[12] P. Littelmann. "Crystal graphs and Young tableaux". In: *Journal of Algebra* 175.1 (1995), pp. 65–87.
[13] S. Mason. "An explicit construction of type A Demazure atoms". In: *Journal of Algebraic Combinatorics* 29 (2007), pp. 295–313.
[14] J. M. Santos. "Symplectic keys and Demazure atoms in type C". In: *arXiv preprint arXiv:1910.14115, accepted in Electron. J. Comb.* (2019).
[15] J. M. Santos. "Symplectic keys and Demazure atoms in type C". In: *Séminaire Lotharingien de Combinatoire - FPSAC 2020* 84B (2020).
[16] J. Sheats. "A symplectic jeu de taquin bijection between the tableaux of King and of De Concini". In: *Transactions of the American Mathematical Society* 351.9 (1999), pp. 3569–3607.
[17] M. Willis. "A Direct Way to Find the Right Key of a Semistandard Young Tableau". In: *Annals of Combinatorics* 17 (2011), pp. 393–400.

João Miguel Santos
CMUC, Department of Mathematics, University of Coimbra, Apartado 3008, 3001–454 Coimbra, Portugal
E-mail address: jmsantos@mat.uc.pt

Pré-Publicações do Departamento de Matemática
Universidade de Coimbra
Preprint Number 21–11

SYMPLECTIC RIGHT KEYS - TYPE C WILLIS' DIRECT WAY

JOÃO MIGUEL SANTOS

Abstract: It is known that the right key of a Kashiwara-Nakashima tableau can be computed using the Lecouvey-Sheats symplectic *jeu de taquin*. Motivated by Willis' direct way of computing type A right keys, we also give a way of computing symplectic right keys without the use of *jeu de taquin*.

En type C, la clé droite d'un tableau de Kashiwara-Nakashima peut se calculer en utilisant le *jeu de taquin* symplectique de Lecouvey-Sheats. Motivés par la manière directe de Willis afin de calculer les clés droites du type A, nous proposons également une méthode de calculer les clés droites symplectiques sans utiliser le *jeu de taquin*.

Keywords: Direct way, Symplectic keys, Right key map, Symplectic *jeu de taquin*.
Math. Subject Classification (2000): 05E05, 05E10, 17B37.

1. Introduction

Symplectic tableaux provide the monomial weight generators for the characters of the symplectic Lie algebra $sp(2n, \mathbb{C})$. Given a partition λ, symplectic Kashiwara-Nakashima tableaux [7], a variation of De Concini tableaux [5], of shape λ are endowed with a crystal structure $\mathfrak{B}_\lambda$ compatible with a plactic monoid and sliding algorithms, studied by Lecouvey in terms of crystal isomorphisms [10]. Let $W\lambda$ be the orbit of λ, where W is the type C_n Weyl group. Type C_n Demazure characters are indexed by vectors v in $W\lambda$ and can be seen as "partial" characters. Kashiwara [8] and Littelmann [12] have shown that they can be obtained by summing the monomial weights over certain subsets $\mathfrak{B}_v$ in the crystal $\mathfrak{B}_\lambda$, called Demazure crystals. Demazure crystals $\mathfrak{B}_v$ can be partitioned into Demazure atom crystals, $\widehat{\mathfrak{B}}_u$, where $u \in W\lambda$ runs in the Bruhat interval $\lambda \leq u \leq v$.

In type A_{n-1}, Lascoux and Schützenberger characterized key tableaux tableaux as semistandard Young tableaux (SSYT) with nested columns [9], and have used the *jeu de taquin* to define the right key map which sends a SSYT to a key tableau, called the right key of that SSYT. In each Demazure atom crystal there exists exactly one key tableau and the right key detects the Demazure atom crystal that contains a given SSYT [9, Theorem 3.8].

Received May 10, 2021.

By direct inspection of a Young tableau, Willis [17] has given an alternative algorithm to compute the right key tableau that does not require the use of *jeu de taquin*. Other methods to compute the type A right key map includes the alcove path model [11], semi skyline augmented fillings [13], and coloured vertex models. For a complete overview in type A, see [3] and the references therein. In type C_n, the symplectic key tableaux are characterized in [2, 14, 15, 6]. Using the Sheats symplectic *jeu de taquin*, a right key map is given, in [14, 15, 6], to send a Kashiwara-Nakashima (KN) tableau T to its right key tableau $K_+(T)$, that detects the Demazure atom crystal which contains T. They are also computed in the type C_n alcove path model [11], as well as for reverse King tableau using the colored five vertex model [4].

