An insertion scheme for C_n crystals

T. H. Baker

Research Institute for Mathematical Sciences,
Kyoto University, Kyoto 606-8502, Japan

We define bumping and sliding algorithms for the C_n tableaux of Kashiwara and Nakashima. Together, they are used to define an insertion scheme for such tableaux which turns out to be a crystal isomorphism.

1 Introduction

The concept of the crystal basis [8] of a representation of quantum (affine) Lie algebras has proved very useful in examining the structure of such representations[6]. The crystal bases for the finite-dimensional irreducible representations of the quantum group $U_q(\mathfrak{g})$ where $\mathfrak{g}$ is one of the classical Lie algebras A_n, B_n, C_n or D_n, were explicitly described by Kashiwara and Nakashima [9] in terms of tableaux. The tensor product rule for crystals allows the decomposition of tensor products of crystals in terms of irreducible crystals to be described in a combinatorial fashion [19]. The action of the Kashiwara operators $\tilde{e}_i$, $\tilde{f}_i$ on the tensor product of two crystals is given by [7]

$$\tilde{f}_i(b_1 \otimes b_2) = \begin{cases} \tilde{f}_i(b_1) \otimes b_2 & \phi_i(b_1) > \epsilon_i(b_2) \\ b_1 \otimes \tilde{f}_i(b_2) & \phi_i(b_1) \leq \epsilon_i(b_2) \end{cases}$$

$$\tilde{e}_i(b_1 \otimes b_2) = \begin{cases} \tilde{e}_i(b_1) \otimes b_2 & \phi_i(b_1) \geq \epsilon_i(b_2) \\ b_1 \otimes \tilde{e}_i(b_2) & \phi_i(b_1) < \epsilon_i(b_2) \end{cases} \tag{1.1}$$

where $\phi_i(b) = \max\{k | (\tilde{f}_i)^k\, b \neq 0\}$, $\varepsilon_i(b) = \max\{k | (\tilde{e}_i)^k\, b \neq 0\}$.

In the case of type A tableaux (i.e. the usual semi-standard tableaux with weakly increasing rows and strictly increasing columns), there is an important connection between the crystal action and the Schensted insertion algorithm [24], which we now briefly recall. The *column* insertion procedure inserts a letter x into a tableau b as follows: let y be the smallest integer in the first column $\geq x$. Then x replaces y in the first column while y is then "bumped" into the next column where the process is repeated. This procedure stops when the entry which is bumped is larger than all the entries in the next column, in which case it is placed at the end of that column. The result of such a procedure is a tableau which is one box larger than b, call it $b \leftarrow x$ say (Note: this notation is usually reserved for row bumping (e.g. in [5]) but we need it for compatibility with the Japanese word reading). Using this insertion procedure, one can define the "product tableaux"

$$b * c := (\cdots((b \leftarrow c_1) \leftarrow c_2) \leftarrow \cdots) \leftarrow c_p) \tag{1.2}$$

where $w(c) = c_1\, c_2 \cdots c_p$ is the *Japanese* reading of c i.e. reading from top to bottom and leftwards down successive columns.

Now, suppose we have two irreducible A_{n-1} crystals $B(\mu)$, and $B(\nu)$. Then it is well-known that their tensor product has a finite decomposition

$$B(\mu) \otimes B(\nu) \cong \bigoplus_j B(\lambda_j)$$

which can be described exactly via the Littlewood–Richardson rule see [19] or [18, p. 143]. In fact, the above isomorphism is given explicitly by

Proposition 1.1 *Let* $\psi_A : B(\mu) \otimes B(\nu) \longrightarrow \bigoplus_j B(\lambda_j)$ *be defined by*

$$\psi_A(b_1 \otimes b_2) = b_1 * b_2$$

Then ψ_A *is the unique crystal isomorphism describing the above tensor product decomposition.*

The aim of the present article is to define an insertion scheme for the C_n tableaux of Kashiwara and Nakashima which will provide the analogue of the above Proposition for the case of C_n crystals. Apart from the tableaux of Kashiwara and Nakashima, analogues of semi-standard Young tableaux which describe irreducible finite-dimensional representations of the classical Lie algebras B_n, C_n and D_n have been introduced by King [10], De Concini [3], King and El–Sharkaway [11], Koike and Terada [14], Sundaram [27], King and Welsh [12] and Proctor [23]. For a good overview of the various bijections existing between them, consult the article of Fulmek and Krattenthaler [4]. For some of these types of tableaux, insertion schemes have been generated, such as the one of Berele [2] (for the C_n tableaux of King), Sundaram [27], Proctor [22] Benkart and Stroomer [1], and Okada [20, 21].

The plan of the paper is as follows: in Section 2 we introduce the necessary notations and basic results about crystals of irreducible finite-dimensional representations of $U_q(C_n)$, in particular their description in terms of tableaux and the action of the Kashiwara operators $\tilde{e}_i$, $\tilde{f}_i$ on such tableaux. In Section 3, the (column) bumping procedure is defined, along with the reverse bumping procedure. In Section 4, the sliding procedure is defined, along with the reverse sliding procedure. In Section 5, these two procedures are combined to define an insertion scheme for the C_n tableaux. In Section 6, the elementary Knuth transformations governing the insertion scheme are deduced. In particular, we show that the C_n sliding procedure preserves Knuth equivalence, and that the Knuth equivalence classes are stable under the action of the Kashiwara operators. In Section 7 we give the main result, that the C_n insertion scheme realizes an isomorphism of C_n crystals. Finally, in Section 8 we present proofs that the bumping and sliding introduced earlier is indeed well-defined.

2 C_n tableaux and crystal action

2.1 Tableaux

Let $\Lambda_i, i = 1, \ldots, n$ denote the fundamental weights of the root system of C_n. The representation $V(\Lambda)$ with $\Lambda = \Lambda_{m_1} + \Lambda_{m_2} + \cdots + \Lambda_{m_p}$, $m_1 \leq m_2 \leq \cdots \leq m_p$ can be represented by a Young diagram consisting of columns of length $m_1, m_2, \ldots, m_p$ (going from right to left). Such representations (respectively crystal bases) can be embedded into the tensor product of multiple copies of the fundamental irreducible C_n representation (resp. crystal base) with highest weight Λ_1.

The representation $V(\Lambda_1)$ has a basis $\{\boxed{i}, \boxed{\overline{i}}\ ; 1 \leq i \leq n\}$ and the action of the $U_q(C_n)$ generators is given as follows

$$\begin{array}{rclcll}
q^h\boxed{i} & = & q^{<h,\epsilon_i>}\boxed{i} & q^h\boxed{\overline{i}} = q^{-<h,\epsilon_i>}\boxed{\overline{i}} & & \\
e_j\boxed{i} & = & \delta_{i,j+1}\boxed{i-1} & e_j\boxed{\overline{i}} = \delta_{i,j}\boxed{\overline{i+1}} & 1 \leq j \leq n-1 & \\
f_j\boxed{i} & = & \delta_{i,j}\boxed{i+1} & f_j\boxed{\overline{i}} = \delta_{i,j+1}\boxed{\overline{i-1}} & 1 \leq j \leq n-1 & (2.1) \\
e_n\boxed{i} & = & 0 & e_n\boxed{\overline{i}} = \delta_{i,n}\boxed{n} & & \\
f_n\boxed{i} & = & \delta_{i,n}\boxed{\overline{n}} & f_n\boxed{\overline{i}} = 0 & &
\end{array}$$

As a result, the crystal graph of $B(\Lambda_1)$ has the following structure

$$\boxed{1} \xrightarrow{1} \boxed{2} \xrightarrow{2} \cdots \xrightarrow{n-1} \boxed{n} \xrightarrow{n} \boxed{\overline{n}} \xrightarrow{n-1} \boxed{\overline{n-1}} \xrightarrow{n-2} \cdots \xrightarrow{2} \boxed{\overline{2}} \xrightarrow{1} \boxed{\overline{1}}$$

The crystal structure on $B(\Lambda)$ for general $\Lambda = \Lambda_{m_1} + \cdots + \Lambda_{m_p}$ can be described by first embedding $B(\Lambda_{m_k})$ into $B(\Lambda_1)^{\otimes m_k}$ for each $k = 1, \ldots p$ and then embedding $B(\Lambda)$ into $B(\Lambda_{m_1}) \otimes \cdots \otimes B(\Lambda_{m_p})$. As a result, the crystal base $B(\Lambda)$ can be described by a set of tableaux of shape Λ on the alphabet $\mathcal{X} = \mathcal{A} \cup \overline{\mathcal{A}}$, $\mathcal{A} := \{1, 2, \ldots, n\}$ with the (total) order $1 < 2 < \cdots n-1 < n < \overline{n} < \overline{n-1} < \cdots \overline{2} < \overline{1}$. The entries of such tableaux obey certain conditions that we now describe.

For the alphabet $\mathcal{X}$, we follow the convention that greek letters (α, β etc) belong to $\mathcal{A} \cup \overline{\mathcal{A}}$ while latin letters x, y etc (resp. $\overline{x}$, $\overline{y}$ etc) belong to $\mathcal{A}$ (resp. $\overline{\mathcal{A}}$). Given a column C of length N containing the entries (reading from top to bottom) $\alpha_1, \alpha_2, \ldots, \alpha_N$, define the following function on C

$$\mathrm{pos}_C(\alpha_k) = \begin{cases} k & \text{if } \alpha_k \in \mathcal{A} \\ N+1-k & \text{if } \alpha_k \in \overline{\mathcal{A}} \end{cases}$$

In other word, $\mathrm{pos}_C(a)$ (resp. $\mathrm{pos}_C(\overline{a})$) is the position of the entry a (resp. $\overline{a}$) w.r.t. the top (resp. bottom) of the column C. Sometimes we shall drop the subscript C when the meaning is clear. We shall also extend the domain of $\mathrm{pos}(\cdot)$ to include empty boxes, in which case $\mathrm{pos}(\square)$ will be the distance of the empty box from the top or the bottom of the column, the choice being clear from the context. Say a column C satisfies the *one column condition* if for all pairs $a, \overline{a}$ in C

$$\mathrm{pos}_C(a) + \mathrm{pos}_C(\overline{a}) \leq a \tag{2.2}$$

Given two adjacent columns C, C' consisting of the entries (reading from top to bottom) $\alpha_1, \ldots \alpha_N$ and $\beta_1, \ldots, \beta_M$ respectively ($M \leq N$), say there is an (a,b) configuration if there exists $i \leq j < k \leq l$ such that either $(\alpha_i, \alpha_j, \alpha_k, \beta_l) = (a, b, \overline{b}, \overline{a})$ or $(\alpha_i, \beta_j, \beta_k, \beta_l) = (a, b, \overline{b}, \overline{a})$ (note we include the possibility that $a = b$). Sometimes we shall distinguish the two cases and call the former a *left* (a,b) configuration and the latter a *right* (a,b) configuration. Given an (a,b) configuration, define a function $p(a, b|\overline{b}, \overline{a}) := (j - i) + (l - k)$. Say a pair of adjacent columns satisfies the *two column condition* if for every (a,b) configuration

$$p(a, b|\overline{b}, \overline{a}) < b - a \tag{2.3}$$

More generally, suppose there exists integers $i \leq j < k \leq l$ such that $(\alpha_i, \alpha_j, \alpha_k, \beta_l) = (a, b, \overline{c}, \overline{d})$ or $(\alpha_i, \beta_j, \beta_k, \beta_l) = (a, b, \overline{c}, \overline{d})$. We call such a configuration a (left/right) $((a, b|\overline{c}, \overline{d})$- configuration.

The crystal base $B(\Lambda)$ then consists of the set of tableaux T on the shape Λ with entries in $\mathcal{X}$ satisfying the following conditions

1. the entries of T increase weakly along the rows.
2. the entries of T increase strictly down the columns.
3. for each column C, the one column condition (1CC) holds.
4. for each pair of adjacent columns C, C', the two column condition (2CC) holds.

Note that (2.3) implies that a C_n tableau T has no (a,a) configurations for any a.

The following lemmas given by Kashiwara and Nakashima [9, Lemmas 4.3.1, 4.4.2], will be essential tools in the proofs of our results

Lemma 2.1 *Suppose a and $\overline{b}$ belong to a column C in a C_n tableau T. Then*

$$\mathrm{pos}_C(a) + \mathrm{pos}_C(\overline{b}) \leq \max(a, b)$$

Lemma 2.2 *For any $(a,b|\overline{c},\overline{d})$-configuration, define $p(a,b|\overline{c},\overline{d}) := (j-i)+(l-k)$. Then we have*

$$p(a,b|\overline{c},\overline{d}) < \max(b,c) - \min(a,d)$$

Remark. During the proofs of our results, we will also need to consider situations where there is a tableau containing an empty box. A careful analysis of the proofs of Lemmas 2.1 and 2.2 show that the upper bounds given there hold when the empty box lies between a and $\overline{b}$ (for Lemma 2.1) or between b and $\overline{c}$ or above a or below $\overline{d}$ (for Lemma 2.2). Moreover, if the empty box lies above a or below $\overline{b}$ (for Lemma 2.1) or between a and b or between $\overline{c}$ and $\overline{d}$ (for Lemma 2.2), then the upper bound increases by 1.

2.2 Crystal action on $B(\Lambda)$

Given a tableau T, define the (Japanese) reading of T, $w(T)$ to be the word obtained by reading the columns of T from top to bottom and right to left (such a reading is necessary to be compatible with the tensor product rule (1.1)). For a fixed i, to compute the action of the Kashiwara operators $\tilde{e}_i$, $\tilde{f}_i$ on a tableau T, perform the following steps (if $i = n$ ignore all expressions containing $i+1$ in what follows)

1. Write down $w(T)$ and ignore all letters different from i, $i+1$, $\overline{i}$, $\overline{i+1}$.
2. Ignore adjacent letters $(i,i+1)$, $(i,\overline{i})$, $(\overline{i+1},i+1)$, $(\overline{i+1},\overline{i})$.
3. Repeat step 2 until the result does not change.

