hline 2 & 3 & 3 & 4 \\ \hline 4 & \bar{4} & \bar{4} & \bar{4} \\ \hline \bar{5} & \bar{3} & \bar{2} & \\ \cline{1-3} \bar{4} & & & \\ \cline{1-1} \bar{3} & & & \\ \cline{1-1}\end{array}\right)$,

$$K_+^1\left(\begin{array}{|c|c|c|}\hline 3 & 3 & 4 \\ \hline \bar{4} & \bar{4} & \bar{4} \\ \hline \bar{3} & \bar{2} & \\ \cline{1-2}\end{array}\right) = K_+^1\left(\begin{array}{|c|c|}\hline 3 & 4 \\ \hline \bar{4} & \bar{4} \\ \hline \bar{2} & \\ \cline{1-1}\end{array}\right) \text{ and } K_+^1\left(\begin{array}{|c|}\hline 4 \\ \hline \bar{4} \\ \hline\end{array}\right).$$

We start by splitting and matching, and every $\mapsto$ marks when new entries, written in blue, are added to a right column, and we do these until there are no columns left.



To compute the left key, via direct way, we need to compute the columns $K_-^1\left(\begin{array}{|c|c|c|c|}\hline 2&3&3&4\\\hline 4&\bar4&\bar4&\bar4\\\hline \bar5&\bar3&\bar2&\\\hline \bar4&&&\\\hline \bar3&&&\\\hline\end{array}\right)$, $K_-^1\left(\begin{array}{|c|c|}\hline 2&3\\\hline 4&\bar4\\\hline \bar5&\bar3\\\hline \bar4&\\\hline \bar3&\\\hline\end{array}\right) = K_-^1\left(\begin{array}{|c|c|c|}\hline 2&3&3\\\hline 4&\bar4&\bar4\\\hline \bar5&\bar3&\bar2\\\hline \bar4&&\\\hline \bar3&&\\\hline\end{array}\right)$ and $K_-^1\left(\begin{array}{|c|}\hline 2\\\hline 4\\\hline \bar5\\\hline \bar4\\\hline \bar3\\\hline\end{array}\right)$. We start by splitting and matching, and every $\mapsto$ marks when entries are removed from a left column, and we do these until there are no columns left. Recall that this algorithm goes from right to left.

$$\begin{array}{|c|c|c|c|}\hline 2&3&3&4\\\hline 4&\bar4&\bar4&\bar4\\\hline \bar5&\bar3&\bar2&\\\hline \bar4&&&\\\hline \bar3&&&\\\hline\end{array} \overset{split}{\rightarrow} \begin{array}{|c|c|c|c|c|c|c|c|}\hline 1&2&2&3&3&3^a&3^a&4\\\hline 2&4&\bar4&\bar4&\bar4&\bar4^b&\bar4^b&\bar3\\\hline \bar5&\bar5&\bar3&\bar2&\bar2&\bar2&&\\\hline \bar4&\bar3&&&&&&\\\hline \bar3&\bar1&&&&&&\\\hline\end{array} \mapsto \begin{array}{|c|c|c|c|c|c|c|c|}\hline 1&2&2&3^a&3^a&3&3&4\\\hline 2&4&\bar4&\bar4^b&\bar4^b&\bar4&\bar4&\bar3\\\hline \bar5&\bar5&\bar3&\bar2&&&&\\\hline \bar4&\bar3&&&&&&\\\hline \bar3&\bar1&&&&&&\\\hline\end{array}$$

$$\mapsto \begin{array}{|c|c|c|c|c|c|c|c|}\hline 1&2^a&2^a&3&3&3&3&4\\\hline 2&4&\bar4^b&\bar4&\bar4&\bar4&\bar4&\bar3\\\hline \bar5&\bar5^b&&&&&&\\\hline \bar4&\bar3&&&&&&\\\hline \bar3&\bar1&&&&&&\\\hline\end{array} \mapsto \begin{array}{|c|c|c|c|c|c|c|c|}\hline &2&2&3&3&3&3&4\\\hline 2&&\bar4&\bar4&\bar4&\bar4&\bar4&\bar3\\\hline \bar5&\bar5&&\bar2&&\bar2&&\\\hline &&&&&&&\\\hline &&&&&&&\\\hline\end{array}$$

In the final step, we are removing $\overline{3}$ from ℓC_1, because the entry directly Northeast of it is $\overline{5}$, because the $\overline{3}$ of rC_1 has already been slid out.

$$\begin{array}{|c|c|}\hline 2&3\\\hline 4&\bar4\\\hline \bar5&\bar3\\\hline \bar4&\\\hline \bar3&\\\hline\end{array} \overset{split}{\rightarrow} \begin{array}{|c|c|c|c|}\hline 1&2^a&2^a&3\\\hline 2&4&\bar4^b&\bar4\\\hline \bar5&\bar5^b&\bar3^c&\bar2\\\hline \bar4&\bar3^c&&\\\hline \bar3&\bar1&&\\\hline\end{array} \mapsto \begin{array}{|c|c|c|c|}\hline &2&2&3\\\hline 2&&\bar4&\bar4\\\hline \bar5&\bar5&\bar3&\bar2\\\hline &\bar3&&\\\hline \bar3&&&\\\hline\end{array}$$

$$\begin{array}{|c|}\hline 2\\\hline 4\\\hline \bar5\\\hline \bar4\\\hline \bar3\\\hline\end{array} \overset{split}{\rightarrow} \begin{array}{|c|c|}\hline 1&2\\\hline 2&4\\\hline \bar5&\bar5\\\hline \bar4&\bar3\\\hline \bar3&\bar1\\\hline\end{array}$$

Chapter 6

Realization of the Lusztig involution in types A_{n-1} and C_n

In this chapter we present an involution on a crystal, known as Lusztig involution [30]. For the type A_{n-1} crystal of SSYT's this involution is usually known as Schützenberger involution or evacuation, and can be realized via *jeu da taquin* or, equivalently, via column insertion. Here we adapt the type A_{n-1} Schützenberger evacuation to type C_n KN tableaux, via SJDT or Baker-Lecouvey insertion. We relate the right and left key maps of a tableau via Lusztig involution.

6.1 Lusztig involution and evacuation algorithms

In type A_{n-1}, the Lusztig involution [30] on the crystal with set $\mathcal{SSYT}(\lambda,n)$ coincides with the Schützenberger involution or evacuation [13, 38, 40], Ev, and takes $T \in \mathcal{SSYT}(\lambda,n)$ to $T^{Ev} \in \mathcal{SSYT}(\lambda,n)$, whose weight is $\omega_0(\mathrm{wt}\,T)$, where ω_0 is the longest permutation of $\mathfrak{S}_n$, in the Bruhat order. Note that $\omega_0(\mathrm{wt}\,T)$ is the vector $\mathrm{wt}\,T$ in reverse order, i.e., $\omega_0(v_1,\dots,v_n) = (v_n,\dots,v_1)$. In type C_n we will work with KN tableaux instead of SSYT's. Consider $T \in \mathcal{KN}(\lambda,n)$. In this case, $T^{Ev} \in \mathcal{KN}(\lambda,n)$ and $\mathrm{wt}\,T = -\mathrm{wt}\,T^{Ev} = \omega_0^C(\mathrm{wt}\,T^{Ev})$, where ω_0^C is the longest permutation of B_n.