Jacon and Lecouvey [6] have suggested that Willis' method should be adaptable to type C_n. Motivated by Willis' direct inspection [17], we create an alternative algorithm, based on the direct inspection of a KN tableau, for the symplectic right key map, that does not use the symplectic *jeu de taquin*. Due to the added technicality of the symplectic *jeu de taquin* compared to the one for SSYT, Willis' *earliest weakly increasing subsequence* will fail to predict what gets slid during the Sheats symplectic *jeu de taquin*. Instead we need a way to calculate, without the use of *jeu de taquin*, what would appear in each column if we were to swap its length with the previous column length via *jeu de taquin*. The role of Willis' sequences will be replaced by our matchings (see Section 4). In type A, these kind of matches were used earlier [1].

The paper is organized as follows. In Section 2, we discuss the type C Kashiwara-Nakashima tableaux and the symplectic *jeu de taquin*. Section 3 briefly recalls the symplectic key tableaux and right key map via symplectic *jeu de taquin* [14]. In Section 4, we give an algorithm for computing the symplectic right key map that does not require the *jeu de taquin*, and prove that it returns the same object as the previous method.

This is an extended abstract of a full paper to appear.

2. Type C Kashiwara-Nakashima tableaux and *jeu de taquin*

We recall the symplectic tableaux introduced by Kashiwara and Nakashima to label the vertices of the type C_n crystal graphs [7]. Fix $n \in \mathbb{N}_{>0}$. 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

Mathematics of the University of Coimbra–UIDB/00324/2020, funded by the Portuguese Government through FCT/MCTES. It was also supported by FCT, through the grant PD/BD/142954/2018, under POCH funds, co-financed by the European Social Fund and Portuguese National Funds from MCTES.

References

[1] O. Azenhas. "Schur functions, pairing of parentheses, jeu de taquin and invariant factors". In: *Mathematical papers in honour of Eduardo Marques de Sá* Textos Mat. Sér. B (2006), pp. 7–24.

[2] O. Azenhas and R. Mamede. "Key polynomials of type C". In: *Slides from the presention in Séminaire Lotharingien de Combinatoire 69, Strobl.* 2012.

[3] B. Brubaker et al. "Colored five-vertex models and Demazure atoms". In: *Journal of Combinatorial Theory, Series A* 178 (2021), p. 105354.

[4] Valentin Buciumas and Travis Scrimshaw. "Quasi-solvable lattice models for $\mathbf{Sp_{2n}}$ and $\mathbf{SO_{2n+1}}$ Demazure atoms and characters". In: *arXiv preprint arXiv:2101.08907* (2021).

[5] C. De Concini. "Symplectic standard tableaux". In: *Advances in Mathematics* 34.1 (1979), pp. 1–27.

[6] Nicolas Jacon and Cédric Lecouvey. "Keys and Demazure crystals for Kac–Moody algebras". In: *Journal of Combinatorial Algebra* 4.4 (2020), pp. 325–358.

[7] M. Kashiwara and T. Nakashima. "Crystal Graphs for Representations of the q-Analogue of Classical Lie Algebras". In: *Journal of Algebra* 165 (1994), pp. 295–345.

[8] M. Kashiwara et al. *Crystal base and Littelmann's refined Demazure character formula*. Kyoto University. Research Institute for Mathematical Sciences [RIMS], 1992.

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

[10] C. Lecouvey. "Schensted-type correspondence, plactic monoid, and jeu de taquin for type Cn". In: *Journal of Algebra* 247.2 (2002), pp. 295–331.

[11] C. Lenart and A. Lubovsky. "A generalization of the alcove model and its applications". In: *Journal of Algebraic Combinatorics* 41.3 (2015), pp. 751–783.