The resulting word will be of the form $u_-^s u_+^t$ where u_- (resp. u_+) is one of $\{i+1,\overline{i}\}$ (resp. $\{i,\overline{i+1}\}$). If $s=0$ (resp. $t=0$), then $\tilde{e}_i T = 0$ (resp. $\tilde{f}_i T = 0$). Otherwise $\tilde{e}_i$ changes the rightmost u_- to the corresponding u_+ i.e. $i \mapsto i+1$ and $\overline{i+1} \mapsto \overline{i}$ (resp. $\tilde{f}_i$ changes the leftmost u_+ to the corresponding u_-).

Example 2.3

T =

1	2	3	$\overline{5}$
2	4	4	$\overline{4}$
4	5	$\overline{5}$	
$\overline{4}$	$\overline{3}$	$\overline{2}$	

$w(T) = \overline{5}\,\overline{4}\,3\,4\,\overline{5}\,\overline{2}\,2\,4\,5\,\overline{3}\,1\,2\,4\,\overline{4}$ $\quad$ $\tilde{f}_4\,T$ =

1	2	3	$\overline{5}$
2	4	5	$\overline{4}$
4	5	$\overline{5}$	
$\overline{4}$	$\overline{3}$	$\overline{2}$	

3 Bumping

3.1 Forward Bumping

The (column) bumping process for type A_{n-1} tableaux is simply described as follows: to insert an element x into a column C, if x is strictly greater than all entries in C, place x at the bottom of the column C. Otherwise select the entry $y \in C$ which is the smallest entry $\geq x$. The entry y is replaced by x, and y is "bumped" into the next column to the right of C.

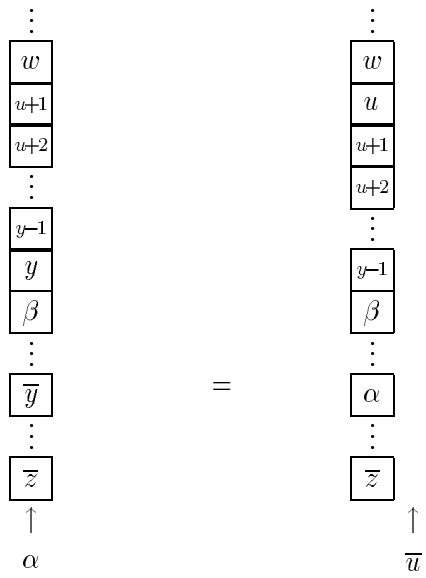
In the case of the C_n tableaux of Kashiwara and Nakashima, to insert an element α into a column C, proceed as in the type A case unless one of the following two situations arise

I. if $\overline{y} \in C$ is the smallest letter $\geq \alpha$ and $y \in C$ also.

II. if $\alpha = x$ and $\overline{x} \in C$.

It can be seen that such conditions are mutually exclusive. If such conditions arise, one must carry out a "special" bump which we now describe.

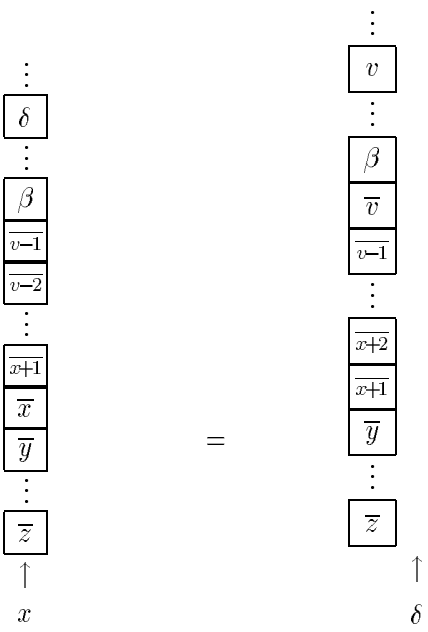
Type I special bump



In this situation, suppose there is a string (possibly empty) of consecutive integers $y-1, y-2, \ldots, u+2, u+1$ lying above the entry y in the column C, with the entry w immediately above $u+1$ being $\leq u-1$. Then the type I bump consists of replacing the entry $\overline{y}$ with α, deleting the entry y and shifting the entries $y-1, \ldots, u+1$ down one position, and inserting an entry u between w and $u+1$ in the column C. The entry $\overline{u}$ is then bumped into the next column to the right.

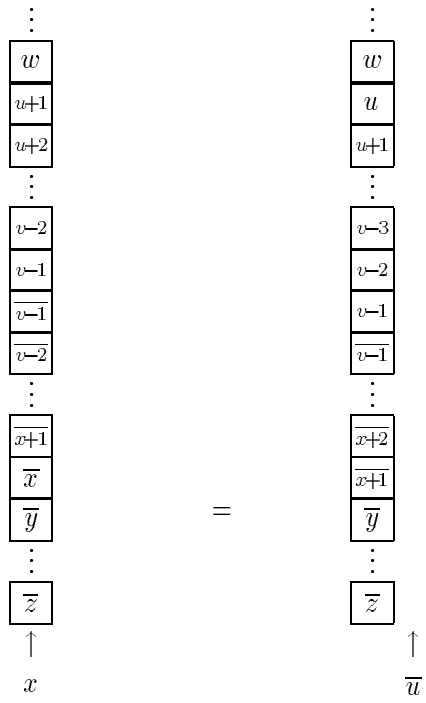
In the second case, suppose that there is a string (possibly empty) of consecutive entries $\overline{x+1}, \overline{x+2}, \ldots, \overline{v-2}, \overline{v-1}$ above the entry $\overline{x} \in C$. Let β be the next entry above $\overline{v-1}$. Then we have two subcases depending on whether (a) $v \leq \beta \leq \overline{v+1}$ or (b) $\beta \leq v-1$ or no such β exists.

Type IIa special bump



In this case, suppose δ is the smallest entry in C which is $\geq v$ where $\overline{v}$ is the first letter absent from the string of consecutive letters above $\overline{x}$. Then the type IIa bump consists of deleting the entry $\overline{x}$, shifting the entries $\overline{x+1}, \ldots, \overline{v-1}$ down one position, inserting entry $\overline{v}$ between $\overline{v-1}$ and β, and replacing δ with v. The entry δ is then bumped into the next column.

Type IIb special bump



In this case, suppose that there is a string of consecutive integers $v-1, v-2, \ldots, u+1$ above the entry $\overline{v-1}$ and let w be the entry immediately above $u+1$ which is $< u$. This only occurs when $\beta = v-1$ in case (b); if $\beta < v-1$ such a string is empty and $u := v-1$. The type IIb bump consists of deleting the entry $\overline{x}$, shifting the entries $\overline{x+1}, \ldots, \overline{v-1}, v-1, \ldots, u+1$ down one position, and inserting an entry u between $u+1$ and w. The entry $\overline{u}$ is then bumped into the next column.

The fact that the above bumping process is well-defined (i.e. the result of such a bump is a valid C_n tableau) is not obvious. The proofs will be provided in section 8. Moreover, although these special bumps appear foreboding, they actually have a very simple description in terms of elementary Knuth transformations, which we describe in section 6.

3.2 Reverse bumping

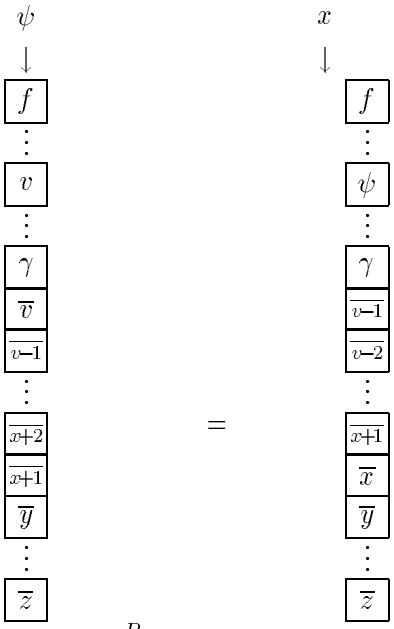
The bumping procedures given in the last section can be reversed. To reverse insert an element ψ into a column C, proceed as in the type A case unless one of the following two situations arise

I^R. if $v \in C$ is the largest letter $\leq \psi$ and $\overline{v} \in C$ also.

II^R. if $\psi = \overline{u}$ and $u \in C$.

Again, it can be seen that such conditions are mutually exclusive. If such conditions arise, one must carry out a "special" reverse bump.

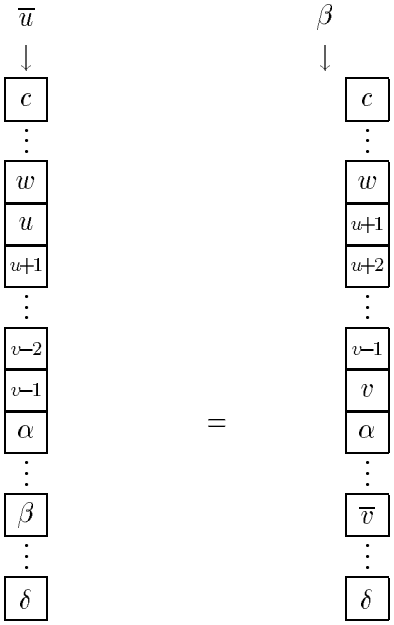
Type $\mathbf{I^R}$ reverse bump



In this case, suppose that there is a string of consecutive letters $\overline{v-1}, \overline{v-2}, \ldots, \overline{x+1}$ underneath $\overline{v}$ with the next letter $\overline{y} > \overline{x}$. The I^R reverse bump consists of replacing the entry v with ψ, deleting the entry $\overline{v}$, moving the string of entries $\overline{v-1}, \ldots, \overline{x+1}$ up one position and inserting $\overline{x}$ into the vacant cell. An entry x is then reverse bumped into the next column.

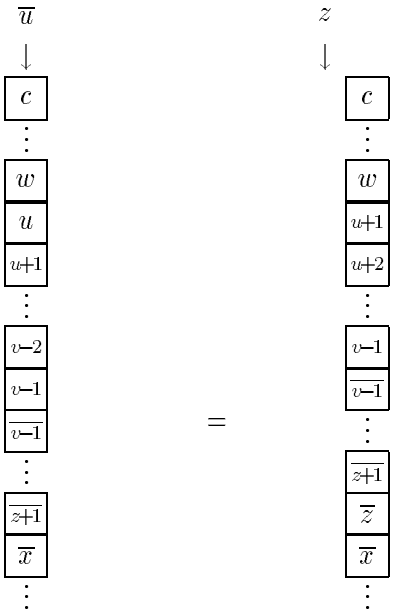
In the case II^R, suppose there is a string of consecutive entries $u+1, u+2, \ldots, v-1$ below the entry $u \in C$. Let $\alpha > v$ be the next entry below $v-1$. Then there are two subcases, depending on whether (a) $v+1 \leq \alpha \leq \overline{v}$ or (b) $\alpha \geq \overline{v-1}$ or no such α exists.

Type II^Ra reverse bump



In this case, β is the largest letter $\leq \overline{v}$ occuring in column C. The II^Ra reverse bump consists of deleting the entry u, shifting the consecutive letters $u+1, u+2, \ldots, v-1$ up one position and inserting the entry v into the vacant cell. The entry β is then replaced by $\overline{v}$ and β is reverse bumped into the next column (on the left).

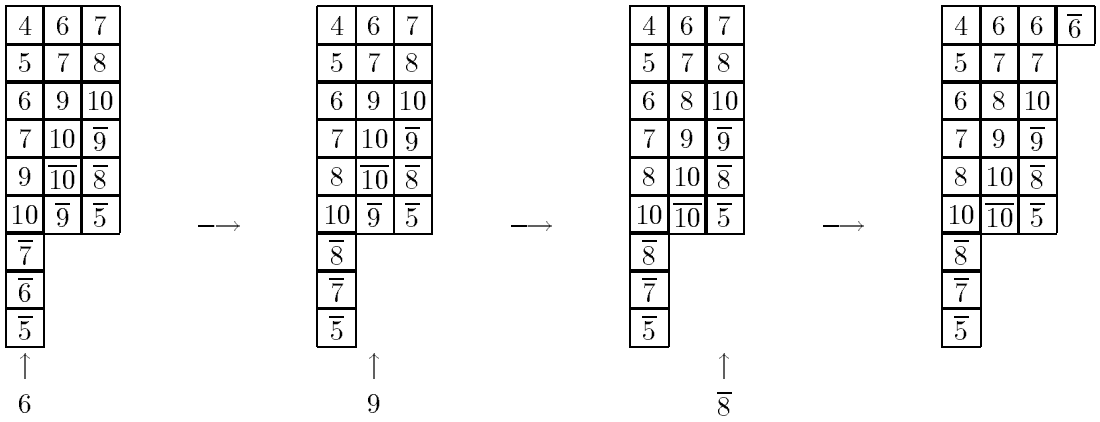
Type II^Rb reverse bump



In this case, suppose $\alpha = \overline{v-1}$ and there is a consecutive string $\overline{v-2}, \ldots, \overline{z+2}, \overline{z+1}$ below α (if $\alpha > \overline{v-1}$ then such a string is empty and $z := v-1$). The II^Rb reverse bump consists of deleting the entry u, moving the consecutive string of letters $u+1, \ldots, v-1, \overline{v-1}, \ldots, \overline{z+1}$ up one position and inserting the entry $\overline{z}$ in the vacant cell. An entry z is then reverse bumped into the next column on the left.

From the definitions it is clear that the reverse bumps I^R, II^Ra, II^Rb are the inverse processes of the forward bumps IIa, I and IIb respectively.

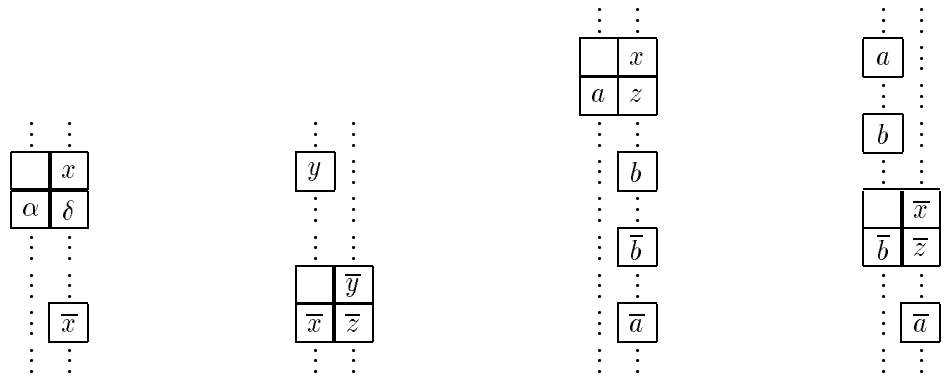
Example. We give an example of the bumping procedures described earlier. In this example, the sequence of bumps is IIa, IIb and then I. By reading the sequence from right to left one can examine the reverse bumps also.