Definition 6.1.1. *The Lusztig involution $L : \mathfrak{B}^\lambda \to \mathfrak{B}^\lambda$ is the only set involution such that for all $i \in I$ ($I = [n-1]$ in type A_{n-1} and $I = [n]$ in type C_n):*

1. $wt(L(x)) = \omega_0(wt(x))$*, where ω_0 is the longest element of the Weyl group;*
2. $e_i(Lx) = L(f_{i'}(x))$ *and* $f_i(Lx) = L(e_{i'}(x))$ *where i' is such that $\omega_0(\alpha_i) = -\alpha_{i'}$ and α_i is the i-th simple root;*
3. $\varepsilon_i(Lx) = \varphi_{i'}(x)$ *and* $\varphi_i(Lx) = \varepsilon_{i'}(x)$.

Remark 6.1.2. *One can prove that the map defined by this three conditions is, in fact, unique [9, Chapter 5].*

For type A_{n-1} we have that ω_0 is the reverse permutation and $i' = n-i$, and for type C_n we have $\omega_0 = -\mathrm{Id}$ and $i' = i$, where Id is the identity map. In type C_n the involution can be seen as flipping the crystal upside down.

The type C_n Lusztig involution can be seen as a realization of the dual crystal:

Definition 6.1.3. *[9] Let $\mathfrak{C}$ be a connected component in the type C_n crystal G_n. The dual crystal $\mathfrak{C}^\vee$ is the crystal obtained from $\mathfrak{C}$ after reversing the direction of all arrows. Also, the if $x \in \mathfrak{C}$, then for its correspondent in $\mathfrak{C}^\vee$, $x^\vee$, we have $wt(x) = -wt(x^\vee)$.*

In type C_n, since $i' = i$ and $\omega_0 = -\mathrm{Id}$, it follows from the definition of Lusztig involution that $\mathfrak{C}$ and $\mathfrak{C}^\vee$, as crystals in G_n, have the same highest weight. Therefore, they are isomorphic. Hence the crystal $\mathfrak{B}^\lambda$ with set $\mathcal{KN}(\lambda, n)$ is self-dual. We shall see other realizations of the dual.

The complement of a tableau or a word in types A_{n-1} or C_n consists in applying ω_0 or ω_0^C, respectively, to all of its entries. In type A_{n-1}, it sends i to $n+1-i$ for all $i \in [n]$, i.e., $\omega_0(i) = n+1-i$ and in type C_n we have $\omega_0(i) = -i$. Given a SSYT, there are several algorithms, due to Schützenberger, to obtain a SSYT with the same shape whose weight is its reverse. We recall some versions of them for which one is able to find analogues for KN tableaux.

Algorithm 6.1.4.

1. *Define $w = cr(T)$.*
2. *Define $w^\star$ the word obtained by complementing its letters and writing it backwards.*
3. $T^{Ev} := P(w^\star)$.

Example 6.1.5. *In type A, the tableau* $T = \begin{array}{|c|c|c|c|}\hline 1&1&2&3\\\hline 2&3&3\\\cline{1-3} 4\\\cline{1-1}\end{array}$ *has reading* $w = 32313124$. *Then* $w^\star = 13424232$, *and the column insertion of this word is* $T^{Ev} = \begin{array}{|c|c|c|c|}\hline 1&2&2&3\\\hline 2&4&4\\\cline{1-3} 3\\\cline{1-1}\end{array}$.

In type C, consider the KN tableau $T = \begin{array}{|c|c|c|}\hline 1&3&\bar{1}\\\hline 3&\bar{3}\\\cline{1-2} \bar{3}\\\cline{1-1}\end{array}$. *Then,* $w = cr(T) = \bar{1}3\bar{3}13\bar{3}$ *and* $w^\star = 3\bar{3}\bar{1}3\bar{3}\bar{3}1$.
So now we insert $w^\star$, obtaining the following sequence of tableaux:

$$\begin{array}{|c|}\hline 3\\\hline\end{array}\quad \begin{array}{|c|}\hline 3\\\hline \bar{3}\\\hline\end{array}\quad \begin{array}{|c|}\hline 3\\\hline \bar{3}\\\hline \bar{1}\\\hline\end{array}\quad \begin{array}{|c|c|}\hline 2&\bar{2}\\\hline 3\\\cline{1-1} \bar{1}\\\cline{1-1}\end{array}\quad \begin{array}{|c|c|}\hline 2&\bar{2}\\\hline 3&\bar{1}\\\hline \bar{3}\\\cline{1-1}\end{array}\quad \begin{array}{|c|c|c|}\hline 1&2&\bar{2}\\\hline 3&\bar{1}\\\cline{1-2} \bar{3}\\\cline{1-1}\end{array} = P(w^\star).$$

Algorithm 6.1.6.

1. *Define $T^0 :=$ complement(π-rotate(T)).*
2. $T^{Ev} :=$ *rectification of* T^0.

Example 6.1.7. *In type A, consider the tableau* $T = \begin{array}{|c|c|c|c|}\hline 1&1&2&3\\\hline 2&3&3\\\cline{1-3} 4\\\cline{1-1}\end{array}$. *After π-rotation and complement we have the skew tableau* $T^0 = \begin{array}{cccc}\cline{4-4} &&&\multicolumn{1}{|c|}{1}\\\cline{2-4} &\multicolumn{1}{|c|}{2}&2&3\\\hline \multicolumn{1}{|c|}{2}&3&4&4\\\hline\end{array}$ *which, after rectification, gives the tableau* $T^{Ev} = \begin{array}{|c|c|c|c|}\hline 1&2&2&3\\\hline 2&4&4\\\cline{1-3} 3\\\cline{1-1}\end{array}$.

In type C, consider the KN tableau $T = \begin{array}{|c|c|c|}\hline 1 & 3 & \bar{1} \\ \hline 3 & \bar{3} \\ \cline{1-2} \bar{3} \\ \cline{1-1}\end{array}$. *Then,* $T_0 = \begin{array}{ccc} & & \boxed{3} \\ & \boxed{3} & \boxed{\bar{3}} \\ \boxed{1} & \boxed{\bar{3}} & \boxed{\bar{1}} \end{array}$. *So now we have to rectify this skew tableau obtaining* $T^{Ev} = \begin{array}{|c|c|c|}\hline 1 & 2 & \bar{2} \\ \hline 3 & \bar{1} \\ \cline{1-2} \bar{3} \\ \cline{1-1}\end{array}$.

Given a KN tableau (resp. SSYT) T, the algorithm characterize T^{Ev} as the unique KN tableau (resp. SSYT) Knuth equivalent to $\mathrm{wt}(T)^\star$ and coplactic equivalent to T.