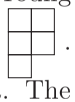
Now let's look to $\boldsymbol{rC_1}$. Before the slide, call $\overline{\boldsymbol{x'}}$ to the biggest unmatched number of $\boldsymbol{rC_1}$ smaller or equal then $\overline{\boldsymbol{x}}$. If no such $\overline{\boldsymbol{x'}}$ exists, then everything in $\boldsymbol{rC_1}$ between $\overline{\boldsymbol{\alpha}}$ and $\overline{\boldsymbol{x}}$ is matched, hence swapping $\overline{\boldsymbol{x}}$ by $\overline{\boldsymbol{\alpha}}$ will keep all of them matched, meaning that the algorithm works in this scenario. Let $\boldsymbol{x' < q_1 < q_2 < \cdots < q_m < \alpha}$ be the numbers between $\boldsymbol{x'}$ and $\boldsymbol{\alpha}$ that does not exist in $\boldsymbol{rC_1}$, right before the horizontal slide. Then, their symmetric exist in $\boldsymbol{rC_1}$. For all numbers in $\boldsymbol{rC_1}$ between $\boldsymbol{x'}$ and $\boldsymbol{\alpha}$, exist, in the same row in $\boldsymbol{\ell C_2}$, a number between $\boldsymbol{x'}$ and $\boldsymbol{\alpha}$, because $\boldsymbol{\alpha}$ is unmatched. Let $\boldsymbol{x' < q'_1 < q'_2 < \cdots < q'_m < \alpha}$ be the missing numbers between $\boldsymbol{x'}$ and $\boldsymbol{\alpha}$ in $\boldsymbol{\ell C_2}$, then $\boldsymbol{q_i \geq q'_i}$. Note that $\overline{\boldsymbol{q_1}} > \overline{\boldsymbol{q_2}} > \cdots > \overline{\boldsymbol{q_m}} > \overline{\boldsymbol{\alpha}}$ exist in $\boldsymbol{rC_1}$ after the horizontal slide and the numbers between $\overline{\boldsymbol{x'}}$ and $\overline{\boldsymbol{\alpha}}$ that can exist in $\boldsymbol{\ell C_2}$ are $\overline{\boldsymbol{q'_1}} > \overline{\boldsymbol{q'_2}} > \cdots > \overline{\boldsymbol{q'_m}}$, and since $\overline{\boldsymbol{q_i}} \leq \overline{\boldsymbol{q'_i}}$, these numbers are matching a number bigger or equal then $\boldsymbol{q_i}$ in $\boldsymbol{rC_1}$, meaning that $\boldsymbol{\alpha}$ is unmatched in $\boldsymbol{rC_1}$. Ignoring signs, the numbers that appear in either $\boldsymbol{rC_1}$ or $\boldsymbol{\ell C_2}$ are the same. So before playing the SJDT, applying the direct way algorithm we have that the unmatched numbers in $\boldsymbol{rC_1}$ are sent to the not used numbers of $\overline{\boldsymbol{q'_1}} > \overline{\boldsymbol{q'_2}} > \cdots > \overline{\boldsymbol{q'_m}}$ in $\boldsymbol{\ell C_2}$ (this is a bijection), and $\overline{\boldsymbol{x'}}$ is sent to the smallest available number, bigger or equal than $\overline{\boldsymbol{x'}}$. Now consider $\boldsymbol{rC_1}$ and $\boldsymbol{\ell C_2}$ after the slide. In $\boldsymbol{rC_1}$ we replace $\boldsymbol{x'}$ by $\overline{\boldsymbol{\alpha}}$ and remove $\boldsymbol{\alpha}$ and in $\boldsymbol{\ell C_2}$ there is $\boldsymbol{\alpha}$ or $\overline{\boldsymbol{\alpha}}$. In the direct algorithm, all unmatched numbers of $\overline{\boldsymbol{q_1}} > \overline{\boldsymbol{q_2}} > \cdots > \overline{\boldsymbol{q_m}} > \overline{\boldsymbol{\alpha}}$ are sent to the not used numbers of $\overline{\boldsymbol{q'_1}} > \overline{\boldsymbol{q'_2}} > \cdots > \overline{\boldsymbol{q'_m}}$ in $\boldsymbol{\ell C_2}$, but now we have more numbers in the first set than in the second, meaning that $\overline{\boldsymbol{\alpha}}$ will bump the image of the least unmatched number, which will bump the image of the second least 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 $\boldsymbol{x'}$ 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. ∎

Remark 4.5. It is possible to modify this algorithm in order to compute left keys.

5. Acknowledgements

I am grateful to O. Azenhas, my Ph.D. advisor, for her help on the preparation of this paper. This work was partially supported by the Centre for

the following order of its elements: $1 < \cdots < n < \overline{n} < \cdots < \overline{1}$ instead of the usual order. A vector $\lambda = (\lambda_1, \ldots, \lambda_n) \in \mathbb{Z}^n$ 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  . 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*, for short 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 SSYT of shape μ. Denote by $\mathrm{SSYT}(\mu/\nu, A)$ the set of all skew SSYT T of shape μ/ν, with entries in A. When $|v| = 0$ we write $\mathrm{SSYT}(\mu, A)$ and when $A = [n]$ we write $\mathrm{SSYT}(\mu/\nu, n)$. 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 heigth of a column ifs 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 (1CC)**.** Let C be a column. The $1CC$ holds for C if for all pairs i and $\overline{i}$ in C, where i is in the a-th row counting from the top of the column, and $\overline{i}$ in the b-th row counting from the bottom, we have $a + b \leq i$.