4 Sliding

4.1 Forward sliding

We define forward sliding for C_n tableaux by proceeding as in the (usual) A case unless one of the following four situations is encountered.



The relevant conditions on the entries are as follows

case (i) $x < \alpha$.

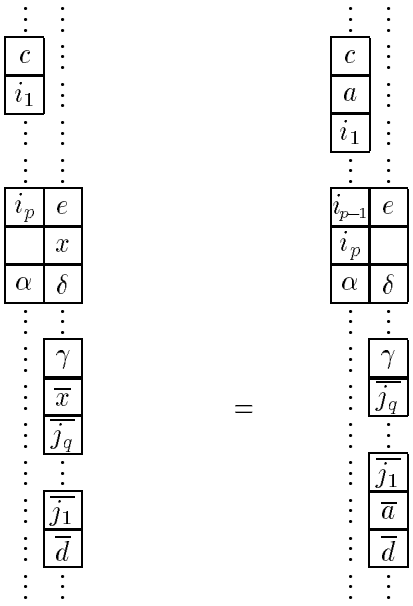
case (ii) $\overline{y} < \overline{x}$.

case (iii) $a \leq x$, $p(a,b|\overline{b},\overline{a}) = b - a - 1$ and b is the smallest integer satisfying such a condition.

case (iv) $\overline{b} \leq \overline{x}$, $p(a,b|\overline{b},\overline{a}) = b - a - 1$ and a is the largest integer satisfying such a condition.

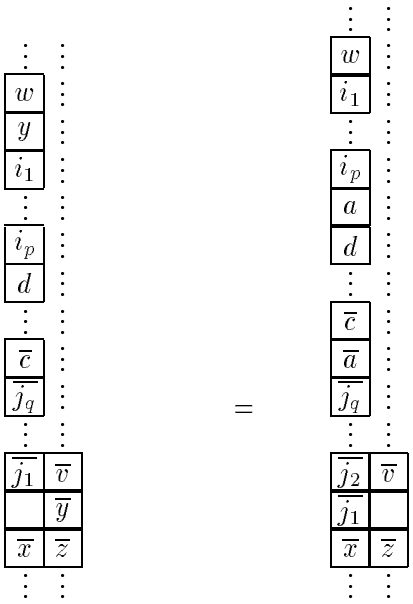
Note that all these cases are mutually exclusive. Thus, in the A_n case, cases (i) and (ii) would correspond to horizontal slides while cases (ii) and (iv) would correspond to vertical slides. In the C_n case, it turns out that all these special slides correspond to horizontal slides (i.e. the empty box moves horizontally) but with some modifications which we now describe.

Type I special slide



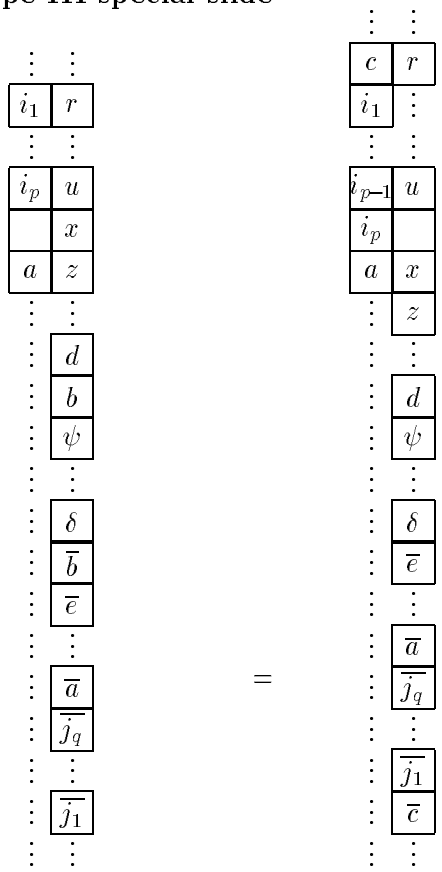
In this diagram, $c \leq a - 1$, $\overline{d} \geq \overline{a-1}$ and $\{i_1,\ldots,i_p\} \sqcup \{j_1,\ldots,j_q\} = \{a+1, a+2,\ldots,x-1\}$. In fact, this union is disjoint as if it were not, it is not difficult to show that such a column would violate the two-column condition. The letter a is thus the largest letter $< x$ which does not appear in the column above the empty box nor (as a barred entry) below the entry $\overline{x}$. The type I slide then consists of: moving the entries $i_1,\ldots,i_p$ down one position and inserting the entry a in the vacant cell; deleting the entry x and replacing it with an empty box; deleting the entry $\overline{x}$, moving the entries $\overline{j_1},\ldots,\overline{j_q}$ up one position and inserting the entry $\overline{a}$ into the vacant cell.

Type II special slide



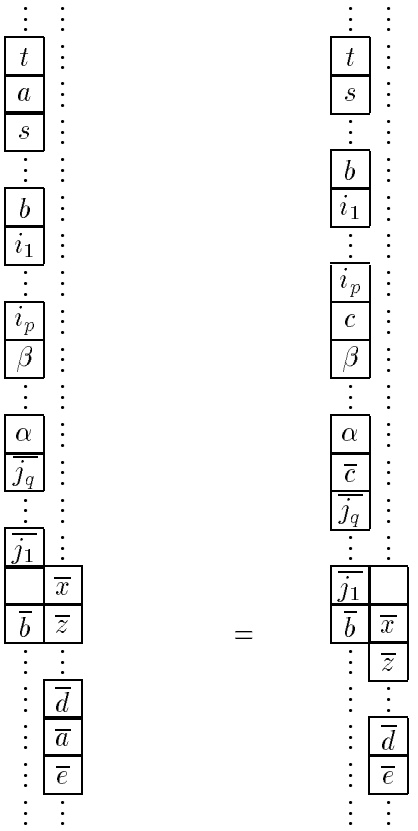
Type III special slide

In this diagram, $d \geq a+1$, $\overline{c} \leq \overline{a+1}$ and $\{i_1, \ldots, i_p\} \sqcup \{j_1, \ldots, j_q\} = \{y+1, y+2, \ldots, a-1\}$. Again, this union is disjoint. Thus a is the smallest letter $> y$ which is not below the entry y, nor (as a barred letter) above the empty box. The type II slide then consists of: deleting the entry y, moving the entries $i_1, \ldots, i_p$ up one position and inserting a into the vacant cell; moving the entries $\overline{j_1}, \ldots, \overline{j_q}$ down one position and inserting an $\overline{a}$ into the vacated cell; deleting $\overline{y}$ in the next column and replacing it with an empty box.



Type IV special slide

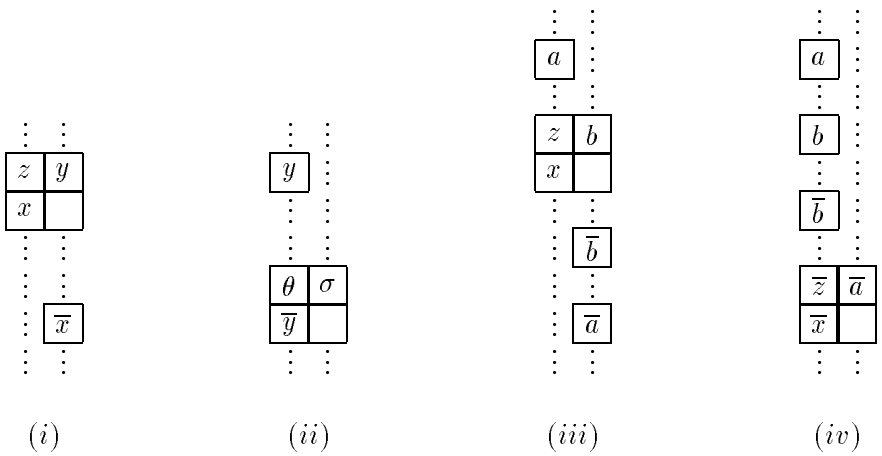
In this diagram, $\{i_1, \ldots, i_p\} \sqcup \{j_1, \ldots, j_q\} = \{c+1, c+2, \ldots, a-1\}$. Thus c is the largest integer $< a$ which does not appear above the empty box nor below the entry $\overline{a}$. The sliding process is as follows: move the entries $i_1, \ldots, i_p$ down one position and insert c into the vacated cell; deleted the entry b and move all entries beginning from the cell to the right of the empty box to the cell immediately above b, down one position, leaving an empty box in the vacated cell; delete the entry $\overline{b}$, move all the entries from the cell below $\overline{b}$ to the entry $\overline{j_q}$ up one position and insert the entry $\overline{c}$ into the vacated cell.



In this diagram, $\{i_1,\dots,i_p\} \sqcup \{j_1,\dots,j_q\} = \{b+1, b+2, \dots, c-1\}$. Hence c here is the smallest integer $> b$ which does not appear below b nor above the empty box. The sliding process is described as follows: delete the entry a, move the entries beginning from the entry below a to the entry i_p up one position and insert an entry c into the vacated cell; move the entries $\overline{j_1},\dots,\overline{j_q}$ down one position and insert the entry $\overline{c}$ into the vacated call; delete the entry $\overline{a}$, move the entries beginning from the cell to the right of the empty box to the cell immediately above $\overline{a}$ down one position and leave an empty box in the vacated cell.

4.2 Reverse sliding

Just as for the bumping procedure, the sliding procedure can be reversed. A type C_n reverse slide consists of an ordinary (type A_n) reverse slide unless one of the following 4 situations is encountered:

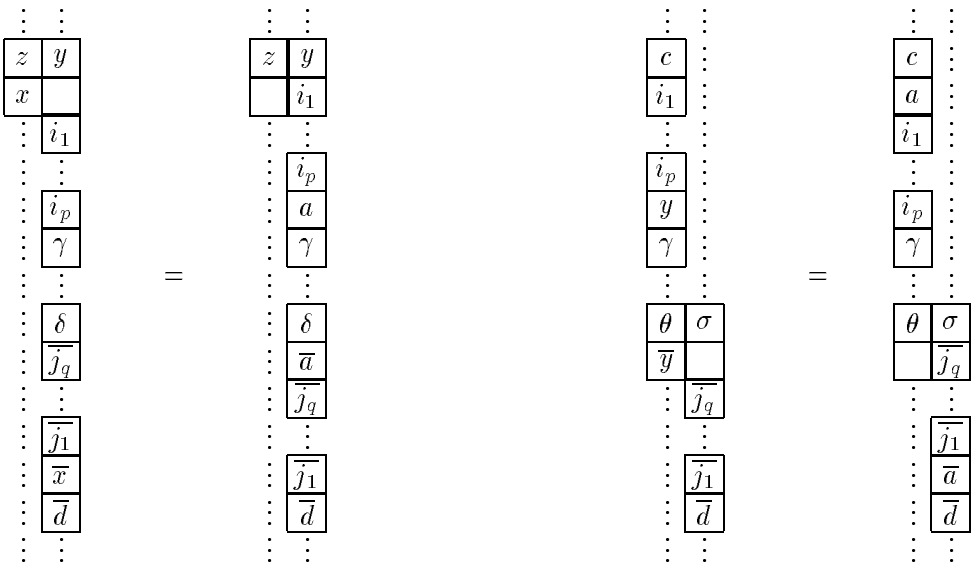


The relevant conditions on the entries are as follows

case (i) $x > y$.

case (ii) $\overline{y} > \sigma$.

case (iii) $x \le b$, $p(a,b|\overline{b},\overline{a}) = b - a - 1$ and a is the largest integer satisfying such a condition.



I^R II^R

Figure 4.1: Type I^R and II^R reverse slides

case (iv) $\overline{x} \leq \overline{b}$, $p(a,b|\overline{b},\overline{a}) = b-a-1$ and b is the smallest integer satisfying such a condition.

Again, note that all these cases are mutually exclusive and that in the A_n case, cases (i) and (ii) would correspond to horizontal slides while cases (iii) and (iv) would correspond to vertical slides. In the C_n case, all these special reverse slides correspond to horizontal reverse slides with some modifications.

In the left of Figure 4.1, $\{i_1,\ldots,i_p\} \sqcup \{j_1,\ldots,j_q\} = \{x+1, x+2, \ldots, a-1\}$, $\gamma \geq a+1$, $\delta \leq \overline{a-1}$. The I^R reverse slide thus proceeds as follows: move the entries $i_1,\ldots,i_p$ up one position into the empty cell, and insert a into the cell vacated; delete the entry $\overline{x}$ and move the entries $\overline{j_1},\ldots,\overline{j_q}$ down one position, inserting $\overline{a}$ into the cell vacated; replace the entry x with an empty box.

Similarly for the II^R reverse slide, $\{i_1,\ldots,i_p\} \sqcup \{j_1,\ldots,j_q\} = \{a+1, a+2, \ldots, y-1\}$, $c \leq a-1$, $\overline{d} \geq \overline{a-1}$ and the slide proceeds as follows: delete the entry y, move the entries $i_1,\ldots,i_p$ down one position and place a into the vacated cell; delete the entry $\overline{y}$ and replace by an empty box; move the entries $\overline{j_1},\ldots,\overline{j_q}$ up one position and insert $\overline{a}$ into the vacated cell.

In Figure 4.2, for the type III^R slide, $\{i_1,\ldots,i_p\} \sqcup \{j_1,\ldots,j_q\} = \{b+1, b+2, \ldots, c-1\}$, $\alpha \geq c+1$ and $\beta \leq \overline{c+1}$. The procedure for the III^R slide is: delete the entry $\overline{a}$ and move all entries from the cell immediately above $\overline{a}$ to $\overline{j_q}$ down one position, inserting the entry $\overline{c}$ into the vacated cell; move the entries $i_1,\ldots,i_p$ up one position, inserting the entry c into the vacated cell; delete the entry a and move all the entries from the cell immediately below a to the cell immediately to the left of the empty box up one position, leaving the vacated cell as the new empty box.