In both Cartan types we have that algorithms 6.1.4 and 6.1.6 produce the same tableau since the column reading of T^0 is $w^\star$, $P(w^\star) = rect(T^0) = rect(w^\star)$, assuming that, in type C_n, T^0 is admissible. This can be concluded using the following lemma.

Lemma 6.1.8. *For type C_n, the split of a column C, $(\ell C, rC)$ is the rotation and complement of the split of the column $C^0 = complement(\pi\text{-}rotate(C))$, $(\ell C^0, rC^0)$.*

Proof. Let's say that $(\ell C, rC) = \begin{array}{|c|c|}\hline A' & A \\ \hline B & B' \\ \hline\end{array}$ where $C = \begin{array}{|c|}\hline A \\ \hline B \\ \hline\end{array}$, $\ell C = \begin{array}{|c|}\hline A' \\ \hline B \\ \hline\end{array}$ and $rC = \begin{array}{|c|}\hline A \\ \hline B' \\ \hline\end{array}$, where A and A' are the unbarred letters of the columns C and ℓC, respectively, and B and rB are the barred letters of C and rC, respectively. Note that ℓC and C share the barred part and C and rC share the unbarred part.

We have that $C^0 = \begin{array}{|c|}\hline B^0 \\ \hline A^0 \\ \hline\end{array}$ and its split $(\ell C^0, rC^0) = \begin{array}{|c|c|}\hline B^{0'} & B^0 \\ \hline A^0 & A^{0'} \\ \hline\end{array}$. Ignoring bars and counting multiplicities, the letters that appear in C and C^0 are the same. Hence $B^{0'}$ has the same letters as B', but they appear unbarred, hence $B^{0'} = B'^0$. The same happens with $A^{0'}$ and A'^0. Now it is easy to see that $(\ell C^0, rC^0)$ is obtained from $(\ell C, rC)$ rotating and complementing. In particular $(rC)^0 = \ell C^0$ and $(\ell C)^0 = rC^0$. □

We now set the Cartan type to be C_n. Given a word $w \in [\pm n]^*$, we define the $w^\star$ like in the Algorithm 6.1.4 and show that the map * preserves Knuth equivalence.

Theorem 6.1.9. *Let $v, w \in [\pm n]^*$. Then $v \sim w$ if and only if $v^\star \sim w^\star$.*

Proof. It is enough to consider v and w only one Knuth relation apart, because all other cases are obtained by composing multiple Knuth relations. It is enough to consider each transformation applied in one direction, since the other direction is the same case, after swapping the roles of v and w.

K1 Consider $v = v_p\gamma\beta\alpha v_s$, with $\gamma < \alpha \le \beta$ and $(\beta,\gamma) \ne (\bar{x}, x)$, where v_p is a prefix of v, v_s is a suffix of v, and $\gamma\beta\alpha$ are three consecutive letters of v. Then, $v \overset{K1}{\sim} w = v_p\beta\gamma\alpha v_s$. Note that $v^\star = v_s^\star\overline{\alpha}\overline{\beta}\overline{\gamma}v_p^\star$ and $w^\star = v_s^\star\overline{\alpha}\overline{\gamma}\overline{\beta}v_p^\star$, with $(\overline{\gamma},\overline{\beta}) \ne (\bar{x}, x)$ and $\overline{\beta} \le \overline{\alpha} < \overline{\gamma}$. Hence $v^\star \overset{K2}{\sim} w^\star$, so they are Knuth related.

K2 Consider $v = v_p\alpha\beta\gamma v_s$, with $\gamma \le \alpha < \beta$ and $(\beta,\gamma) \ne (\bar{x}, x)$, where v_p is a prefix of v, v_s is a suffix of v, and $\alpha\beta\gamma$ are three consecutive letters of v. Then, $v \overset{K2}{\sim} w = v_p\alpha\gamma\beta v_s$. Note that $v^\star = v_s^\star\overline{\gamma}\overline{\beta}\overline{\alpha}v_p^\star$ and $w^\star = v_s^\star\overline{\beta}\overline{\gamma}\overline{\alpha}v_p^\star$, with $(\overline{\gamma},\overline{\beta}) \ne (\bar{x}, x)$ and $\overline{\beta} < \overline{\alpha} \le \overline{\gamma}$. Hence $v^\star \overset{K1}{\sim} w^\star$, so they are Knuth related.

K3 Consider $v = v_p(y+1)\overline{y+1}\beta v_s$, with $y < \beta < \bar{y}$, where v_p is a prefix of v, v_s is a suffix of v, and $(y+1)\overline{y+1}\beta$ are three consecutive letters of v. Then, $v \overset{K3}{\sim} w = v_p\bar{y}y\beta v_s$. Note that $v^\star = v_s^\star\overline{\beta}(y+1)\overline{y+1}v_p^\star$ and $w^\star = v_s^\star\overline{\beta}\bar{y}yv_p^\star$, with $y < \beta < \bar{y}$. Hence $v^\star \overset{K4}{\sim} w^\star$, so they are Knuth related.

K4 Consider $v = v_p\alpha\bar{x}xv_s$, with $x < \alpha < \bar{x}$, where v_p is a prefix of v, v_s is a suffix of v, and $\alpha\bar{x}x$ are three consecutive letters of v. Then, $v \overset{K4}{\sim} w = v_p\alpha(x+1)\overline{x+1}v_s$. Note that $v^\star = v_s^\star\bar{x}x\overline{\alpha}v_p^\star$ and $w^\star = v_s^\star(x+1)\overline{x+1}\overline{\alpha}v_p^\star$, with $x < \alpha < \bar{x}$. Hence $v^\star \overset{K3}{\sim} w^\star$, so they are Knuth related.

K5 Consider w and $\{z,\bar{z}\} \in w$ such that $w \overset{K5}{\sim} w \setminus \{z,\bar{z}\}$. It is clear to see that a word v breaks the $1CC$ at z if and only if $v^\star$ breaks the $1CC$ at z. So, if w is non admissible and all its factors are admissible then the same will happen to $w^\star$, because all of its factors are obtained after applying * to a factor of w. So we have that $w^\star \overset{K5}{\sim} w^\star \setminus \{z,\bar{z}\}$.

Hence the word operator * preserves Knuth equivalence. □

Consider a KN tableau T with column reading w. The column reading of the tableau obtained after applying Algorithm 6.1.4 to T is Knuth-related to $w^\star$, because both give the same tableau if inserted. Since * is an involution ($(w^\star)^\star = w$), if we apply the algorithm again we will get a tableau whose column reading, by the last theorem, is Knuth equivalent to $(w^\star)^\star = w$, hence we will have T again. So Algorithm 6.1.4 is an involution. Next we conclude that algorithms 6.1.4 and 6.1.6 is a realization of the Lusztig involution for type C_n.