If a column C satisfies the $1CC$ then C has at most n letters. If $1CC$ doesn't hold for C we say that C *breaks the* $1CC$ *at* z, where z is the minimal such that z and $\overline{z}$ exist in C and there are more than z numbers in C with absolute value less or equal than z. For instance, the column $\begin{array}{|c|}\hline 1 \\ \hline 2 \\ \hline \overline{1} \\ \hline \end{array}$ breaks the

$1CC$ at 1, and $\begin{array}{|c|}\hline 2 \\ \hline 3 \\ \hline \overline{3} \\ \hline \end{array}$ is an admissible column. The following definition states conditions to when C can be *split*:

Definition 2.2. Let C be a column and let $I = \{z_1 > \cdots > z_r\}$ be the set of unbarred letters z such that the pair $(z, \overline{z})$ 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. $\boldsymbol{t_1}$ is the greatest letter of $\boldsymbol{[n]}$ satisfying $\boldsymbol{t_1 < z_1}$, $\boldsymbol{t_1 \notin C}$, and $\boldsymbol{\overline{t_1} \notin C}$,
2. for $\boldsymbol{i = 2, \ldots, r}$, we have that $\boldsymbol{t_i}$ is the greatest letter of $\boldsymbol{[n]}$ satisfying $\boldsymbol{t_i < \min(t_{i-1}, z_i)}$, $\boldsymbol{t_i \notin C}$, and $\boldsymbol{\overline{t_i} \notin C}$.

The column C is admissible if and only if C can be split [16, Lemma 3.1]. If C can be split then we define *right column* of C, $\mathrm{r}C$, and the *left column* of C, ℓC, as follows:

1. $\boldsymbol{rC}$ is obtained by changing in $\boldsymbol{C}$, $\boldsymbol{\overline{z_i}}$ into $\boldsymbol{\overline{t_i}}$ for each $\boldsymbol{z_i \in I}$ and reordering,
2. $\boldsymbol{\ell C}$ is obtained by changing in $\boldsymbol{C}$, $\boldsymbol{z_i}$ into $\boldsymbol{t_i}$ for each $\boldsymbol{z_i \in I}$ and reordering.

If $\boldsymbol{C}$ is admissible then $\boldsymbol{\ell C \leq C \leq rC}$ by entrywise comparison, where $\boldsymbol{\ell C}$ has the same barred part as $\boldsymbol{C}$ and $\boldsymbol{rC}$ the same unbarred part. If $\boldsymbol{C}$ doesn't have symmetric entries, then $\boldsymbol{C}$ is admissible and $\boldsymbol{\ell C = C = rC}$. In the next definition we give conditions for a column $\boldsymbol{C}$ to be *coadmissible*.

Definition 2.3. We say that a column $\boldsymbol{C}$ is coadmissible if for every pair $\boldsymbol{i}$ and $\boldsymbol{\overline{i}}$ on $\boldsymbol{C}$, where $\boldsymbol{i}$ is on the $\boldsymbol{a}$-th row counting from the top of the column, and $\boldsymbol{\overline{i}}$ on the $\boldsymbol{b}$-th row counting from the top, then $\boldsymbol{b - a \leq n - i}$.

Given an admissible column $\boldsymbol{C}$, consider the map $\boldsymbol{\Phi}$ that sends $\boldsymbol{C}$ to the column $\boldsymbol{\Phi(C)}$ of the same size in which the unbarred entries are taken from $\boldsymbol{\ell C}$ and the barred entries are taken from $\boldsymbol{rC}$. The *column* $\boldsymbol{\Phi(C)}$ is coadmissible and the algorithm to form $\boldsymbol{\Phi(C)}$ from $\boldsymbol{C}$ is reversible [10, Section 2.2]. In particular, every column without symmetric entries is simultaneously admissible and coadmissible.

its symmetric exist in $\boldsymbol{rC_2}$. Finally, if $\boldsymbol{\alpha}$ existed in $\boldsymbol{C_2}$, then $\boldsymbol{\overline{\alpha}}$ also existed in $\boldsymbol{C_2}$. That means that whatever number got sent to $\boldsymbol{\alpha}$ in the construction of $\boldsymbol{C_2}$, from $\boldsymbol{\Phi(C_2)}$, will be sent to the next available number, meaning that in $\boldsymbol{rC_2}$ will appear a new number, the smallest number bigger than $\boldsymbol{\alpha}$ such that neither it nor its symmetric exist in $\boldsymbol{rC_2}$. ∎

Proof of Theorem 4.1: Each SJDT in $\boldsymbol{T}$, 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 $\boldsymbol{C_2'}$ does not change. The cells on $\boldsymbol{\ell C_2}$ without cells to its left do not get to be matched. When we slide horizontally, the columns $\boldsymbol{rC_1}$ and $\boldsymbol{\ell C_2}$ may change more than the adding/removal of $\boldsymbol{\alpha}$, 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 $\boldsymbol{\alpha}$ are matched to the entry in the same row in $\boldsymbol{\ell C_2}$.