Similarly, in the diagram for the type IV^R slide, $\{i_1,\ldots,i_p\} \sqcup \{j_1,\ldots,j_q\} = \{c+1, c+2, \ldots, a-1\}$, $f \leq c-1$ and $\overline{g} \geq \overline{c-1}$. The procedure for the IV^R slide is: move the entries $\overline{j_1},\ldots,\overline{j_q}$ up one position, placing the entry $\overline{c}$ into the vacated cell; delete the entry $\overline{b}$ and move all the entries from the cell immediately to the left of the empty box to the cell immediately below b up one position, leaving the vacated cell as the new empty box; delete the entry b and move all the entries from i_1 to the cell immediately above b down one position, inserting the entry c into the vacated cell.

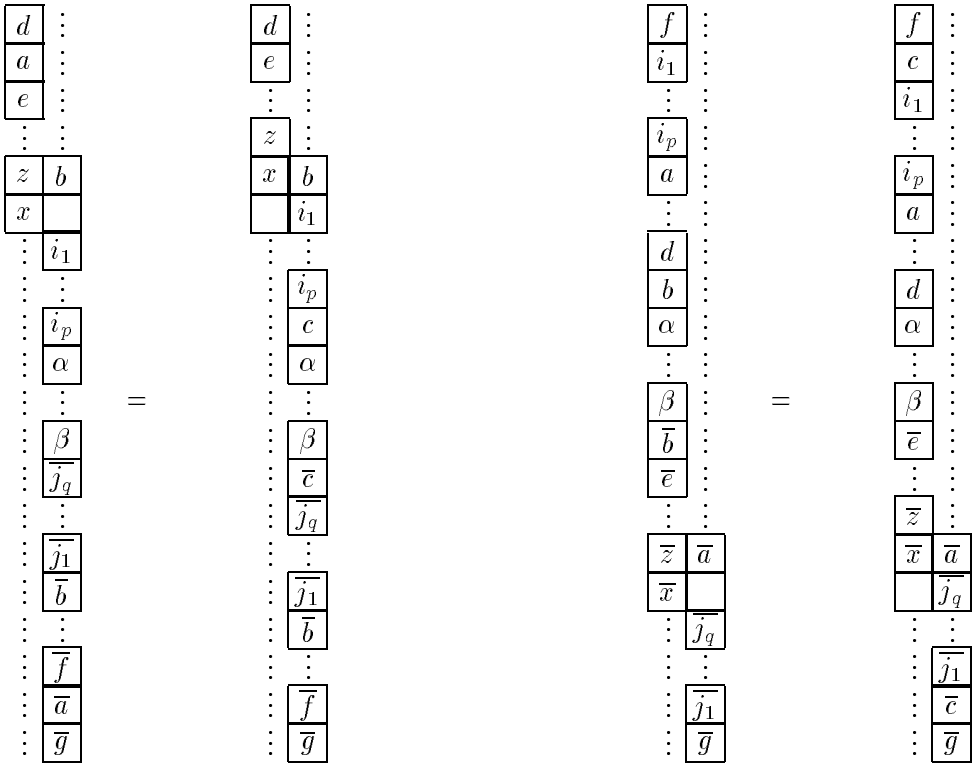
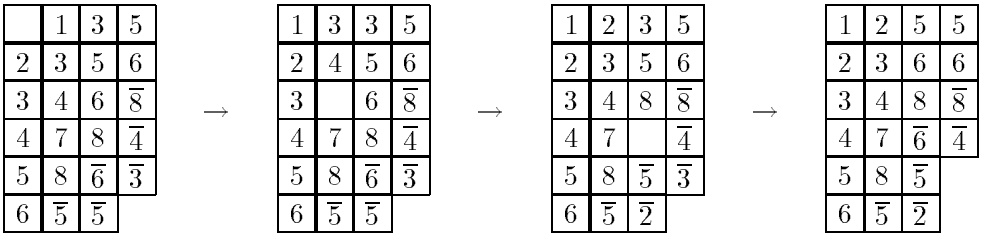


Figure 4.2: Type III^R and IV^R reverse slides

Example. In this example, we give a sequence of forward slides (omitting the vertical slides).



5 The insertion scheme

Having defined bumping and sliding procedures for C_n tableaux, we can now combine them into a procedure for inserting a letter α into a tableau b, the result being a new C_n tableau whose size is one box greater or less than b.

Before doing so, it is instructive to look at the corresponding insertion scheme of Berele [2] when inserting letters into the symplectic tableaux of King [10]. The symplectic tableaux in that case are tableaux on the letters $\{1,2,\ldots,n\} \cup \{\overline{1},\overline{2},\ldots,\overline{n}\}$ with the order $1 < \overline{1} < 2 < \overline{2} < \cdots < n < \overline{n}$. The conditions on the tableaux are threefold: the rows are weakly increasing with respect to the aforementioned order; the columns are strictly increasing; in row i, all the entries are $\geq i$. The Berele insertion consists of *row* bumping, and the bumping procedure is exactly as it is for ordinary (type A) tableaux. However, it may happen that an entry $\overline{i}$ in row i is bumped (by an i) into row $i+1$. This would cause a violation of the symplectic row condition. Berele's solution was that, instead of the $\overline{i}$ being bumped into the next row, it gets annihilated by the i, thus creating an empty box which is then slid out of the tableau using the usual sliding procedure. Let us call the stage in Berele's insertion scheme where bumping becomes sliding, the *bumping–sliding transition.*

Our insertion scheme also has a stage where a bump would cause a violation of the tableau

conditions of Kashiwara and Nakashima, and hence we also have an annihilation process and subsequent sliding. This occurs when an element $\overline{x}$ is inserted into a column and is greater than all the elements in the column, and hence placed at the end of the column. It may happen that a 1CC violation occurs: that is, before the insertion, there were pairs a-$\overline{a}$, b-$\overline{b}$ etc such that $\mathrm{pos}(a) + \mathrm{pos}(\overline{a}) = a$ etc. and the addition of the extra box at the end of the column has increased $\mathrm{pos}(\overline{a})$ etc by one. Our procedure is as follows: locate the smallest a such that $\mathrm{pos}(a) + \mathrm{pos}(\overline{a}) = a + 1$, and replace the cells containing both a and $\overline{a}$ by empty boxes. Now slide these boxes out using the sliding rules described in Section 4. We then have the following result

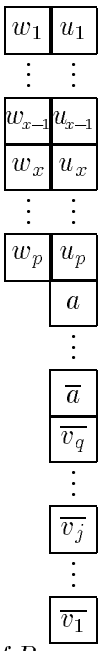
Proposition 5.1

a. The bumping–sliding transition can only occur in the first column.

b. The lower box always slides out (with ordinary slides) vertically.

Before proceeding to the proof, let us first prove a technical result:

Claim: Assume the pair a–$\overline{a}$ in the diagram below satisfies $\mathrm{pos}(a) + \mathrm{pos}(\overline{a}) = a$, and let v_1 be the bottommost element in the column containing such a pair. Then $\{1, 2, \ldots, v_1 - 1\} \subseteq \{w_1, w_2, \ldots, w_p\}$.



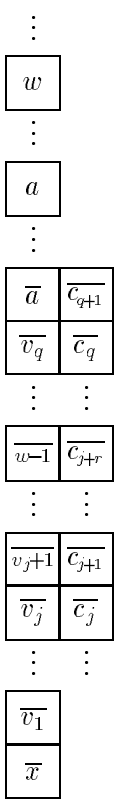
Suppose not and suppose $x < v_1$ is the smallest letter not in $\{w_1, \ldots, w_p\}$. That is, $w_i = i$ for $i = 1, \ldots, x-1$ and $w_x \geq x+1$. We consider 2 cases depending on whether $w_x \leq v_1$ or $w_x > v_1$.

In the former case, Lemma 2.2 implies $p(w_x, a|\overline{a}, \overline{v_1}) \leq a - w_x - 1 \leq a - x - 2$. Now, by the definition of a, $p+q+2 = a$ and thus from the diagram we have $p(w_x, a|\overline{a}, \overline{v_1}) = q+p+1-x = a-x-1$ which is a contradiction.

In the latter case, there exists a j such that $v_{j-1} < w_x \leq v_j$ (we include the case where $j = q+1$ i.e. $v_q < w_x \leq v_{q+1} := a$). From the diagram we see that $p(w_x, a|\overline{a}, \overline{v_j}) = (p + 1 - x) + (q + 1 - j) = a - x - j$. Also, note that the strictly increasing column condition implies $v_{j-1} \geq v_1 + j - 2 > x + j - 2$, so that $v_{j-1} \geq x + j - 1$. Now the inequality $v_{j-1} < w_x \leq v_j$ and Lemma 2.2 implies $p(w_x, a|\overline{a}, \overline{v_j}) \leq a - 1 - \min(w_x, v_j) = a - 1 - w_x < a - 1 - v_{j-1} \leq a - 1 - (x + j - 1) = a - x - j$ which is a contradiction. □

Proof of Prop. 5.1a. Suppose $\overline{z}$ gets appended below $\overline{v_1}$ at some stage in the bumping procedure. That is $\overline{z} > \overline{v_1} \Leftrightarrow z < v_1$. How did $\overline{z}$ arise from the previous column? Certainly not from an ordinary bump, as the above claim implies that z is in the previous column. Thus, $\overline{z}$ must have arisen as a result of a type I or IIb bump. But before such a special bump, z must have been absent from the column, again in contradiction with the above claim. □

Proof of Prop. 5.1b.



We first show that $\overline{v_j} \leq \overline{c_{j+1}}$ for all j such that $\overline{c_{j+1}}$ exists (we include the possibility that $\overline{c_1}$ exists so let us define $\overline{v_0} := \overline{x}$ for ease of notation). Suppose for a contradiction, that $\overline{v_j} > \overline{c_{j+1}} \Leftrightarrow v_j < c_{j+1}$. Now, the fact that the a–$\overline{a}$ pair satisfies $\mathrm{pos}(a)+\mathrm{pos}(\overline{a}) = a+1$ and a is minimal means that the entries above a and below $\overline{a}$ have a disjoint union equal to the set $\{1, 2, \ldots, a-1\}$. Suppose w is the smallest integer $> v_j$ appearing above a. Then the above remark implies that there is a consecutive string $\overline{v_j}, \overline{v_j+1}, \ldots, \overline{w-1}$ appearing below $\overline{a}$. This is depicted in the diagram on the left, with entries $\overline{c_j}, \ldots, \overline{c_{j+r}}$ to their immediate right, where $r := w-1-v_j$. Note that $c_{j+r} \geq c_{j+1} + r - 1 \geq v_j + r = w-1$. Certainly the entries between w and a, and between $\overline{w-1}$ and a form a disjoint union of $\{w+1, w+2, \ldots, a-1\}$. Thus

$$p(w, a|\overline{a}, \overline{c_{j+r}}) = \#\{w-1, w, \ldots, a-1\} = a-w+1$$

But Lemma 2.2 implies $p(w, a|\overline{a}, \overline{c_{j+r}}) \leq a - 1 - \min(w, c_{j+r}) = a - w$ which is a contradiction.

Thus when the $\overline{a}$ is replaced by an empty box and such a box is slid out, the slides are ordinary vertical slides (and in particular never a type IV slide). □

Finally, we show that the bumping–sliding transition can be reversed. That is, after a sequence of reverse slides produces an empty box in the first column, we can recreate the elements a–$\overline{a}$ and their positions in the first column from whence they vanished.

In Figure 5.1(a), suppose there is a pair $i_{p+1} = a$, $\overline{j_{q+1}} = \overline{a}$ such that $p+q+2 = a$, and we insert an $\overline{x}$ such that $\overline{x} > \overline{j_1}$. After we replace the a–$\overline{a}$ pair with a pair of empty boxes, we slide out the bottom box vertically, in accordance with Prop. 5.1 to produce Figure 5.1(b). In this diagram, $\mathrm{pos}(i_p) = p$ $\mathrm{pos}(\overline{j_q}) = q+1$ and i_p (respectively j_q) is the largest (resp. smallest) letter $< a$ (resp. $> \overline{a}$) appearing in the first column.

The condition for the reverse transition is now clear: suppose after a sequence of reverse slides, the empty box is in the first column and let i be the entry in the cell immediately above it. Let a be the smallest integer $> i$ which does not appear in the barred section of the column and let $\overline{j}$ be the smallest integer $> \overline{a}$ such that $\overline{j}$ *does* appear in the barred section of the column. Then if $\mathrm{pos}(i)+\mathrm{pos}(\overline{j}) = a-1$, insert a in the empty box, shift the entries below (and including) $\overline{j}$ down one position, inserting an $\overline{a}$ in the vacated cell, and finally bump the lowest cell in the column out of the tableau.