Theorem 6.1.10. *Let $w \in [\pm n]^*$. The connected component of the crystal G_n that contains the word w is isomorphic to the one that contains the word $w^\star$. Therefore $P(w)$ and $P(w^\star)$ have the same shape and weights of opposite sign. Moreover, the two crystals are dual of each other and the * map is a realization of the dual crystal.*

Proof. Remember the crystal operators e_i and f_i from the definition of crystal. Note that $(f_i(w))^\star = e_i(w^\star)$, because in the signature rule applied to w and $w^\star$, the distance of the leftmost unbracketed $+$ of w to the beginning of the word is equal to the distance of the rightmost unbracketed $-$ of $w^\star$ to the end of this word. Hence, the letter that changes when applying f_i to w is the complement of the letter that changes when applying e_i to $w^\star$, and the letter obtained on their position after applying the crystal operators are also complement of each other. Hence the crystal that contains the word $w^\star$ is the dual to the one that contains w. But the crystal that contains w is self-dual, hence the crystals that contains any of the words are isomorphic. From [25, Theorem 3.2.8] $P(w)$ and $P(w^\star)$ have the same shape. □

6.2 Right and left keys and Lusztig involution

The next result shows that the right and left key maps defined for KN tableaux anticommutes with the Lusztig involution. The evacuation of the right key of a tableau is the left key of the evacuation of the same tableau.

Proposition 6.2.1. *Let T be a KN tableau and Ev the type C_n Lusztig involution. Then*

$$K_+(T)^{Ev} = K_-(T^{Ev}).$$

Proof. Since the tableaux $K_+(T)$ and $K_-(T^{Ev})$ are key tableaux, they are completely determined by their weights. Then we just need to prove that their weights are symmetric.

Fix a column C of $K_+(T)$. There is a frank word w, Knuth related to $cr(T)$, such that C is the right column of the first column of w. Let's say the w_k is the first column of w. From Proposition 6.1.9, $w^\star$ is Knuth related to $cr(T)^\star$, hence $P(w^\star) = T^{Ev}$. Also note that the $w^\star$ has the same number of columns of each length as w, hence it is a frank word, and its last column is $w_k^\star$. Note that Lemma 6.1.8 implies that if v is an admissible column, then $l(v^\star) = (rv)^\star$. So we have that $l(w_k^\star) = (rw_k)^\star$ is a column of $K_-(T^{Ev})$. Therefore, for each column C of $K_+(T)$ there is a column of $K_-(T^{Ev})$ whose weight is $\omega_0(\mathrm{wt}(C))$, hence $K_+(T)$ and $K_-(T^{Ev})$ have symmetric weights. □

Remark 6.2.2. *Using Proposition 6.2.1 and the definition of the Lusztig involution, it is now clear that the tableau weights in $\mathfrak{B}_v$ and in $\mathfrak{B}_{-v}^{op}$ are symmetric.*

Chapter 7

Final remarks and open questions

In this chapter we discuss some unfinished work and open problems related to the topics presented in this thesis.

7.1 Type C_n Fu-Lascoux non-symmetric Cauchy kernel

Let's start by recalling the type C_n Fu-Lascoux non-symmetric Cauchy kernel:

$$\frac{\prod_{1\le i<j\le n}(1-x_ix_j)}{\prod_{i,j=1}^{n}(1-x_iy_j)\prod_{i,j=1}^{n}(1-x_i/y_j)} = \sum_{v\in\mathbb{N}^n}\widehat{\kappa}_v(x_1,\dots,x_n)\kappa_{-v}(y_1,\dots,y_n)$$
$$\Leftrightarrow \prod_{1\le i,j\le n}(1-x_iy_j)^{-1}\prod_{1\le i\le j\le n}(1-x_iy_j^{-1})^{-1} = \prod_{1\le i<j\le n}(1-x_ix_j)^{-1}\sum_{v\in\mathbb{N}^n}\hat{\kappa}_v(x)\kappa_{-v}(y).$$

In this section we present a combinatorial interpretation for each side of the last identity, and propose an algorithm that relates both combinatorial interpretations.

7.1.1 Warm up for the combinatorial interpretations

Let λ_v be the only partition on the $\mathfrak{S}_n$-orbit of v, for $v\in\mathbb{N}^n$. The next lemma is a reformulation of the tableau criterion for the Bruhat order on $\mathfrak{S}_n$.

Lemma 7.1.1. *Let $u=(u_1,\dots,u_n), v=(v_1,\dots,v_n)\in\mathfrak{S}_n\lambda$.*

$$u\le v \text{ if and only if } \forall i\in[n], \lambda_{(u_1,\dots,u_i)}\supseteq\lambda_{(v_1,\dots,v_i)}.$$

Proof. From Theorem 3.4.4, restricted to the symmetric group, we have that $u\le v$ if and only if $K(u)\le K(v)$. For all $i\in[n]$, if we know the first i entries of u then we know all entries of $K(u)$ less or equal than i. Also, the shape occupied by this entries is exactly $\lambda_{(u_1,\dots,u_i)}$. The shape occupied by the entries less or equal than i in $K(u)$ contains the shape occupied by the entries less or equal than i in $K(v)$. Hence $K(u)\le K(v)$ if and only if $\lambda_{(u_1,\dots,u_i)}\supseteq\lambda_{(v_1,\dots,v_i)}$, $\forall i\in[n]$. □

Due to the natural embedding of B_n into $\mathfrak{S}_{2n}$, this can be extended to B_n. Given a $v \in \mathbb{Z}_n$, its correspondent in $\mathbb{N}_{2n}$ is the vector $v^{\#}$, with $2n$ entries indexed by $[\pm n]$, where $v_i^{\#} = \begin{cases} 0 \text{ if } i \times v_{|i|} \leq 0 \\ |v_{|i|}| \text{ otherwise} \end{cases}$.
For instance, if $u = (4,1,\overline{3},\overline{2},3)$. Then $u^{\#} = (4,1,0,0,3,0,2,3,0,0)$.

Embedding B_n into $\mathfrak{S}_{2n}$ and applying the previous lemma, we have the following:

Corollary 7.1.2. *Let $u = (u_1,\dots,u_n), v = (v_1,\dots,v_n) \in \mathfrak{B}_n\lambda$.*

$$u \leq v \text{ if and only if } \forall i \in [\pm n], \lambda_{(u_1^{\#},\dots,u_i^{\#})} \supseteq \lambda_{(v_1^{\#},\dots,v_i^{\#})}.$$

The next proposition is the conjugate version of [13, Proposition 7, Appendix *A*].

Proposition 7.1.3. *Let T be a SSYT tableau of shape λ. Let μ/ν be a skew diagram with same number of rows of each length as T. Then there is a unique KN skew tableau S with shape μ/ν that rectifies to T.*

Proof. The number of skew tableaux of shape μ/ν that rectify to T of shape λ is given by the Littlewood-Richardson coefficient $c^{\mu}_{\lambda\,\nu}$.