If $\boldsymbol{\alpha}$ is barred then, the remaining barred entries of $\boldsymbol{rC_1}$ and $\boldsymbol{\ell C_2}$ remain unchanged, and since all entries above $\boldsymbol{\alpha}$, including the unbarred ones, are matched to the entry directly on their right, there is no noteworthy change and everything runs as expected.

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

Let's look to $\boldsymbol{\ell C_2}$. Let $\boldsymbol{\alpha < p_1 < p_2 < \cdots < p_m = y}$ be the numbers between $\boldsymbol{\alpha}$ and $\boldsymbol{y}$ that does not exist in $\boldsymbol{\ell C_2}$, right before the horizontal slide. Then, their symmetric exist in $\boldsymbol{\ell C_2}$. For all numbers in $\boldsymbol{rC_2}$ between $\boldsymbol{\alpha}$ and $\boldsymbol{y}$, exist, in the same row in $\boldsymbol{rC_1}$, a number between $\boldsymbol{\alpha}$ and $\boldsymbol{y}$. Let $\boldsymbol{\alpha < p_1' < p_2' < \cdots < p_m' = y}$ be the missing numbers between $\boldsymbol{\alpha}$ and $\boldsymbol{y}$ in $\boldsymbol{rC_1}$, then $\boldsymbol{p_i \leq p_i'}$. Note that $\boldsymbol{\overline{p_1} > \overline{p_2} > \cdots > \overline{p_m} = \overline{y}}$ exist in $\boldsymbol{\ell C_2}$ after the horizontal slide and that the biggest numbers between $\overline{\boldsymbol{\alpha}}$ and $\overline{\boldsymbol{y}}$ (not including $\overline{\boldsymbol{\alpha}}$) that can exist in $\boldsymbol{rC_1}$ are $\boldsymbol{\overline{p_1'} > \overline{p_2'} > \cdots > \overline{p_m'}}$, and since $\boldsymbol{\overline{p_i} \geq \overline{p_i'}}$, the matching holds for this interval after swapping $\overline{\boldsymbol{\alpha}}$ by $\overline{\boldsymbol{y}}$ in $\boldsymbol{\ell C_2}$.

• *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 4.4. Let $T = \begin{array}{|c|c|}\hline 2&3\\\hline 3&4\\\hline 5&\overline{5}\\\hline \overline{5}\\\hline \overline{2}\\\cline{1-1}\end{array}$. 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

Example 2.4. Let $n = 4$ and $C = \begin{array}{|c|}\hline 2\\\hline 4\\\hline \overline{2}\\\cline{1-1}\end{array}$ be an admissible column. Then $\ell C = \begin{array}{|c|}\hline 1\\\hline 4\\\hline \overline{2}\\\cline{1-1}\end{array}$ and $rC = \begin{array}{|c|}\hline 2\\\hline 4\\\hline \overline{1}\\\cline{1-1}\end{array}$. So $\Phi(C) = \begin{array}{|c|}\hline 1\\\hline 4\\\hline \overline{1}\\\cline{1-1}\end{array}$ is coadmissible. C is also coadmissible and $\Phi^{-1}(C) = \begin{array}{|c|}\hline 3\\\hline 4\\\hline \overline{3}\\\cline{1-1}\end{array}$.

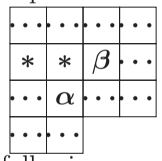
Let T be a skew tableau with all of its columns admissible. The *split form* of a skew tableau T, $spl(T)$, is the skew tableau obtained after replacing each column C of T by the two columns $\ell C\, rC$. The tableau $spl(T)$ has double the amount of columns of T.

A skew SSYT T is a *Kashiwara-Nakashima (KN) skew tableau* if its split form is a skew SSYT. 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 $\overline{i}$.

Example 2.5. The split of the tableau $T = \begin{array}{|c|c|}\hline 2&2\\\hline 3&3\\\hline \overline{3}\\\cline{1-1}\end{array}$ is the tableau $spl(T) = \begin{array}{|c|c|c|c|}\hline 1&2&2&2\\\hline 2&3&3&3\\\hline \overline{3}&\overline{1}\\\cline{1-2}\end{array}$. Hence $T \in \mathcal{KN}((2,2,1),3)$ and weight $\mathrm{wt}T = (0,2,1)$.

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 $\overline{i}$ 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 $SSYT(\mu/\nu, n) \subseteq \mathcal{KN}(\mu/\nu, n)$.