We must check that the above conditions specify a and the positions of a and $\overline{a}$ in the first column *uniquely*. Suppose they didn't i.e. suppose we continue *forward* sliding vertically down until we reach the situation in Figure 5.1(c) where the above conditions hold (note that we are implicitly using the fact that all vertical slides are ordinary slides, so the entries in the first column don't change). Thus, suppose b is the smallest integer $> i_s$ which doesn't appear among the barred entries, and suppose $\overline{j_t}$ is the smallest integer $> \overline{b}$ such that $\overline{j_t}$ *does* appear and that

$$\mathrm{pos}_A(i_s) + \mathrm{pos}_A(\overline{j_t}) = b - 1.$$

Here, A denotes the column as it appears *after* the bumping–sliding transition. Now, from the definition of i_s and $\overline{j_t}$, there must be a consecutive string below the entry $\overline{j_t}$ ending at $\overline{i_s+1}$

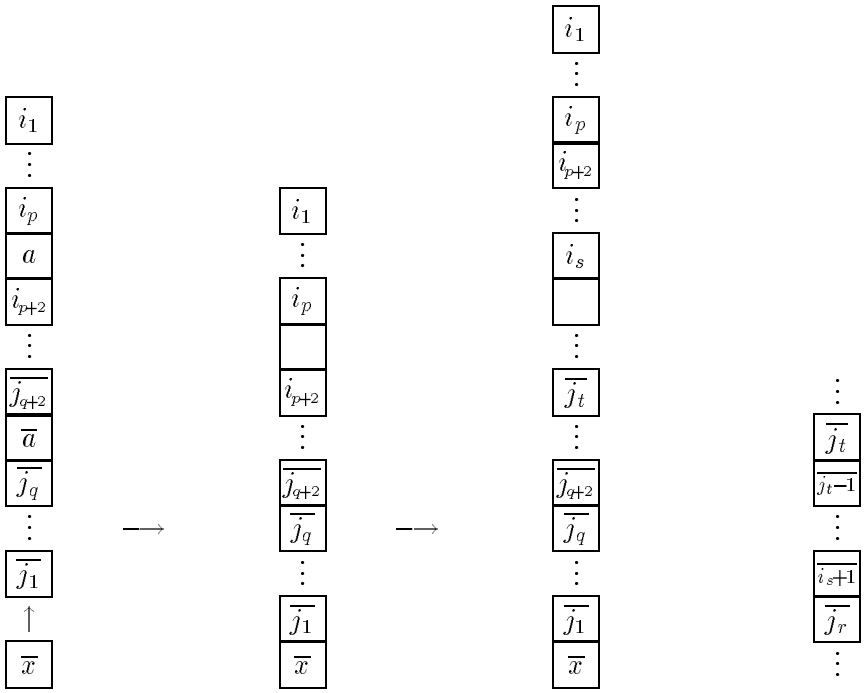


Figure 5.1: Reversibility of bumping–sliding transition

so that the next entry $\overline{j_r}$ is such that $\overline{j_r} \geq \overline{i_s} \Leftrightarrow j_r \leq i_s$ (see Figure 5.1(d)). Let B denote the column as it appears *before* the bump (i.e. in Figure 5.1(a)). Then

$$\mathrm{pos}_B(i_s) = \mathrm{pos}_A(i_s) + 1, \qquad \mathrm{pos}_B(\overline{j_r}) = \mathrm{pos}_A(\overline{j_r}), \qquad \mathrm{pos}_B(i_s) + \mathrm{pos}_B(\overline{j_r}) \leq i_s \tag{5.1}$$

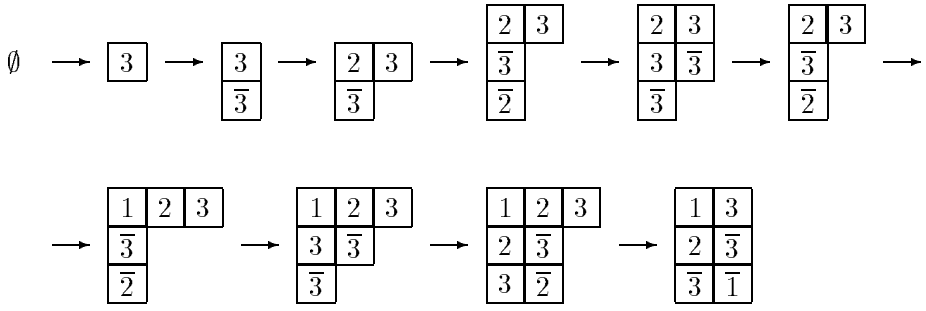
the last inequality following from Lemma 2.1. Since $\mathrm{pos}_A(\overline{j_t}) = \mathrm{pos}_A(\overline{j_r}) + j_t - i_s$ we have from (5.1)

$$\mathrm{pos}_A(i_s) + \mathrm{pos}_A(\overline{j_t}) = \mathrm{pos}_A(i_s) + \mathrm{pos}_A(\overline{j_r}) + j_t - i_s \leq (j_t - i_s) + i_s - 1 = j_t - 1 < b - 1$$

which is a contradiction. □

Thus inserting a letter α into a tableau b results in a new C_n tableau $b \leftarrow \alpha$ whose shape is one box larger or smaller than the shape of b. Thus we can define a correspondence $w \leftrightarrow (P(w), Q(w))$ where $P(w) := (\cdots((\emptyset \leftarrow w_1) \leftarrow w_2) \cdots \leftarrow w_p)$ and $Q(w)$ is an *oscillating* tableau i.e. a sequence of Young diagrams whose shape differs by one [26].

Example With $w = 3\overline{3}2\overline{2}2\overline{2}123\overline{1}$ we have the following sequence of insertions:



Thus we can associate the word $w = (3\overline{3}2\overline{2}2\overline{2}123\overline{1})$ with the pair

$$\left(\begin{array}{|c|c|}\hline 1 & 3 \\ \hline 2 & \overline{3} \\ \hline \overline{3} & \overline{1} \\ \hline \end{array} \quad ; \quad \ydiagram{2,2,2,1,1,1

A similar analysis reveals that the IIa and IIb bumps can also be analysed in terms of sequences of elementary Knuth relations, so that

$$\begin{aligned} \text{type IIa bump} &\quad\leftrightarrow\quad \mathrm{K1}^d\ \mathrm{K2}^c\ \mathrm{K4}^b\ \mathrm{K2}^a \\ \text{type IIb bump} &\quad\leftrightarrow\quad \mathrm{K1}^d\ \mathrm{K3}^c\ \mathrm{K4}^b\ \mathrm{K2}^a \end{aligned}$$

There is one final Knuth transformation which governs the transformation of the column reading of the tableaux in the insertion scheme, namely when there is a bumping–sliding transition in the first column. This Knuth transformation reads

$$w_1\, w_2 \cdots w_p\, a\, \beta_1 \cdots \beta_r\, \overline{a}\, \overline{v_q} \cdots \overline{v_1} \;\rightarrow\; w_1\, w_2 \cdots w_p\, \beta_1 \cdots \beta_r\, \overline{v_q} \cdots \overline{v_1} \qquad \textbf{(K5)}$$

where $p+q = a-1$, $w_1 < \cdots < w_p < a < \beta_1 < \cdots < \beta_r < \overline{a} < \overline{v_q} < \cdots < \overline{v_1}$ and $\{w_1, \ldots, w_p\} \cap \{v_1, \ldots, v_q\} = \emptyset$.

We note here that the C_n Knuth relations (K1)–(K5) were first derived by Lascoux, Leclerc and Thibon [15] (see also [17]).

6.2 Sliding and Knuth transformations

In this section we aim to show that the sliding procedures introduced in Section 4 preserve the Knuth relations (K1)–(K5) (the fact that the ordinary slides preserve K1 and K2 is well-known). To begin with, let us first consider the type I slide. To show that they are related by a sequence of Knuth transformations, it is sufficient to take the readings of the 2 columns which change under the slide, and insert them into the empty tableau: if they produce the same (2 column) tableau, then we are done.

Note that such a process is not trivial, even in the case of inserting the column reading of a 2-column tableaux into the empty tableau. In particular given a 2-column tableau with columns C_1 and C_2, it may happen that there are pairs a–$\overline{a}$ where $a \in C_1$ and $\overline{a} \in C_2$. Let us denote the set of such elements as $C_1^+ \cap C_2^-$. In this case for example, after having inserted the letters of C_2 into the empty tableau, there will exist an $\overline{a}$ in the first column and so when we come to insert the letter a, we will require a type IIa or IIb bump. In general we can group the barred elements of C_2 into disjoint groups of consecutive strings of letters containing elements of $C_1^+ \cap C_2^-$. Clearly any such elements must lie consecutively in C_1. As an example, let us consider the case where there are 2 elements in $C_1^+ \cap C_2^-$ so that the column reading of such a tableau reads thus

$$w = \begin{pmatrix} c_1 & \cdots & c_p & c_{p+1} & c_{p+2} & c_{p+3} & \cdots & c_t & \overline{d_1} & \cdots & \overline{d_v} & \overline{u-1} & \cdots & \overline{f} & \cdots & \overline{e} \\ a_1 & \cdots & a_p & e & f & \alpha_1 & \cdots & \cdots & \cdots & \cdots & \alpha_q & \beta_{u-1} & \cdots & \beta_f & \cdots & \beta_e \end{pmatrix}$$

Here for convenience, we write the word of the tableau in such a way as to reflect the relative positions of the elements in the tableau. We assume for simplicity that $C_1^+ \cap C_2^- = \{e, f\}$ (and hence $\overline{u-1} \cdots \overline{f} \cdots \overline{e}$ are consecutive integers) and that $\overline{e}$ lies at the bottom of the column. Note that Lemma 2.2 applied to the $(e, c_{p+1} | \overline{u-1}, \overline{e})$ configuration implies that that $c_{p+1} \geq u$ (and so $c_{p+j} \geq u+j-1$ for $1 \leq j \leq t-p$). Similar considerations show that $\beta_j \leq \overline{j+1}$ for $u-1 \leq j \leq e$.

With these relations in mind the sequence of insertions is as follows (inserting the letters of C_2 as well as $a_1, \ldots, a_p$ is clear and hence omitted)

$$\begin{pmatrix} c_1 & \cdots & c_p & & & & & & & & & & & & & \\ a_1 & \cdots & a_p & c_{p+1} & c_{p+2} & c_{p+3} & \cdots & c_t & \overline{d_1} & \cdots & \overline{d_v} & \overline{u-1} & \cdots & \overline{f} & \cdots & \overline{e} \end{pmatrix}$$

$$\xrightarrow[\text{IIa}]{e} \quad \begin{pmatrix} c_1 & \cdots & c_p & c_{p+1} & & & & & & & & & & & & \\ a_1 & \cdots & a_p & u & c_{p+2} & c_{p+3} & \cdots & c_t & \overline{d_1} & \cdots & \overline{d_v} & \overline{u} & \cdots & \overline{f} & \cdots & \overline{e+1} \end{pmatrix}$$

$$
\begin{array}{cc}
\xrightarrow[\text{IIa}]{f} & \begin{pmatrix} c_1 & \cdots & c_p & c_{p+1} & c_{p+2} & & & & & & & & & & & & \\ a_1 & \cdots & a_p & u & u+1 & c_{p+3} & \cdots & c_t & \overline{d_1} & \cdots & \overline{d_v} & \overline{u+1} & \cdots & \overline{f+1} & \overline{f-1} & \cdots & \overline{e+1} \end{pmatrix} \\
\xrightarrow{\{\alpha_1,\ldots,\alpha_q\}} & \begin{pmatrix} c_1 & \cdots & c_p & c_{p+1} & c_{p+2} & c_{p+3} & \cdots & c_t & \overline{d_1} & \cdots & \overline{d_v} & & & & & & \\ a_1 & \cdots & a_p & u & u+1 & \alpha_1 & \cdots & \cdots & \cdots & \alpha_q & \overline{u+1} & \cdots & \overline{f+1} & \overline{f-1} & \cdots & \overline{e+1} \end{pmatrix} \\
\xrightarrow[\text{I}]{\beta_{u-1}} & \begin{pmatrix} c_1 & \cdots & c_p & c_{p+1} & c_{p+2} & c_{p+3} & \cdots & c_t & \overline{d_1} & \cdots & \overline{d_v} & \overline{u-1} & & & & & \\ a_1 & \cdots & a_p & u-1 & u & \alpha_1 & \cdots & \cdots & \cdots & \alpha_q & \beta_{u-1} & \cdots & \overline{f+1} & \overline{f-1} & \cdots & \overline{e+1} \end{pmatrix} \\
\vdots & \\
\xrightarrow[\text{I}]{\beta_{f-1}} & \begin{pmatrix} c_1 & \cdots & c_p & c_{p+1} & c_{p+2} & c_{p+3} & \cdots & c_t & \overline{d_1} & \cdots & \overline{d_v} & \overline{u-1} & \cdots & \overline{f-1} & & & \\ a_1 & \cdots & a_p & f-1 & f & \alpha_1 & \cdots & \cdots & \cdots & \alpha_q & \beta_{u-1} & \cdots & \beta_{f-1} & \overline{f-1} & \cdots & \overline{e+1} \end{pmatrix} \\
\xrightarrow[\text{I}]{\beta_{f-2}} & \begin{pmatrix} c_1 & \cdots & c_p & c_{p+1} & c_{p+2} & c_{p+3} & \cdots & c_t & \overline{d_1} & \cdots & \overline{d_v} & \overline{u-1} & \cdots & \overline{f-1} & \overline{f-2} & & \\ a_1 & \cdots & a_p & f-2 & f & \alpha_1 & \cdots & \cdots & \cdots & \alpha_q & \beta_{u-1} & \cdots & \beta_{f-1} & \beta_{f-2} & \cdots & \overline{e+1} \end{pmatrix} \\
\vdots & \\
\xrightarrow[\text{I}]{\beta_e} & \begin{pmatrix} c_1 & \cdots & c_p & c_{p+1} & c_{p+2} & c_{p+3} & \cdots & c_t & \overline{d_1} & \cdots & \overline{d_v} & \overline{u-1} & \cdots & \overline{f} & \cdots & \overline{e} \\ a_1 & \cdots & a_p & e & f & \alpha_1 & \cdots & \cdots & \cdots & \cdots & \alpha_q & \beta_{u-1} & \cdots & \beta_f & \cdots & \beta_e \end{pmatrix}
\end{array} \tag{6.1}
$$

The cases where the consecutive string in C_2^- contains more than 2 elements, follow a similar pattern as above. We can also consider the case where there are several *disjoint* consecutive strings containing elements of C_1^+, in which case the highest block could give rise to type IIb bumps instead of IIa bumps, but the details are similar.