Remember the identity $c^{\mu}_{\lambda\,\nu} = c^{\mu'}_{\lambda'\,\nu'}$ [1, 2, 15], where μ', ν' and λ' are the conjugate diagrams of μ, ν and λ, respectively. Given λ, μ and ν like in the statement we have that λ' and μ'/ν' have the same number of columns of each length, hence from [13, Proposition 7, Appendix *A*], we have that $c^{\mu'}_{\lambda'\,\nu'} = 1$. So $c^{\mu}_{\lambda\,\nu} = 1$. □

We can generalize the last proposition to KN tableaux, obtaining the following:

Proposition 7.1.4. *Let T be a KN tableau of shape λ. Let μ/ν be a skew diagram with same number of rows of each length as T. Then there is a unique KN skew tableau S with shape μ/ν that rectifies to T.*

Proof. If T is the Yamanouchi tableau $K(\lambda)$ and $S \in \mathcal{KN}(\mu/\nu,n)$ rectifies to $K(\lambda)$, then, since S and $K(\lambda)$ have the same number of cells, all entries of S are unbarred, hence S is a semistandard skew tableau. So, it follows from 7.1.3 that S exists and is unique. If T is not the Yamanouchi tableau, note that T is crystal connected to $K(\lambda)$ and from [25, Theorem 6.3.8] we have that the SJDT slides commute with the action of the crystal operators. Consider Y_λ the only tableau on the skew shape μ/ν that rectifies to $K(\lambda)$, which exists due to 7.1.3. Since S rectifies to T, which is crystal connected to $K(\lambda)$, and Y_λ rectifies to $K(\lambda)$, then S is crystal connected to Y_λ and the path has same sequence of colours as the one from T to $K(\lambda)$. Hence S exists and is uniquely defined. □

7.1.2 Combinatorial interpretation of the left hand side of the identity

Let's look at the left hand side of type C_n Fu-Lascoux non-symmetric Cauchy kernel:

$$\prod_{1\leq i,j\leq n}(1-x_iy_j)^{-1}\prod_{1\leq i\leq j\leq n}(1-x_iy_j^{-1})^{-1}.$$

In the same spirit of the combinatorial proof of the Cauchy identity via RSK correspondence [13, Section 4], we can see that this is the generating function of all biwords in which the top column only have positive letters and if i is the top letter of a column then the bottom letter is less or equal than $\bar{i}$. If a word satisfies this, we say it satisfies the *LHS (left hand side) condition*. In same spirit of the biwords from E_n^r, from Subsection 4.4.1, we will consider our biletters ordered lexicographically, obtaining:

$$\begin{pmatrix} 1 & \dots & 1 & 2 & \dots & 2 & \dots & n \\ x_1 & \leq & x_i & x_{i+1} & \leq & x_{i'} & \dots & x_{i''} \end{pmatrix}$$

Note that the top row separates the bottom row into weakly increasing sequence of words.

7.1.3 Combinatorial interpretation of the right hand side of the identity

Now, recall the right hand side of type C_n Fu-Lascoux non-symmetric Cauchy kernel:

$$\prod_{1\leq i<j\leq n}(1-x_ix_j)^{-1}\sum_{v\in\mathbb{N}^n}\hat{\kappa}_v(x)\kappa_{-v}(y).$$

Our combinatorial interpretation of this side will be a tuple consisting of three entries: a biword, a reverse SSYT and a KN tableau. Our biword will have billeters ordered lexicographically, and all entries in $[n]$, and with the top entry bigger than the respective bottom entry. Before translating the key polynomial and the Demazure atom as generating functions of some sets of tableaux, we will start by doing an algebraic manipulation, and with that in mind we will need the following lemma:

Lemma 7.1.5. *[6, Proposition 2.3.4] Let $u,v\in B_n\lambda$. If $u\leq v$ then $-v\leq -u$.*

Hence:

$$\begin{aligned}\sum_{v\in\mathbb{N}^n}\hat{\kappa}_v(x)\kappa_{-v}(y) &= \sum_{v\in\mathbb{N}^n}\hat{\kappa}_v(x)\sum_{u\leq -v}\hat{\kappa}_u(y)\\ &= \sum_{v\in\mathbb{N}^n}\sum_{u\leq -v}\hat{\kappa}_v(x)\hat{\kappa}_u(y)\\ &= \sum_{u\in\mathbb{Z}^n}\sum_{\substack{v\in\mathbb{N}^n\\ u\leq -v}}\hat{\kappa}_v(x)\hat{\kappa}_u(y)\\ &= \sum_{u\in\mathbb{Z}^n}\sum_{\substack{v\in\mathbb{N}^n\\ v\leq -u}}\hat{\kappa}_v(x)\hat{\kappa}_u(y)\end{aligned}$$

Using the tableau criterion for the Bruhat order on Lemma 7.1.1, we can find a maximal u' such that $u'\in\mathbb{N}^n\subset\mathbb{Z}^n$ and $u'\leq -u$ in the following way:

We will determine the entries of u' recursively. The first entry of u' is $-u_1$ if $-u_1$ is positive and λ_n otherwise. For i going from 2 to n, u'_i is the minimal not yet used entry of λ such that $\lambda_{(-u_1^\#,\dots,-u_i^\#)}\supseteq\lambda_{(u'_1,\dots,u'_i)}$. Hence we find a vector u' such that

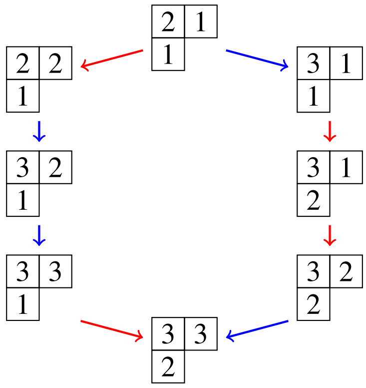
$$\sum_{\substack{v\in\mathbb{N}^n\\ v\leq -u}}\hat{\kappa}_v(x)=\kappa_{u'}(x).$$

Example 7.1.6. *Consider $-u=(1,-4,4,0,-3)$. Then $u'=(1,0,4,3,4)$.*

7.1.4 Reverse SSYT's

Let λ be a partition and consider the type A_{n-1} crystal $\mathcal{SSYT}(\lambda,n)$. For every SSYT in $\mathcal{SSYT}(\lambda,n)$, π-rotate it, obtaining a reverse skew SSYT. Now rectify each tableau, using the *jeu de taquin* for reverse SSYT, obtaining a straight reverse SSYT's. Now we have a crystal of reverse SSYT's. We call this crystal $\mathcal{RSSYT}(\lambda,n)$. Their reading word also follows the signature rule if applied backwards (swapping left and right).

Example 7.1.7. *The type A_2 crystal of reverse SSYT's $\mathfrak{B}^{(2,1,0)}$ is represented by the graph, where red represents f_1 and blue represents f_2.*



7.1.5 Another cocrystal of KN tableaux

Inspired by Lascoux' double crystal graph, we create a cocrystal, similar to the one in Section 4.4. Although, the cocrystal here is created based on RSK correspondence, instead of the dual RSK, meaning that the crystal operators on the cocrystal are *jeu de taquin* moves between consecutive rows.

Fix a partition λ and a KN tableau $T \in \mathcal{KN}(\lambda,n)$. Starting from the bottom, label n rows with the numbers 1 to n. In these rows we will place some cells (possibly 0) that form a skew tableau.