2.1. Sheats symplectic *jeu de taquin*. Sheats symplectic *jeu de taquin* (SJDT) [10, 16] is a procedure on KN skew tableaux, compatible with *Knuth equivalence* (or plactic equivalence on words over the alphabet $[\pm n]$) [10], 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 C_1C_2 KN skew tableaux. 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 $\boldsymbol{T}$ be a punctured skew tableau with two columns $\boldsymbol{C_1}$ and $\boldsymbol{C_2}$ with the puncture in $\boldsymbol{C_1}$. In that case, the puncture splits into two punctures in $\boldsymbol{spl(T)}$, and ignoring the punctures, $\boldsymbol{spl(T)}$ must be semistandard. Let $\boldsymbol{\alpha}$ be the entry under the puncture of $\boldsymbol{rC_1}$, and $\boldsymbol{\beta}$ the entry to the right of the puncture of $\boldsymbol{rC_1}$ ($\boldsymbol{\alpha}$ or $\boldsymbol{\beta}$ may not exist): $\boldsymbol{spl(T) = \ell C_1 rC_1 \ell C_2 rC_2 =}$  .

The elementary steps of SJDT are the following:

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

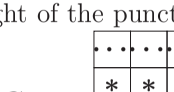
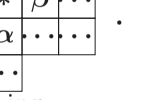
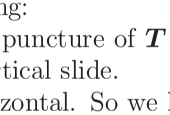
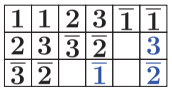
B. If the slide is not vertical, then it is horizontal. So we have $\boldsymbol{\alpha > \beta}$ or $\boldsymbol{\alpha}$ does not exist. Let $\boldsymbol{C_1'}$ and $\boldsymbol{C_2'}$ be the columns obtained after the slide. We have two subcases, depending on the sign of $\boldsymbol{\beta}$:

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

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

Applying successively elementary SJDT steps, eventually, the puncture will be a cell such that $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ do not exist. Then, we redefine the shape to not include this cell and SJDT ends. Given an admissible tableau $\boldsymbol{T}$ of shape $\boldsymbol{\mu/\nu}$, a box of the diagram of shape $\boldsymbol{\nu}$ such that boxes under it and to the right are not in that shape is called an *inner corner* of $\boldsymbol{\mu/\nu}$. An outside corner is a box of $\boldsymbol{\mu}$ such that boxes under it and to the right are not in the shape $\boldsymbol{\mu}$. The rectification of $\boldsymbol{T}$ consists in playing the SJDT until we get a

obtain a column $\boldsymbol{C_k'}$, consisting of $\boldsymbol{rC_k}$ together with extra cells, with the same length as $\boldsymbol{rC_1}$. We claim that $\boldsymbol{C_k' = K_+^1(T)}$.

Example 4.2. Let $\boldsymbol{T = C_1C_2C_3 =}$, with split form $\boldsymbol{spl(T) =}$  . We match $\boldsymbol{rC_1}$ and $\boldsymbol{\ell C_2}$, as indicated by the letters $\boldsymbol{a}$ and $\boldsymbol{b}$:  . Hence $\overline{\mathbf{2}}$ creates a $\overline{\mathbf{1}}$ in $\boldsymbol{rC_2}$, completing the right column $\boldsymbol{rC_2}$:  . Now we match $\boldsymbol{C_2'}$ and $\boldsymbol{\ell C_3}$, which is already done, and see what new cells $\mathbf{3}$ and $\overline{\mathbf{2}}$ create in $\boldsymbol{rC_3}$, obtaining  . Hence $\boldsymbol{K_+^1(T) =}$ is obtained from $\boldsymbol{C_3'}$ after reordering its entries.

4.1. The proof of Theorem 4.1. It is enough to prove that by the end of this algorithm, the entries in $\boldsymbol{C_k'}$ are the entries on the right column of the rightmost column of $\boldsymbol{T_k'}$ from Algorithm 3.3. In fact, it is enough to do this for $\boldsymbol{k = 2}$. For bigger $\boldsymbol{k}$ note that the entries that are "slid" into $\boldsymbol{C_k}$ come from $\boldsymbol{rC_{k-1}}$, 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 $\boldsymbol{rC_2}$ given that we know $\boldsymbol{\alpha}$, the entry that is horizontally slid:

Lemma 4.3. *Suppose that $\boldsymbol{T = C_1C_2}$ 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 $\boldsymbol{\alpha}$, on the left of the puncture, is an unmatched cell of $\boldsymbol{rC_1}$.*

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 with the rightmost column of T'_{i-1} and the i-th column of T, and go back to **2.**. Else, return the right column of the rightmost column of T'_i.

This algorithm is exemplified on the bottom path of (1).

Corollary 3.4. *If T is a rectangular tableau, $K_+(T) = rC_k rC_k \dots rC_k$ (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 [17]. It is a simplified version of the algorithm presented here.

4. 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 4.1 (The direct way algorithm for the right key)**.** *Let T be a straight KN tableau with columns $C_1, C_2, \dots, C_k$, and consider its split form $spl(T)$. For every right column $rC_2, \dots, rC_k$, 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

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 [10, Corollary 6.3.9].