Turning to the type I slide, let us consider the case where there are no entries above $i_1,\ldots,i_p$ nor below $\overline{j_q},\ldots,\overline{j_1}$ nor between x and $\overline{x}$. The column reading of such a configuration looks thus :

$$
w = \begin{pmatrix} c_1 & \cdots & \cdots & \cdots & \cdots & \cdots & c_t & x & \overline{x} & \overline{j_q} & \cdots & \overline{j_1} \\ a+1 & \cdots & < j_1 > & \cdots & < j_q > & \cdots & x-1 & \square & \alpha_{q+1} & \alpha_q & \cdots & \alpha_1 \end{pmatrix}
$$

Here $t = x-a-q-1$ and we consider the entries $i_1,\ldots,i_p$ as the consecutive string $a+1, a+2,\ldots,x-2,x-1$ punctuated by the absences of $j_1,\ldots,j_q$. If we now start inserting this word into the empty tableau we get the following sequence of bumps (again, inserting the upper row and the letters up to $x-1$ in the bottom row is clear and hence omitted)

$$
\begin{array}{cc}
 & \begin{pmatrix} c_1 & \cdots & c_{j_q-a-q} & c_{j_q-a-q+1} & \cdots & c_t & & & & & \\ a+1 & \cdots & j_q-1 & j_q+1 & \cdots & x-1 & x & \overline{x} & \overline{j_q} & \cdots & \overline{j_1} \end{pmatrix} \\
\xrightarrow[\text{I}]{\alpha_{q+1}} & \begin{pmatrix} c_1 & \cdots & c_{j_{q-1}-a-q+1} & c_{j_{q-1}-a-q+2} & \cdots & c_{j_q-a-q} & c_{j_q-a-q+1} & \cdots & c_t & \overline{j_q} & & & & \\ a+1 & \cdots & j_{q-1}-1 & j_{q-1}+1 & \cdots & j_q-1 & j_q & \cdots & x-2 & x-1 & \alpha_{q+1} & \overline{j_q} & \cdots & \overline{j_1} \end{pmatrix} \\
\xrightarrow[\text{I}]{\alpha_q} & \begin{pmatrix} c_1 & \cdots & c_{j_{q-1}-a-q+1} & c_{j_{q-1}-a-q+2} & \cdots & c_{j_q-a-q+1} & c_{j_q-a-q+2} & \cdots & c_t & \overline{j_q} & \overline{j_{q-1}} & & & \\ a+1 & \cdots & j_{q-1}-1 & j_{q-1} & \cdots & j_q-1 & j_q+1 & \cdots & x-2 & x-1 & \alpha_{q+1} & \alpha_q & \overline{j_{q-1}} & \cdots & \overline{j_1} \end{pmatrix} \\
\vdots & \\
\xrightarrow[\text{I}]{\alpha_1} & \begin{pmatrix} c_1 & \cdots & c_{j_k-a-k+1} & c_{j_k-a-k+2} & \cdots & c_t & \overline{j_q} & \overline{j_{q-1}} & \cdots & \overline{j_1} & \overline{a} \\ a & \cdots & j_k-1 & j_k+1 & \cdots & x-2 & x-1 & \alpha_{q+1} & \cdots & \alpha_3 & \alpha_2 & \alpha_1 \end{pmatrix}, \qquad 1 \le k \le q
\end{array}
$$

which is precisely the column reading of the end product of the type I slide. For the general case of a type I slide, one simply combines the two types of insertions shown above. This is possible since there is always a "gap" between $\overline{j_1}$ and the entry immediately below it, as well as between i_1 and the entry immediately above it, so the above 2 types of sequences of insertions are "disjoint" (i.e. do not interfere with each other).

The case of type II slides is a bit more complicated. Let us consider the simplest case where there is only one i. That is, we want to show the following Knuth equivalence

$$\begin{pmatrix} \gamma_0 & \gamma_1 & \overline{a-1} & \cdots & \overline{k+1} & \overline{k-1} & \cdots & \overline{i} & \overline{i-1} & \cdots & \overline{y+1} & \overline{y} \\ y & i & \overline{a-1} & \cdots & \overline{k+1} & \overline{k} & \cdots & \overline{i+1} & \overline{i-1} & \cdots & \overline{y+1} & \square \end{pmatrix}$$
$$\sim \begin{pmatrix} \gamma_0 & \gamma_1 & \overline{a-1} & \cdots & \overline{k+1} & \overline{k-1} & \cdots & \overline{i-1} & \overline{i-2} & \cdots & \overline{y+1} & \\ i & a & \overline{a} & \cdots & \overline{k+2} & \overline{k+1} & \cdots & \overline{i+1} & \overline{i-1} & \cdots & \overline{y+2} & \overline{y+1} \end{pmatrix} \tag{6.2}$$

Here, we consider the string $\overline{j_1},\ldots,\overline{j_q}$ as the consecutive string $y+1, y+2, \ldots, a-1$ punctuated by the absence of i. We also consider the entries to the immediate right of the j's as the string $y+1, y+2, \ldots, a-1$ punctuated by the absence of k and column strictness implies that $k \geq i$. Note also that Lemma 2.2 implies (using an argument by contradiction) that $\gamma_0 \geq k$ and $\gamma_1 \geq a$. With these facts in mind, we can insert the word on the left hand side of (6.2) into the empty tableau, the result of which is the following sequence of insertions

$$(\gamma_0\ \gamma_1\ \overline{a-1}\ \cdots \overline{k+1}\ \overline{k-1}\ \cdots\ \overline{i}\ \overline{i-1}\ \cdots\ \overline{y+1}\ \overline{y})$$
$$\xrightarrow[\text{IIa}]{y} \begin{pmatrix} \gamma_0 & & & & & & & & & & & \\ k & \gamma_1 & \overline{a-1} & \cdots & \overline{k+1} & \overline{k} & \overline{k-1} & \cdots & \overline{i} & \overline{i-1} & \cdots & \overline{y+1} \end{pmatrix}$$
$$\xrightarrow[\text{IIa}]{i} \begin{pmatrix} \gamma_0 & \gamma_1 & & & & & & & & & & \\ k & a & \overline{a} & \overline{a-1} & \cdots & \overline{k+1} & \overline{k} & \cdots & \overline{i+1} & \overline{i-1} & \cdots & \overline{y+1} \end{pmatrix}$$
$$\xrightarrow{\{\overline{a-1},\ldots,\overline{k+1}\}} \begin{pmatrix} \gamma_0 & \gamma_1 & \overline{a-1} & \overline{a-2} & \cdots & \overline{k+1} & & & & & & \\ k & a & \overline{a} & \overline{a-1} & \cdots & \overline{k+2} & \overline{k+1} & \cdots & \overline{i+1} & \overline{i-1} & \cdots & \overline{y+1} \end{pmatrix}$$
$$\xrightarrow[\text{I}]{\overline{k}} \begin{pmatrix} \gamma_0 & \gamma_1 & \overline{a-1} & \cdots & \overline{k+1} & \overline{k-1} & & & & & & \\ k-1 & a & \overline{a} & \cdots & \overline{k+2} & \overline{k+1} & \cdots & \overline{i+1} & \overline{i-1} & \cdots & \overline{y+1} \end{pmatrix}$$
$$\xrightarrow[\text{I}]{\{\overline{k-1},\ldots,\overline{i+1}\}} \begin{pmatrix} \gamma_0 & \gamma_1 & \overline{a-1} & \cdots & \overline{k+1} & \overline{k-1} & \cdots & \overline{i} & & & & \\ i & a & \overline{a} & \cdots & \overline{k+2} & \overline{k+1} & \cdots & \overline{i+1} & \overline{i-1} & \cdots & \overline{y+1} & \end{pmatrix}$$
$$\xrightarrow{\{\overline{i-1},\ldots,\overline{y+1}\}} \begin{pmatrix} \gamma_0 & \gamma_1 & \overline{a-1} & \cdots & \overline{k+1} & \overline{k-1} & \cdots & \overline{i-1} & \overline{i-2} & \cdots & \overline{y+1} & \\ i & a & \overline{a} & \cdots & \overline{k+2} & \overline{k+1} & \cdots & \overline{i+1} & \overline{i-1} & \cdots & \overline{y+2} & \overline{y+1} \end{pmatrix}$$

and the proof of (6.2) is complete. The situation for the more general case of $i_1,\ldots,i_p$ is similar (although more complicated). Again, we must combine such a sequence of insertions with the sequence (6.1) for elements in $C_1^+ \cap C_2^-$ (apart from y), but again, such sequences are "disjoint" for reasons mentioned above, so there is no problem.

For type III and IV slides, the arguments are similar as for the type I and II slides respectively. Thus, since both the bumping *and* sliding processes preserve Knuth equivalence, we have in fact proved the following

Proposition 6.1 *If $b \leftarrow \alpha$ denotes the tableau which results after inserting α into the tableau b, then*

$$w(b \leftarrow \alpha) \sim w(b).\alpha$$

6.3 Stability of Knuth classes under $\tilde{e}_i$, $\tilde{f}_i$

In this subsection, we shall show that the Knuth classes are stable under the action of the Kashiwara operators. We consider only the case for the lowering operators $\tilde{f}_i$ with $1 \leq i \leq n-1$, as the proofs for the case $i = n$ and for $\tilde{e}_i$ are similar. Thus, we must show that if $w \sim w'$ then $\tilde{f}_i w \sim \tilde{f}_i w'$. It suffices to consider the cases when w and w' are related by an elementary Knuth relations. We first consider the transformations (K1)–(K4) and treat (K5) separately.

Recalling the action of $\tilde{f}_i$ given in Section 2.2, let us call the letter upon which $\tilde{f}_i$ acts, the *pivot* letter. For the transformations (K1)–(K4), it can be seen that whenever the pivot letter is *not* part of the 3 letters comprising the transformation, the pivot letter remains unchanged and hence the result follows immediately. Thus we can restrict ourselves to the cases when the pivot letter *is* one of the 3 letters comprising the transformation.

In the case of a K1 or K2 transformation, the pivot letter can only change when the 2 letters which get swapped are of the form u_+u_- or u_-u_+ in the notation of section 2.2. In this case, the inequalities governing the transformations leave us with the following 2 situations where the pivot letter changes

$$\begin{array}{ccc} i\ i{+}1\ \boldsymbol{i} & \xrightarrow{\text{K2}} & \boldsymbol{i}\ i\ i{+}1 \\ \downarrow \tilde{f}_i & & \downarrow \tilde{f}_i \\ i\ i{+}1\ i{+}1 & \xrightarrow{\text{K1}} & i{+}1\ i\ i{+}1 \end{array} \qquad \begin{array}{ccc} \overline{i{+}1}\ \overline{i}\ \overline{\boldsymbol{i{+}1}} & \xrightarrow{\text{K2}} & \overline{\boldsymbol{i{+}1}}\ \overline{i{+}1}\ \overline{i} \\ \downarrow \tilde{f}_i & & \downarrow \tilde{f}_i \\ \overline{i{+}1}\ \overline{i}\ \overline{i} & \xrightarrow{\text{K1}} & \overline{i}\ \overline{i{+}1}\ \overline{i} \end{array}$$

where the pivot letter is shown in bold. Hence in these cases, the images under $\tilde{f}_i$ are also related by an elementary Knuth transformation.

For K1 and K2 transformations, we must also consider the case when the letters getting swapped are of the form u_+u_+ but are of different type i.e. $i\ \overline{i{+}1}$ or $\overline{i{+}1}\ i$ since in these cases, we have the following situation

$$\begin{array}{ccc} \alpha\ \overline{\boldsymbol{i{+}1}}\ i & \xrightarrow{\text{K2}} & \alpha\ \boldsymbol{i}\ \overline{i{+}1} \\ \downarrow \tilde{f}_i & & \downarrow \tilde{f}_i \\ \alpha\ \overline{i}\ i & \xrightarrow{\text{K4}} & \alpha\ i{+}1\ \overline{i{+}1} \end{array} \quad i < \alpha < \overline{i+1} \qquad \begin{array}{ccc} \boldsymbol{i}\ \overline{i{+}1}\ \beta & \xrightarrow{\text{K1}} & \overline{\boldsymbol{i{+}1}}\ i\ \beta \\ \downarrow \tilde{f}_i & & \downarrow \tilde{f}_i \\ i{+}1\ \overline{i{+}1}\ \beta & \xrightarrow{\text{K3}} & \overline{i}\ i\ \beta \end{array} \quad i < \beta \leq \overline{i+1}$$

Note that in the first case, the K2 inequalities allows for $\alpha = i$ but it is easy to show that in such a case, α becomes the pivot letter and the images under $\tilde{f}_i$ remain related by a K2 transformation.

For K3 and K4 transformations, a similar situation occurs, namely

$$\begin{array}{ccc} \alpha\ \overline{i}\ \boldsymbol{i} & \xrightarrow{\text{K4}} & \alpha\ i{+}1\ \overline{\boldsymbol{i{+}1}} \\ \downarrow \tilde{f}_i & & \downarrow \tilde{f}_i \\ \alpha\ \overline{i}\ i{+}1 & \xrightarrow{\text{K2}} & \alpha\ i{+}1\ \overline{i} \end{array} \quad i < \alpha < \overline{i} \qquad \begin{array}{ccc} i{+}1\ \overline{\boldsymbol{i{+}1}}\ \beta & \xrightarrow{\text{K3}} & \overline{i}\ \boldsymbol{i}\ \beta \\ \downarrow \tilde{f}_i & & \downarrow \tilde{f}_i \\ i{+}1\ \overline{i}\ \beta & \xrightarrow{\text{K1}} & \overline{i}\ i{+}1\ \beta \end{array} \quad i+1 < \beta < \overline{i}$$

Again, note that the K3 inequalities allow for $\beta = i + 1$ but in such a case the middle letter is never the pivot letter, so there is never a problem.

Finally, turning to the K5 transformation, it suffices to consider only the cases where the pivot letter is annihilated. Hence for each i, there are 2 subcases (a) the pivot letter is i (b) the pivot letter is $\overline{i+1}$.

In subcase (a), suppose $w = w_1 \cdots w_p\ \boldsymbol{i}\ \beta_1 \cdots \beta_r\ \overline{i}\ \overline{v_q} \cdots \overline{v_1}$ with $p + q = i - 1$. Then the only way i can be a pivot letter is if $\beta_1 \geq i + 2$ and $\beta_r = \overline{i+1}$. In such a situation we have the

following diagram

$$\begin{array}{ccc} w_1\cdots w_p\ \boldsymbol{i}\ \beta_1\cdots\overline{i+1}\ \overline{i}\ \overline{v_q}\cdots\overline{v_1} & \xrightarrow{\text{K5}} & w_1\cdots w_p\ \beta_1\cdots\overline{\boldsymbol{i+1}}\ \overline{v_q}\cdots\overline{v_1} \\ \downarrow \tilde{f}_i & & \downarrow \tilde{f}_i \\ w_1\cdots w_p\ i{+}1\ \beta_1\cdots\overline{i+1}\ \overline{i}\ \overline{v_q}\cdots\overline{v_1} & \xrightarrow{\text{K5}} & w_1\cdots w_p\ \beta_1\cdots\overline{i}\ \overline{v_q}\cdots\overline{v_1} \end{array}$$

Note that the condition $p+q=i-1$ is the necessary K5 condition for both w and $\tilde{f}_i w$. In case (b), a similar argument holds so that the only non-trivial case is the following

$$\begin{array}{ccc} w_1\cdots w_{p-1}\ i\ i{+}1\ \beta_1\cdots\beta_r\ \overline{\boldsymbol{i+1}}\ \overline{v_q}\cdots\overline{v_1} & \xrightarrow{\text{K5}} & w_1\cdots w_{p-1}\ \boldsymbol{i}\ \beta_1\cdots\beta_r\ \overline{v_q}\cdots\overline{v_1} \\ \downarrow \tilde{f}_i & & \downarrow \tilde{f}_i \\ w_1\cdots w_{p-1}\ i\ i{+}1\ \beta_1\cdots\beta_r\ \overline{i}\ \overline{v_q}\cdots\overline{v_1} & \xrightarrow{\text{K5}} & w_1\cdots w_{p-1}\ i{+}1\ \beta_1\cdots\beta_r\ \overline{v_q}\cdots\overline{v_1} \end{array}$$

where $p+q=i$ and $\overline{v_q}\geq\overline{i-1}$.