Now we create a map from $\mathcal{RSSYT}(\lambda,n)$ to cocrystal consisting of skew KN tableaux connected to T via SJDT. We will also force all skew tableau to have columns of length 1, by moving all rows horizontally via SJDT. So all of skew tableau in the cocrystal will only have columns of length 1 and are SJDT connected to T. This forces the reverse of the row reading word (the word obtain after concatenating each row, from bottom to top) of our skew tableaux to be Knuth equivalent to T. This happens because the reverse of the row reading word of the skew tableaux will be its column reading word, and since the skew tableau rectifies to T, the column reading word of the skew tableau is Knuth related to the column reading word of T.

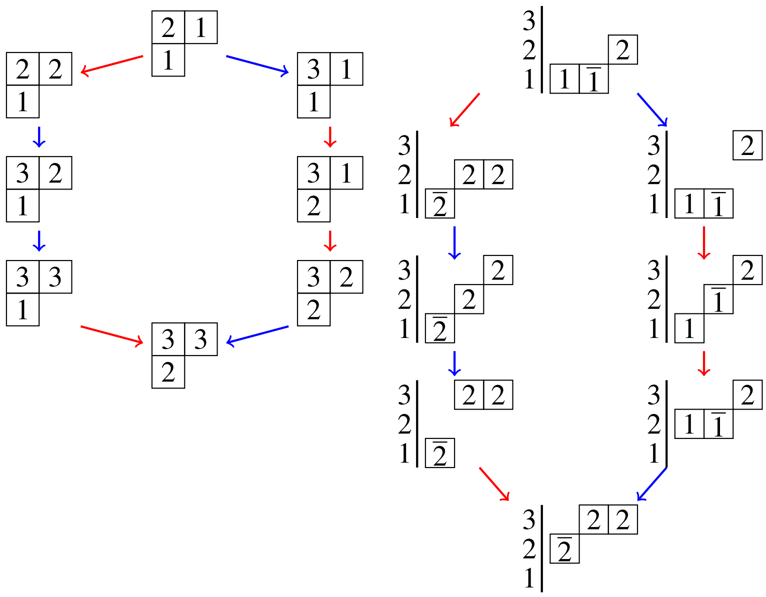
The highest weight of the cocrystal is a KN skew tableau Knuth equivalent to T, with exactly one cell per column, and with λ_i cells in the i-th row. This tableau is unique thanks to Proposition 7.1.4. It is obtained from T by doing some horizontal SJDT slides.

Given a tableau Q in $\mathcal{RSSYT}(\lambda,n)$, if we can apply f_i to it (i.e. $f_i(Q) \neq 0$) then the correspondent of $f_i(Q)$ in the cocrystal is obtained from the correspondent of Q by taking a cell from the i-th row to the $i+1$-th row via SJDT. An arrow of colour i (just like in the the crystal) connects both skew tableaux. Thus, we create a graph isomorphic (as a coloured graph) to $\mathcal{SSYT}(\lambda,n)$, in which each vertex is

Knuth equivalent to T and the number of cells in the row i is the weight of i in the corresponding vertex on the crystal graph. This is done by Lascoux in [22] when T is a SSYT. The extension of the cocrystal to type C_n objects (even though it is still a type A crystal) follows the same idea presented in Section 4.4.

Hence, given a KN tableau and a RSSYT with same shape, we have a unique skew tableau without two cells in the same column. Note that in Section 4.4 we had something similar to this, where given a skew SSYT T we could get a biword, and RSK^* sent that word to a pair consisting of two conjugated tableaux, one identifying the Knuth class of T and the other identifying its skew shape.

Example 7.1.8. *Let* $T = \begin{array}{|c|c|}\hline 2 & 2 \\ \hline \bar{2} \\ \cline{1-1}\end{array}$. *Here we have, side by side, the crystal* $\mathcal{RSSYT}(\lambda, n)$ *and the cocrystal associated to it with respect to* T*, where the crystal operators are SJDT moves on consecutive rows. To the left of each tableau in the cocrystal we have a numbering of the rows, in order to be easier to keep track of its row lengths. They are isomorphic type* A_2 *crystal.*



7.1.6 The algorithm

In this subsection we present an algorithm that given L, a biword with positive entries, where the top letter is strictly bigger then the bottom letter, a KN tableau T with right key $K(u)$ and a reverse SSYT Q in the Demazure crystal $\mathfrak{B}_{u'}$, returns a biword that satisfies the LHS condition. The creation of the biword is somewhat close to Sundaram's combinatorial bijection for the type C_n symmetric Cauchy kernel in [41].

The algorithm relies heavily on the inverse of the Baker-Lecouvey insertion [5, 25]. The inverse algorithm accepts two possible inputs, each one having its output:

- Case 1: The input is a KN tableau T and one of its corners. The algorithm returns a tableau $\widetilde{T}$ without that corner and a number $y \in [\pm n]$, such that if we insert y in $\widetilde{T}$ we recover T.
- Case 2: The input is a KN tableau T and a cell outside of T such that the shape of T together with that cell is a Ferrers diagram of shape λ. The inverse Baker-Lecouvey insertion returns a

tableau $\widetilde{T}$ of shape λ and a number $y \in [\pm n]$, such that if we insert y in $\widetilde{T}$ we recover T. Note that, in this case, Baker-Lecouvey insertion of y in $\widetilde{T}$ incurs in a loss of cells.

The algorithm starts by looking to the smallest number in L or Q, x. In each step of the algorithm there are two different cases, with the second case having two subcases:

- Case 1: If x is in Q, delete the rightmost x in Q, and, via the inverse of the Baker-Lecouvey insertion, we remove the entry in T in the same position, obtaining a y, and we write $\begin{matrix} x \\ y \end{matrix}$ in the biword.
- Case 2: if x is in L and not in Q, we row insert in Q the biggest x' paired with x in L, obtaining a tableau one cell bigger, and let y be the letter that causes that change on the shape. Note that y is uniquely determined by the reverse Baker-Lecouvey insertion the case where the insertion makes the tableau lose cells, hence its reverse makes the tableau gain cells, which is this case. So
  - Case 2a: if $-y \geq x$, we add $\begin{matrix} x \\ y \end{matrix}$ in the biword.
  - Case 2b: if $-y < x$ we add $\begin{matrix} x & x' \\ \bar{x} & x \end{matrix}$ to the biword (the second biletter may be misplaced, and needs to be corrected later). Q, T returns to what they were in the beginning of this case, and L loses the biletter $\begin{matrix} x' \\ x \end{matrix}$.

Example 7.1.9. *Let* $T = \begin{array}{|c|c|}\hline 2 & 2 \\ \hline \bar{2} \\ \cline{1-1}\end{array}$. *Then* $u = wt(K_+(T)) = (-1, 2, 0)$. *Then* $u' = (1, 0, 2)$. *Let* $Q = \begin{array}{|c|c|}\hline 3 & 1 \\ \hline 2 \\ \cline{1-1}\end{array} \in \mathfrak{B}_{u'}$ *and let* $L = \begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix}$.