Consider the KN skew tableau $T = \begin{array}{|c|}\hline 2 \\ \hline 1 \\ \hline 2 \\ \hline \end{array}\!\!\begin{array}{|c|}\hline 3 \\ \hline \overline{1} \\ \hline \end{array}$. To rectify T via SJDT, one creates a puncture in a inner corner and split, obtaining $\begin{array}{|c|c|c|c|}\hline * & * & 2 & 2 \\ \hline 1 & 1 & 3 & 3 \\ \hline 2 & 2 & \overline{1} & \overline{1} \\ \hline \end{array}$. So, the first two slides are vertical, obtaining $\begin{array}{|c|c|c|c|}\hline 1 & 1 & 2 & 2 \\ \hline 2 & 2 & 3 & 3 \\ \hline * & * & \overline{1} & \overline{1} \\ \hline \end{array}$. Finally, we do an horizontal slide, of type $B.1$, in which we take $\overline{1}$ from the second column of T and add it to the coadmissible column of the first column of T, obtaining the tableau $\begin{array}{|c|c|}\hline 2 & 2 \\ \hline 3 & 3 \\ \hline \overline{3} \\ \cline{1-1} \end{array}$. 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: $\begin{array}{|c|c|}\hline \multicolumn{2}{|c|}{} & * \\ \hline 1 & \overline{1} \\ \hline 2 \\ \cline{1-1} \end{array} \mapsto \begin{array}{|c|c|}\hline \multicolumn{2}{|c|}{} & \\ \hline 1 & \overline{1} \\ \hline 2 \\ \cline{1-1} \end{array}$; $\begin{array}{|c|c|}\hline \multicolumn{2}{|c|}{} \\ \hline 1 & \overline{1} \\ \hline 2 & * \\ \hline \end{array} \mapsto \begin{array}{|c|c|}\hline \multicolumn{2}{|c|}{} \\ \hline \multicolumn{2}{|c|}{} \\ \hline 2 & 2 \\ \hline \end{array}$.

If columns C_1 and C_2 do not have negative entries then the SJDT applied to C_1C_2 coincides with the *jeu de taquin* known for SSYT. Next section, 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. Hence, SJDT will not incur in a loss/gain of boxes, that could happen in the elementary step $B.2$.

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

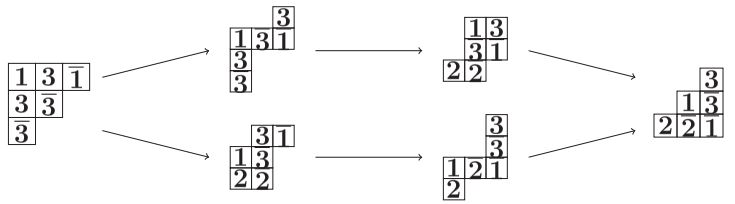
A *key tableau* of shape $\boldsymbol{\lambda}$, in type $\boldsymbol{C_n}$, is a KN tableau in $\boldsymbol{\mathcal{KN}(\lambda, n)}$, in which the set of elements of each column, right to left, contains the set of elements of the previous column, if any, and the letters $\boldsymbol{i}$ and $\boldsymbol{\bar{i}}$ do not appear simultaneously as entries, for any $\boldsymbol{i \in [n]}$. The split form of a KN key tableau consists of the duplication of each column.

The set $\boldsymbol{\mathcal{KN}(\lambda, n)}$ is endowed with a symplectic crystal structure, denoted $\boldsymbol{\mathfrak{B}_\lambda}$ [7, 10]. The key tableaux in $\boldsymbol{\mathcal{KN}(\lambda, n)}$ are the unique tableaux in $\boldsymbol{\mathcal{KN}(\lambda, n)}$ whose weight is $\boldsymbol{\alpha\lambda}$, for all elements $\boldsymbol{\alpha}$ in the Weyl group $\boldsymbol{W}$, denoted $\boldsymbol{key(\alpha\lambda)}$. The orbit of $\boldsymbol{key(\lambda)}$, the highest weight element of $\boldsymbol{\mathfrak{B}_\lambda}$, under the action of the Weyl group $\boldsymbol{W}$, is defined to be $\boldsymbol{O(\lambda) = \{}key\boldsymbol{(\alpha\lambda) : \alpha \in W\}}$. The *right key map* $\boldsymbol{K_+}$ sends each $\boldsymbol{T \in \mathcal{KN}(\lambda, n)}$ to one element $\boldsymbol{K_+(T)}$ in the orbit $\boldsymbol{O(\lambda)}$, called the *right key tableau* of $\boldsymbol{T}$. The right key of a KN tableau $\boldsymbol{T}$ is a key tableau of the same shape as $\boldsymbol{T}$, entrywise "slightly" bigger than $\boldsymbol{T}$. In [14, 15] such tableau is computed using the aforementioned SJDT.