Summarizing, we have proven the following

Proposition 6.2 *The Kashiwara operators are compatible with the Knuth equivalence classes:*

$$\tilde{e}_j\, w_1 \sim \tilde{e}_j\, w_2 \ \Leftrightarrow\ w_1 \sim w_2 \ \Leftrightarrow\ \tilde{f}_j\, w_1 \sim \tilde{f}_j\, w_2$$

for all $1\leq j\leq n$.

Remark. It follows from the above proposition that the Knuth equivalence classes have unique representative which is the word of a C_n tableau. This is because from Proposition 6.2 and the fact that every word is connected to a highest-weight (Yamanouchi) word by a sequence $\tilde{e}_{i_1}\cdots\tilde{e}_{i_p}$ of raising operators, it is only necessary to consider the case where the Knuth class contains highest-weight words, in which case the result follows from the A_{n-1} case. It would be nice to have a direct (i.e. combinatorial) argument of this statement in terms of analyzing longest increasing subsequences and an induction argument, à la [5, Chapt. 3].

7 Crystal isomorphism

Recall that we use the same symbols $\tilde{f}_i$, $\tilde{e}_i$ to denote the action of the Kashiwara operators on tableaux or on words. Thus if w denotes the operator which takes the column reading of a tableau, we have symbolically, $w\tilde{f}_i=\tilde{f}_i w$.

A very important consequence of Props 6.1 and 6.2 is that

$$w(\tilde{f}_j(b\leftarrow\alpha))\sim\tilde{f}_j(w(b).\alpha) \tag{7.1}$$

In particular, this implies that for $1\leq j<n$

$$\phi_j(b\leftarrow i)=\begin{cases}\max(\phi_j(b)-1,0) & j=i-1\\ \phi_j(b)+1 & j=i\\ \phi_j(b) & \text{otherwise}\end{cases}\qquad \phi_j(b\leftarrow\overline{i})=\begin{cases}\phi_j(b)+1 & j=i-1\\ \max(\phi_j(b)-1,0) & j=i\\ \phi_j(b) & \text{otherwise}\end{cases} \tag{7.2}$$

while for $j=n$

$$\phi_n(b\leftarrow i)=\begin{cases}\phi_n(b)+1 & i=n\\ \phi_n(b) & \text{otherwise}\end{cases}\qquad \phi_n(b\leftarrow\overline{i})=\begin{cases}\max(\phi_n(b)-1,0) & i=n\\ \phi_n(b) & \text{otherwise}\end{cases} \tag{7.3}$$

along with similar expressions for $\epsilon_j(b\leftarrow i)$ etc. These are the key ingredients which allows us to prove the C_n analogue of Prop. 1.1.

Proposition 7.1 *Let $\psi_C : B(\mu) \otimes B(\nu) \longrightarrow \bigoplus_j B(\lambda_j)$ be defined by*

$$\psi_C(b_1 \otimes b_2) = b_1 * b_2$$

where $*$ is defined as in (1.2) but with A_n insertion replaced by C_n insertion. Then ψ_C is the unique crystal isomorphism describing the above tensor product decomposition.

Proof. By induction, it suffices to consider the case where b_2 consists of a single box. Thus one must show for all $1 \leq i, j \leq n$

$$\tilde{f}_j \, \psi_C \left(b \otimes \boxed{i} \right) = \psi_C \, \tilde{f}_j \left(b \otimes \boxed{i} \right) \qquad \tilde{f}_j \, \psi_C \left(b \otimes \boxed{\bar{i}} \right) = \psi_C \, \tilde{f}_j \left(b \otimes \boxed{\bar{i}} \right)$$

and similarly for $\tilde{e}_j$. We consider only the case of $\tilde{f}_j$ with the unbarred box $\boxed{i}$, as the other cases can be proved in a similar manner.

Case $j \neq i-1, i$ From the tensor product rule (1.1) we have

$$\tilde{f}_j(b \otimes \boxed{i}) = \begin{cases} (\tilde{f}_j b) \otimes \boxed{i} & \phi_j(b) > 0 \\ 0 & \phi_j(b) = 0 \end{cases}$$

In the latter case, when $\phi_j(b) = 0$, then certainly $\psi_C \tilde{f}_j(b \otimes \boxed{i}) = 0$. Also (7.2) implies $\tilde{f}_j(b \leftarrow i) = 0$ so $\tilde{f} \psi_C(b \otimes \boxed{i}) = 0$ hence the result is true in this case.

In the former case, when $\phi_j(b) > 0$ we have from Prop. 6.1

$$w(\psi_C \tilde{f}_j(b \otimes \boxed{i})) = w((\tilde{f}_j b) \leftarrow i) \sim w(\tilde{f}_j b) \, . \, i$$

while from (7.1) we have

$$w(\tilde{f}_j \psi_C(b \otimes \boxed{i})) = w(\tilde{f}_j(b \rightarrow i)) \sim \tilde{f}_j(w(b) \, . \, i) = w(\tilde{f}_j b) \, . \, i$$

Since the words of $\psi_C \tilde{f}_j(b \otimes \boxed{i})$ and $\tilde{f}_j \psi_C(b \otimes \boxed{i})$ are Knuth equivalent, and they are both the words of a tableau, they must in fact be the same tableau (see remark below Prop. 6.2). Hence the result is proved in this case.

Case $j = i$ From the tensor product rule (1.1) we have

$$\tilde{f}_i(b \otimes \boxed{i}) = \begin{cases} (\tilde{f}_i b) \otimes \boxed{i} & \phi_i(b) > 0 \\ b \otimes \boxed{i+1} & \phi_i(b) = 0 \end{cases}$$

In the case where $\phi_i(b) = 0$,

$$w(\psi_C \tilde{f}_i(b \otimes \boxed{i}) = w(b \leftarrow (i+1)) \sim w(b) \, . \, i+1$$

while

$$w(\tilde{f}_i \psi_C(b \otimes \boxed{i})) \sim \tilde{f}_i(w(b) \, . \, i) = w(b) \, . \, i + 1$$

where we have successively used (7.1) and the fact that $\phi_i(b) = 0$.

In the case where $\phi_i(b) > 0$, we have similarly $w(\psi_C \tilde{f}_i(b \otimes \boxed{i})) = w((\tilde{f}_i b) \leftarrow i) \sim w(\tilde{f}_i b) \, . \, i$ (using Prop. 6.1) while $w(\tilde{f}_i \psi_C(b \otimes \boxed{i})) \sim \tilde{f}_i(w(b) \, . \, i) = w(\tilde{f}_i b) \, . \, i$ using first (7.1) and then the fact that $\phi_i(b) > 0$. Hence the result is proved in this case.

Case $j = i - 1$ The arguments are similar in this case and will not be repeated, the only thing special to note is that when $\phi_{i-1}(b) \leq 1$, (7.1) implies $\tilde{f}_{i-1}(b \leftarrow i) = 0$. □

8 Proofs

In this section, we give the proofs of the claim that the bumping and sliding procedures defined in Sections 3 and 4 yield valid C_n tableaux. We first remark that the bumping procedure is unambiguously defined and the elements u, v in the type I, IIb and IIa bumps really exist (i.e. $1 \leq u, v \leq n$). For example, in the type I bump there would be a problem if $u+1=1$. But in that case, we would have $\mathrm{pos}(y) + \mathrm{pos}(\overline{y}) \geq y+1$ which is invalid to begin with. A similar remark applies to the IIb bump. For the IIa bump, we must always have $v \leq n-1$ for β to satisfy the conditions for a IIa bump, so there will never be a problem. Similar remarks apply to the reverse bumps.

With regards to the sliding procedures, we can make similar observations. Thus, in the case of the forward type I slide, we would be in trouble if $a < 1$ i.e. $\{i_1, \ldots, i_p\} \sqcup \{j_1, \ldots, j_q\} = \{1, 2, \ldots, x-1\}$. But in that case, noting that the empty box must have slid down vertically from above the entry i_1 (see first claim in section 8.5) Lemma 2.2 applied to the $(i_1, x | \overline{x}, \overline{j_1})$ configuration would yield a contradiction. Similar remarks apply to the remaining forward and reverse slides.

Given that we can carry out one of these bumps or slides, we must check that the result is a valid C_n tableaux. Thus there are 4 conditions we must check: strictly increasing columns, weakly increasing rows, the 1CC and the 2CC. For both bumping and sliding, the first condition is clear by the construction. In the following subsections we check the remaining conditions.

8.1 Bumping and horizontal condition

There are 4 different types of bumps we must consider: ordinary, and types I, IIa and IIb.

For the ordinary bump, the fact that bumping preserves the horizontal condition is well-known.

For the type I bump, because the string of consecutive letters $u+1, u+2, \ldots, y-1$ moves down one position, a possible horizontal violation can only occur if before the bump, there is some v in such a string which has a cell on its left which also contains a v. But the horizontal and vertical conditions then force the cells containing $v+1, \ldots, y$ to have the same entries to their immediate left (see Figure 8.1(a)). This would mean that those two columns are in a (y, y) configuration which contradicts the fact that we began with a valid C_n tableau.

For a type IIa bump, a similar situation occurs. Suppose we are bumping in x, and $\overline{x}$ already appears in the column. Then the only possible horizontal violation can occur when there exists a $\overline{w}$ in the consecutive string $\overline{v-1}, \ldots, \overline{x}$ and again the horizontal and vertical conditions before the bump force the letters $\overline{w}, \overline{w-1}, \ldots, \overline{x}$ to have the same entries in the cells to their immediate left. Now, since we are bumping in an x, such an x must have been bumped out of the previous column by either an ordinary or IIa bump. If it was a ordinary bump, then the situation before this bump must have been as appears in Figure 8.1(b). If it was a type IIa bump, then the a consecutive string of barred letters may have moved during such a bump. By definition, the top of this string $\overline{v}$ is such that $v \leq x \Leftrightarrow \overline{x} \geq \overline{v}$. Thus the consecutive string $\overline{w}, \ldots, \overline{x+1}, \overline{x}$ does not move and the situation is also as appears in Figure 8.1(b). Thus, before the bump, there was an (x, x) configuration which is a contradiction. The argument for the type IIb bump runs along similar lines.

8.2 Bumping and one column condition

For an ordinary bump, a 1CC violation can occur in two possible cases: (i) a bumps an x and there exists an $\overline{a}$ such that $\mathrm{pos}(x) + \mathrm{pos}(\overline{a}) > a$ (ii) $\overline{a}$ bumps an $\overline{x}$ and there exists an a such that $\mathrm{pos}(a) + \mathrm{pos}(\overline{x}) > a$. In case (i), this cannot happen as such a bump must be, by definition, a type II bump. In case (ii), we must have $a \geq x$ and so by Lemma 2.1 $\mathrm{pos}(a) + \mathrm{pos}(\overline{x}) \leq \max(a, x) = a$.

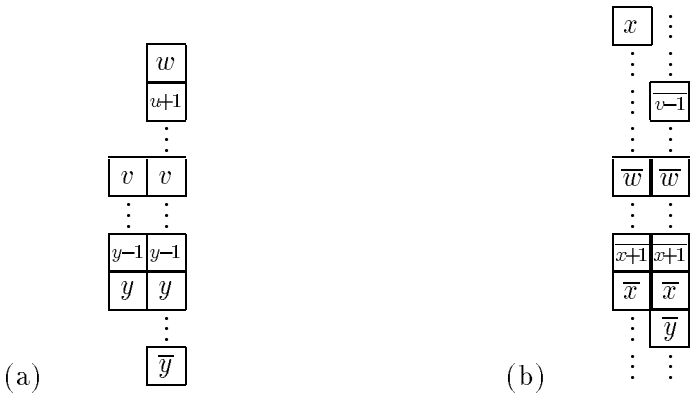


Figure 8.1: Bumping and the horizontal condition

We must also examine the case where the element inserted into a column is strictly greater than all the elements in the column, in which case it gets placed at the end of that column (this clearly only occurs in the case of an ordinary bump). We claim that, *apart from the case when we are inserting into the first column*, such a situation never gives rise to a 1CC violation. In the case of a 1CC violation in the first column, this signals that we must delete some boxes and start sliding (recall section 5).