We will now apply the algorithm:

<table>
<tr><th>Case applied</th><th>T</th><th>Q</th><th>L</th><th>biword</th></tr>
<tr><td></td><td>$\begin{array}{|c|c|}\hline 2 & 2 \\ \hline \bar{2} \\ \cline{1-1}\end{array}$</td><td>$\begin{array}{|c|c|}\hline 3 & 1 \\ \hline 2 \\ \cline{1-1}\end{array}$</td><td>$\begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix}$</td><td>$\emptyset$</td></tr>
<tr><td>Case 1</td><td>$\begin{array}{|c|}\hline 2 \\ \hline \bar{1} \\ \hline\end{array}$</td><td>$\begin{array}{|c|}\hline 3 \\ \hline 2 \\ \hline\end{array}$</td><td>$\begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix}$</td><td>$\begin{matrix} 1 \\ 1 \end{matrix}$</td></tr>
<tr><td>Case 2a, $y = \bar{1}$</td><td>$\begin{array}{|c|c|}\hline 2 & 2 \\ \hline \bar{2} \\ \cline{1-1}\end{array}$</td><td>$\begin{array}{|c|c|}\hline 3 & 3 \\ \hline 2 \\ \cline{1-1}\end{array}$</td><td>$\begin{pmatrix} 2 \\ 1 \end{pmatrix}$</td><td>$\begin{matrix} 1 & 1 \\ 1 & \bar{1} \end{matrix}$</td></tr>
<tr><td>Case 2a, $y = \bar{1}$</td><td>$\begin{array}{|c|c|c|}\hline 1 & 2 & 2 \\ \hline \bar{2} \\ \cline{1-1}\end{array}$</td><td>$\begin{array}{|c|c|c|}\hline 3 & 3 & 2 \\ \hline 2 \\ \cline{1-1}\end{array}$</td><td>$\emptyset$</td><td>$\begin{matrix} 1 & 1 & 1 \\ 1 & \bar{1} & \bar{1} \end{matrix}$</td></tr>
<tr><td>Case 1, four times</td><td>$\emptyset$</td><td>$\emptyset$</td><td>$\emptyset$</td><td>$\begin{matrix} 1 & 1 & 1 & 2 & 2 & 3 & 3 \\ 1 & \bar{1} & \bar{1} & 1 & \bar{2} & 2 & 2 \end{matrix}$</td></tr>
</table>

And now we shall see a small example in which we have the case $2b$.

Example 7.1.10. *Let* $T = \emptyset$*, the empty tableau. Then* $u = wt(K_+(T)) = (0) = u'$ *and* $Q = \emptyset$ *and let* $L = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$.

Case applied	T	Q	L	*biword*
	$\emptyset$	$\emptyset$	$\begin{pmatrix}3\\2\end{pmatrix}$	$\emptyset$
Case 2b, $y=\overline{1}$	$\emptyset$	$\emptyset$	$\emptyset$	$\begin{matrix}2 & 3\\2 & \overline{2}\end{matrix}$

Conjecture 7.1.11. *The algorithm is a bijection between all tuples consisting of a biword with positive entries where the top letter is strictly bigger then the bottom letter, a KN tableau in $\hat{\mathfrak{B}}_u$ and a reverse SSYT in $\mathfrak{B}_{u'}$, and biwords that satisfies the LHS condition.*

If this conjecture is true, then we have a combinatorial proof of the type C_n Fu-Lascoux non-symmetric Cauchy kernel:

$$\prod_{1\le i,j\le n}(1-x_iy_j)^{-1}\prod_{1\le i\le j\le n}(1-x_iy_j^{-1})^{-1}=\prod_{1\le i<j\le n}(1-x_ix_j)^{-1}\sum_{\nu\in\mathbb{N}^n}\hat{\kappa}_\nu(x)\kappa_{-\nu}(y).$$

In fact, it is not even clear that the output of the algorithm always satisfies the LHS condition. Here we will not prove the conjecture, and we will not even prove that its output satisfies the LHS condition. However, when L is empty, then the output of the algorithm satisfies the LHS condition:

Lemma 7.1.12. *Let L be empty and $u\in\mathbb{Z}^n$. If $T\in\hat{\mathfrak{B}}_u$ and $Q\in\mathfrak{B}_{u'}$ LHS condition holds for the biword produced by the algorithm.*

Proof. Remember the double cocrystal in Subsection 7.1.5.

Given T and Q, consider the skew tableau obtained from T and Q. The biword obtained from T and Q is found in this skew tableau, because each biletter represents a row and an entry of each cell of the skew tableau. In particular, if $T=K(u)$ and $Q=K(u')$, the LHS condition holds because of the way u' was constructed.

If $T\le K(u)$ by entrywise comparison, the skew tableau associated to T with Q-symbol $K(u')$ is, entrywise, smaller or equal than the skew tableau associated to $K(u)$ with Q-symbol $K(u')$. Hence, the LHS condition holds.

Now we assume $T=K(u)$. Fix an element Q in $\mathfrak{B}_{u'}$. Since $Q\le K(u')$, the skew tableau associated to Q is obtained from the skew tableau with Q-symbol $K(\lambda)$ by pushing less cells to higher rows than the skew tableau with Q-symbol $K(u')$, which satisfies. Hence, the LHS condition holds fro the biword obtained from $K(u)$ and $Q\in\mathfrak{B}_{u'}$.

So, LHS condition holds for the biword obtained from $T\in\hat{\mathfrak{B}}_u$ and $Q\in\mathfrak{B}_{u'}$.

□

7.2 Further questions

7.2.1 Types *B* and *D* Kashiwara-Nakashima tableaux

In [20], Kashiwara-Nakashima also presented two other families of tableaux, compatible with a type B and type D crystal structure. Lecouvey, in [26], explored the crystal of these families of tableaux and endowed them with a plactic monoid compatible with a RSK correspondence and an insertion algorithm. In fact, for type B he even introduced a *jeu de taquin*. Hence, a natural question is to ask whether the algorithms developed in this thesis, to compute right or left keys with or without *jeu de taquin*, are applicable to these types.

7.2.2 Type C_n semi skyline augmented filling

In [32], Mason showed that Demazure atoms are specializations of non-symmetric Macdonald polynomials of type A_{n-1} with $q = t = 0$. This allowed us to use the shapes of semi-skyline augmented fillings, in the combinatorial formula of non-symmetric Macdonald polynomials [14], which are in weight preserving bijection with semi standard Young tableaux, to detect the right keys. It would be interesting to obtain a similar object for a KN tableau in type C_n. For example, semi-skyline augmented fillings have been instrumental to obtain a RSK type bijective proof [3] for the Lascoux non-symmetric Cauchy identity in type A_{n-1} [22]. Such a generalization of skyline fillings for type C could also lead to a combinatorial formula for some specialization of nonsymmetric Macdonald polynomials in type C_n.