Lemma 3.1. *[14, 15] Given $\boldsymbol{T \in \mathcal{KN}(\lambda, n)}$ and a skew shape whose column lengths are a permutation of the column lengths of $\boldsymbol{T}$, then there is exactly one skew tableau with that shape that rectifies to $\boldsymbol{T}$. Futhermore, the last column only depends on its length.*

Definition 3.2 (Right key map)**.** [14, 15] Given $\boldsymbol{T \in \mathcal{KN}(\lambda, n)}$, we consider the KN skew tableaux with the same number of columns of each length as $\boldsymbol{T}$, each one corresponding to a permutation of its column lengths. Then we replace each column of $\boldsymbol{T}$ with a column of the same size taken from the right columns of the last columns of all those skew tableaux associated to $\boldsymbol{T}$. This tableau is the right key tableau of $\boldsymbol{T}$, $\boldsymbol{K_+(T)}$. This map restricted to $\boldsymbol{SSYT(\lambda, n)}$ recovers the Lascoux-Schützenberger right key map [9].

Given $\boldsymbol{T \in \mathcal{KN}(\lambda, n)}$ we apply the SJDT on consecutive columns to compute all skew tableaux in the conditions the previous definition. For instance, the tableau $\boldsymbol{T} = \begin{array}{|c|c|c|}\hline 1 & 3 & \bar{1} \\ \hline 3 & \bar{3} \\ \cline{1-2} \bar{3} \\ \cline{1-1}\end{array}$ gives rise to six KN skew tableaux with the same number of columns of each length as $\boldsymbol{T}$, each one corresponding to a permutation of its column lengths.



(1)

The right key tableau of $\boldsymbol{T}$ has columns $\boldsymbol{r}\begin{array}{|c|}\hline 3 \\ \hline \bar{3} \\ \hline \bar{1} \\ \hline\end{array}$, $\boldsymbol{r}\begin{array}{|c|}\hline 3 \\ \hline \bar{1} \\ \hline\end{array}$ and $\boldsymbol{r}\begin{array}{|c|}\hline \bar{1} \\ \hline\end{array}$. Hence $\boldsymbol{K_+(T)} = \begin{array}{|c|c|c|}\hline 3 & 3 & \bar{1} \\ \hline \bar{2} & \bar{1} \\ \cline{1-2} \bar{1} \\ \cline{1-1}\end{array}$. Lemma 3.1 shows that the column commutation action defined by the SJDT on two consecutive columns of a straight KN tableau $\boldsymbol{T}$ of shape $\boldsymbol{\lambda}$ gives rise to a permutohedron where the vertices are all the KN skew tableaux in the Knuth class of $\boldsymbol{T}$ whose column length sequence is a permutation of the column length sequence of $\boldsymbol{T}$ [9]. For instance, (1) is a permutohedron (hexagon) for $\boldsymbol{\mathfrak{S}_3}$.

Let $\boldsymbol{T = C_1C_2\cdots C_k}$ be a straight KN tableau with columns $\boldsymbol{C_1, C_2, \ldots, C_k}$. Note that, to compute which entries appear in the $\boldsymbol{i}$-th column of $\boldsymbol{K_+(T)}$ we do not need to look to the first $\boldsymbol{i-1}$ columns of $\boldsymbol{T}$. We only need the last column of a skew tableau obtained by applying the SJDT to the columns $\boldsymbol{C_i \ldots C_k}$ of $\boldsymbol{T}$, so that the last column has the length of $\boldsymbol{C_i}$, because by Lemma 3.1 all last columns of skew tableaux associated to $\boldsymbol{T}$ with the same length are equal. Let $\boldsymbol{K^1_+(T)}$ be the map that given a tableau returns the first column of $\boldsymbol{K_+(T)}$. This is noticeable in 1 where $\boldsymbol{K_+(T) = K^1_+(C1C2C3)K^1_+(C2C3)K^1_+(C3)}$. In general, $\boldsymbol{K_+(T) = K^1_+(C_1\cdots C_k)K^1_+(C_2\cdots C_k)\cdots K^1_+(C_k)}$. Based on this observation and Lemma 3.1, next algorithm refines Definition 3.2 to compute $\boldsymbol{K^+_1(T)}$ using SJDT:

Algorithm 3.3. Let $\boldsymbol{T}$ be a straight KN tableau:

1. Let $\boldsymbol{i = 2}$.

2. If $\boldsymbol{T}$ has exactly one column, return the right column of $\boldsymbol{T}$. Otherwise, let $\boldsymbol{T_i := T_2}$ be the tableau formed by the first two columns of $\boldsymbol{T}$.