References

[1] Azenhas, O. (1999). The admissible interval for the invariant factors of a product of matrices. *Linear and Multilinear Algebra*, 46(1-2):51–99.

[2] Azenhas, O., Conflitti, A., and Mamede, R. (2009). Linear time equivalence of Littlewood-Richardson coefficient symmetry maps. *Discrete Mathematics & Theoretical Computer Science*, pages 127–144.

[3] Azenhas, O. and Emami, A. (2015). An analogue of the Robinson–Schensted–Knuth correspondence and non-symmetric Cauchy kernels for truncated staircases. *European Journal of Combinatorics*, 46:16–44.

[4] Azenhas, O. and Mamede, R. (2012). Key polynomials of type C. Slides from the presention in The 69th Séminaire Lotharingien de Combinatoire, Strobl, http://www.mat.uc.pt/∼oazenhas/KC.pdf.

[5] Baker, T. H. (2000). An insertion scheme for C_n crystals. In *Physical Combinatorics*, pages 1–48. Springer.

[6] Bjorner, A. and Brenti, F. (2006). *Combinatorics of Coxeter groups*, volume 231. Springer Science & Business Media.

[7] Brubaker, B., Buciumas, V., Bump, D., and Gustafsson, H. (2021). Colored five-vertex models and Demazure atoms. *Journal of Combinatorial Theory, Series A*, 178:105–354.

[8] Buciumas, V. and Scrimshaw, T. (2021). Quasi-solvable lattice models for Sp_{2n} and SO_{2n+1} Demazure atoms and characters. *arXiv preprint arXiv:2101.08907*.

[9] Bump, D. and Schilling, A. (2017). *Crystal bases: representations and combinatorics*. World Scientific Publishing Company.

[10] Choi, S.-I. and Kwon, J.-H. (2018). Lakshmibai-Seshadri paths and non-symmetric Cauchy identity. *Algebras and Representation Theory*, 21(6):1381–1394.

[11] De Concini, C. (1979). Symplectic standard tableaux. *Advances in Mathematics*, 34(1):1–27.

[12] Fu, A. and Lascoux, A. (2009). Non-symmetric Cauchy kernels for the classical groups. *Journal of Combinatorial Theory, Series A*, 116(4):903–917.

[13] Fulton, W. (1997). *Young tableaux: with applications to representation theory and geometry*, volume 35. Cambridge University Press.

[14] Haglund, J., Haiman, M., and Loehr, N. (2008). A combinatorial formula for nonsymmetric Macdonald polynomials. *American Journal of Mathematics*, 130(2):359–383.

[15] Hanlon, P. and Sundaram, S. (1992). On a bijection between Littlewood-Richardson fillings of conjugate shape. *Journal of Combinatorial Theory, Series A*, 60(1):1–18.

[16] Heo, T.-H. and Kwon, J.-H. (2020). Combinatorial Howe duality of symplectic type. *arXiv:2008.05093*, pages 1–36.

[17] Jacon, N. and Lecouvey, C. (2020). Keys and Demazure crystals for Kac–Moody algebras. *Journal of Combinatorial Algebra*, 4(4):325–358.

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

[19] Kashiwara, M. (1994). On crystal bases, Representations of groups (Banff, AB), 155–197. In *CMS Conf. Proc.*, volume 16.

[20] Kashiwara, M. and Nakashima, T. (1994). Crystal graphs for representations of the q-analogue of classical Lie algebras. *Journal of Algebra*, 165:295–345.

[21] King, R. C. (1976). Weight multiplicities for the classical groups. In *Group theoretical methods in physics*, pages 490–499. Springer.

[22] Lascoux, A. (2003). Double crystal graphs. In *Studies in memory of Issai Schur*, pages 95–114. Springer.

[23] Lascoux, A., Leclerc, B., and Thibon, J.-Y. (1995). Crystal graphs and q-analogues of weight multiplicities for the root system A_n. *Letters in Mathematical Physics*, 35(4):359–374.

[24] Lascoux, A. and Schützenberger, M.-P. (1990). Keys and standard bases. *Invariant theory and tableaux*, (19):125–144.

[25] Lecouvey, C. (2002). Schensted-type correspondence, plactic monoid, and jeu de taquin for type C_n. *Journal of Algebra*, 247(2):295–331.

[26] Lecouvey, C. (2007). Combinatorics of crystal graphs for the root systems of types A_n,B_n, C_n, D_n and G_2. In *Combinatorial aspect of integrable systems*, pages 11–41. Mathematical Society of Japan.

[27] Lenart, C. (2012). From macdonald polynomials to a charge statistic beyond type a. *Journal of Combinatorial Theory, Series A*, 119(3):683–712.

[28] Lenart, C. and Lubovsky, A. (2015). A generalization of the alcove model and its applications. *Journal of Algebraic Combinatorics*, 41(3):751–783.

[29] Littelmann, P. (1995). Crystal graphs and Young tableaux. *Journal of Algebra*, 175(1):65–87.

[30] Lusztig, G. (2010). *Introduction to quantum groups*. Springer Science & Business Media.

[31] Mason, S. (2007). An explicit construction of type a Demazure atoms. *Journal of Algebraic Combinatorics*, 29:295–313.

[32] Mason, S. (2008). A decomposition of Schur functions and an analogue of the Robinson–Schensted–Knuth algorithm. *Séminaire Lotharingien de Combinatoire*, 57:B57e.

[33] Proctor, R. A. (1980). *Interactions between combinatorics, Lie theory and algebraic geometry via the Bruhat orders*. PhD thesis, Massachusetts Institute of Technology.

[34] Santos, J. M. (2020). Symplectic keys and Demazure atoms in type C. *Séminaire Lotharingien de Combinatoire - FPSAC 2020*, 84B, 12 pages.

[35] Santos, J. M. (2021a). Symplectic keys–Type C Willis' direct way. *arXiv preprint arXiv:2111.11160, 30 pages*.

[36] Santos, J. M. (2021b). Symplectic keys and Demazure atoms in type C. *Electronic Journal of Combinatorics*, 28(2), 33 pages.

[37] Santos, J. M. (2021c). Symplectic right keys - Type C Willis' direct way. *Séminaire Lotharingien de Combinatoire - FPSAC 2021*, 85B, 12 pages.

[38] Schützenberger, M. (1963). Quelques remarques sur une construction de schensted. *Mathematica Scandinavica*, 12:117–128.

[39] Sheats, J. (1999). A symplectic jeu de taquin bijection between the tableaux of King and of De Concini. *Transactions of the American Mathematical Society*, 351(9):3569–3607.

[40] Stanley, R. P. (1999). Enumerative combinatorics, volume 2 (with appendix by S. Fomin). *Cambridge Studies in Advanced Mathematics. Cambridge University Press,.*

[41] Sundaram, S. (1990). The Cauchy identity for sp_{2n}. *Journal of Combinatorial Theory, Series A*, 53(2):209–238.

[42] Willis, M. (2011). A direct way to find the right key of a semistandard Young tableau. *Annals of Combinatorics*, 17:393–400.