For a type I bump, we must check the 1CC for any s-$\overline{s}$ pair where $s \in \{u+1, u+2, \ldots, y-1\}$, for a possible u-$\overline{u}$ pair, where u is a new element appearing at the top of the string, and for any x-$\overline{x}$ pair in the case where $\alpha = \overline{x}$. Let B (resp. A) be the column as it appears before (resp. after) the bump. We shall use this convention from now on. In the first case, we have by the definition of the 1CC

$$\mathrm{pos}_B(s) + \mathrm{pos}_B(\overline{y}) = \mathrm{pos}_B(y) + s - y + \mathrm{pos}_B(\overline{y}) \leq s \tag{8.1}$$

Since $s \leq y-1$, we must have $\mathrm{pos}_B(\overline{s}) \leq \mathrm{pos}_B(\overline{y}) - 1$, and hence from (8.1) $\mathrm{pos}_B(s) + \mathrm{pos}_B(\overline{s}) \leq s - 1$ and thus $\mathrm{pos}_A(s) + \mathrm{pos}_A(\overline{s}) \leq s$. In the second case, we know $\mathrm{pos}_A(u) = \mathrm{pos}_B(u+1)$ and $\mathrm{pos}_A(\overline{u}) = \mathrm{pos}_B(\overline{u}) < \mathrm{pos}_B(\overline{y})$. Thus

$$\mathrm{pos}_A(u) + \mathrm{pos}_A(\overline{u}) < \mathrm{pos}_B(u+1) + \mathrm{pos}_B(\overline{y}) \leq u+1$$

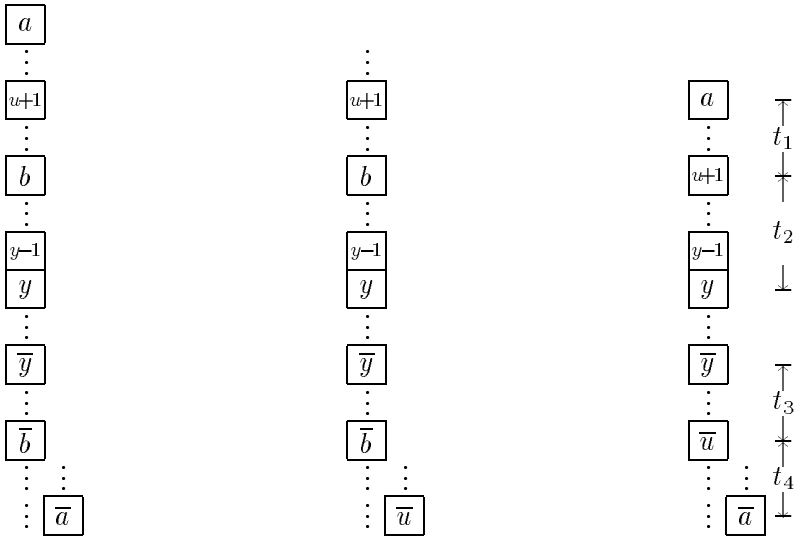
where, in the last inequality, we have used (8.1). Hence $\mathrm{pos}_A(u) + \mathrm{pos}_A(\overline{u}) \leq u$. Finally, if $\alpha = \overline{x}$ and there exists an $x \in A$, then since $x \geq y$, Lemma 2.1 implies $\mathrm{pos}_B(x) + \mathrm{pos}_B(\overline{y}) \leq x$ and so $\mathrm{pos}_A(x) + \mathrm{pos}_A(\overline{x}) \leq x$.

For a type IIa bump, the only letters which move downwards are barred letters, so the 1CC will always remain valid for such letters. However, we must check the 1CC for the new v-$\overline{v}$ pair which appears. Suppose that δ is the smallest letter $\geq v$ in the column B, which gets replaced during the bump by a v. Then either $\mathrm{pos}_B(\delta) = 1$ or there exists a $w < v$ in the cell immediately above δ. In the former case $\mathrm{pos}_B(\delta) + \mathrm{pos}_B(\overline{v-1}) \leq 1 + v - 1 = v$ and thus $\mathrm{pos}_A(v) + \mathrm{pos}_A(\overline{v}) \leq v$. In the latter case Lemma 2.1 implies $\mathrm{pos}_B(w) + \mathrm{pos}_B(\overline{v-1}) \leq v-1$. Thus $\mathrm{pos}_A(v) + \mathrm{pos}_A(\overline{v}) \leq v$ as required.

For a type IIb bump, first note that since $\mathrm{pos}_B(v-1) + \mathrm{pos}_B(\overline{v-1}) \leq v-1$, for any other pair e-$\overline{e}$ in the string $u+1, \ldots, v-1, \overline{v-1}, \ldots, \overline{x}$ we have

$$\mathrm{pos}_B(e) + \mathrm{pos}_B(\overline{e}) \leq e - (v - 1 - e) \tag{8.2}$$

We consider the two cases when $u \geq x$ and $u < x$. In the former case, consider all pairs s-$\overline{s}$ with $u < s \leq v-1$. Since the entire string $u+1, \ldots, v-1, \overline{v-1}, \ldots, \overline{x}$ moves down, the net effect on $\mathrm{pos}(s) + \mathrm{pos}(\overline{s})$ is zero. Similarly, we must check the 1CC for the newly created $u - -\overline{u}$ pair. There, Lemma 2.1 implies that $\mathrm{pos}_B(u+1) + \mathrm{pos}_B(\overline{u}) \leq u+1$, hence the result follows from noting that $\mathrm{pos}_B(u+1) = \mathrm{pos}_A(u)$ and $\mathrm{pos}_B(\overline{u}) = \mathrm{pos}_A(\overline{u}) + 1$.



(a) (b) (c)

Figure 8.2: I bumps and 2CC violations

In the latter case, there may be a pair t-$\overline{t}$ with $u < t < x$ so that $\mathrm{pos}(t)$ increases by one and $\mathrm{pos}(\overline{t})$ remains unchanged. But (8.2) implies that

$$\mathrm{pos}_B(t) + \mathrm{pos}_B(\overline{t}) < \mathrm{pos}_B(x) + t - x + \mathrm{pos}_B(\overline{x}) \leq t + x - (v-1) < t$$

which gives the required result. We must also check the 1CC if there is an entry $\overline{u}$ below $\overline{x}$. By definition, we have $\mathrm{pos}_B(x) + \mathrm{pos}_B(\overline{x}) \leq x$ and so $\mathrm{pos}_B(u+1) + \mathrm{pos}_B(\overline{u}) \leq x - (x-u-1) = u+1$ and the result now follows.

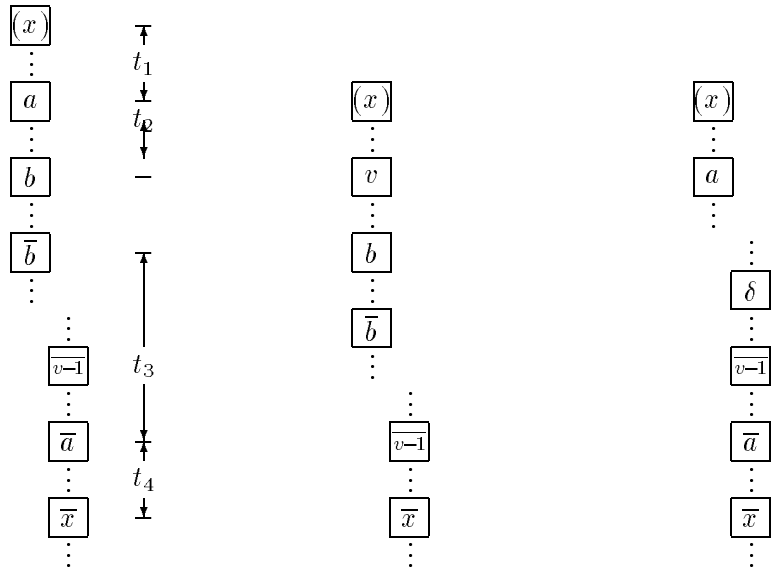
8.3 Bumping and two column condition

First consider the case of an ordinary bump. From Lemma 2.2, the only possible 2CC violations could occur when we bump into the second or fourth entry of an $(a, \delta|\overline{b}, \overline{a})$ or $(a, b|\overline{b}, \overline{x})$ configuration respectively. To see why this is so, suppose for example that a bumps an x in an $(x, b|\overline{b}, \overline{a})$ configuration. Then by definition $a \leq x$ and Lemma 2.2 implies that $p(x, b|\overline{b}, \overline{a}) < b - \min(x, a) = b - a$ and so the 2CC will always hold in such a case. Similarly for the case when $\overline{b}$ gets bumped into the third position of an $(a, b|\overline{x}, \overline{a})$ configuration.

In the case when b bumps an x in a (left or right) $(a, x|\overline{b}, \overline{a})$ configuration this is by definition a type II bump, so we don't consider that here.

In the case where $\overline{a}$ bumps an $\overline{x}$ in an $(a, b|\overline{b}, \overline{x})$ configuration, then (calling the left column C and the right column C') the entry $\overline{a}$ being inserted into the column C' must have resulted from a type I or IIb bump in column C (but *neither* from an ordinary bump, *nor* a IIa bump). Before such bumps, there must have been a string of consecutive letters $a+1, a+2, \ldots, c$ and a letter $\overline{c}$ in column C, Moreover, the letter in the cell above $\overline{x}$ must have been a barred letter, $\overline{y}$ say, so that $\overline{y} < \overline{a} < \overline{x}$. Thus for the $(a+1, b|\overline{b}, \overline{y})$ configuration, Lemma 2.2 implies $p(a+1, b|\overline{b}, \overline{y}) \leq b - a - 1$. Hence $p(a+1, b|\overline{b}, \overline{x}) \leq b - a - 1$ and since a replaces $a+1$ after such bumps, the 2CC holds (note that in the case of a left configuration, b (and possibly $\overline{b}$) will move up one position, so the result will still hold).

For a type I bump, the only possible 2CC violations can come from the following 3 cases (see Figure 8.2):



(a) (b) (c)

Figure 8.3: IIa bumps and 2CC violations

(a) the entry b in a (left or right) $(a, b|\overline{b}, \overline{a})$ configuration belongs to the consecutive string which moves down one position in the type I bump (and a does not).

(b) there is a $(u+1, b|\overline{b}, \overline{u})$ configuration where b belongs to the consecutive string below $u+1$ and u gets created at the top of the string.

(c) there is an $(a, u+1|\overline{u}, \overline{a})$ configuration and u gets created at the top of the string $u+1, \ldots, y$.

The proofs that such 2CC violations can never occur are similar, so we give only one of them, case (c). In that case, first note that Lemma 2.2 implies that $p(a, u+1|\overline{u}, \overline{a}) \leq u - a$ so a problem can only occur if equality holds there. Let us assume therefore that $t_1 + t_4 = u - a$. Since $u+1$ and y belong to a string of consecutive integers, $t_2 = y - u - 1$. Also, from the validity of the (a, y) configuration, we have $t_1 + t_2 + t_3 + t_4 \leq y - a - 1$. All of these facts imply $t_3 \leq 0$ which is impossible.

Note that in cases (a) and (c), we must also consider the cases of right configurations, but the arguments are identical to the cases of left configurations. Also, the letter α which replaces the entry $\overline{y}$ as part of the type I bump might also cause a 2CC violation, but the reason that it doesn't is the same as the reasons given above for the validity of the ordinary bump.

For a type IIa bump there are also 3 situations which might result in a 2CC violation:

(a) the entry $\overline{a}$ in an (a, b) configuration moves down one position as part of a consecutive string.

(b) there is an $(v, b|\overline{b}, \overline{v-1})$ configuration and $\overline{v}$ appears at the top of the consecutive string $\overline{v-1}, \ldots, \overline{x+1}, \overline{x}$.

(c) there is an $(a, \delta|\overline{v-1}, \overline{a})$ where $\overline{v-1}$ is at the top of a consecutive string, $\overline{a}$ appears in the string, and δ is bumped by the v which is created.

See Figure 8.3. Since the proofs are all similar, we prove only (a).

First note that since x is being bumped into a IIa situation, it must have been bumped from a previous column, either from an ordinary or a type IIa bump. Let us assume first of all, that it came from an ordinary bump. Figure 8.3(a) shows this pre-bump situation, where x is

still in the left-hand column. Now, by the definition of an (a,b) configuration, we must have $p(a,b|\overline{b},\overline{a}) \leq b-a-1$. Indeed, we can assume equality holds, otherwise we will never have a problem. Thus $t_2+t_3 = b-a-1$. Also, since $\overline{a}$ is part of a consecutive string, $t_4 = a-x$. Finally the valid $(x,b|\overline{b},\overline{x})$ configuration tells us that $t_1+t_2+t_3+t_4 \leq b-x-1$. Hence $t_1 \leq 0$ which is not possible. If, on the other hand, x came from a type IIa bump, then either $\overline{b}$ was part of a consecutive string which moved down one cell (in which case no 2CC violation could have resulted anyway) or it wasn't, in which case the above argument also holds.

Finally we must consider type IIb bumps. In this case, there are 4 situation in which a 2CC violation might arise:

(a) in an (a,b) configuration, the entry b moves down one cell (but all other entries do not).

(b) in an (a,b) configuration, the entry $\overline{a}$ moves down one cell (but all other entries do not).

(c) there is an $(a+1,b|\overline{b},\overline{a})$ configuration with $a+1$ being at the top of the consecutive string, and b part of the consecutive string (but all other entries are not).

(d) there is an $(a,b+1|\overline{b},\overline{a})$ configuration with $b+1$ being at the top of the consecutive string, and no other entries are part of it.

The proofs that such situations lead to a contradiction is similar to the type IIa case and will not be given.

8.4 Ordinary sliding

In this subsection, we show that ordinary slides can never lead to a violation of the 4 conditions of C_n tableaux. For the weakly increasing row conditions and the strictly increasing column conditions, this is a classical fact [5].

One column condition

There are 3 possibilities where an ordinary slide might cause a violation of the 1CC:

(i) the entry $\overline{a}$ slides left one position and creates an a–$\overline{a}$ pair whose positions violate the 1CC.

(ii) the entry a slides left one position and creates an a–$\overline{a}$ pair whose positions violate the 1CC.

(iii) the entry $\overline{a}$ of an a–$\overline{a}$ pair slides up one position.

Case (i) This cannot happen as such a slide must be a type II special slide, not an ordinary slide.

Case (ii) We must consider how the empty box arrived in the column containing $\overline{a}$. If it came from an ordinary horizontal slide or a type I slide, then the entry $\overline{a}$ doesn't move during such a slide. Thus, if x occupied the cell to the left of a before such a slide then $x \leq a$ and Lemma 2.1 implies that $\mathrm{pos}(x)+\mathrm{pos}(\overline{a}) \leq a$ so there can never be a 1CC violation in such a case. On the other hand, if the empty box came from a type III slide, then $\overline{a}$ may or may not move up one position during such a slide. If it doesn't move, the previous argument applies. If it does move, then before the type III slide, the entry, z say, to the left of a also moves down one position, so we must have $z \leq a$. But when the emptry box slides down to the left of a, z is below the empty box, and a horizontal slide then takes place, so we must have $z > a$, which is a contradiction.

Case (iii)

First note that the only way for a 1CC violation to occur, is if the path of the box is as appears in Figure 8.4(a) (if the empty box continues down vertically below $\overline{a}$ and dissappears from the end of the column, no 1CC violation can ever occur). Lemma 2.1 tells us that in Figure

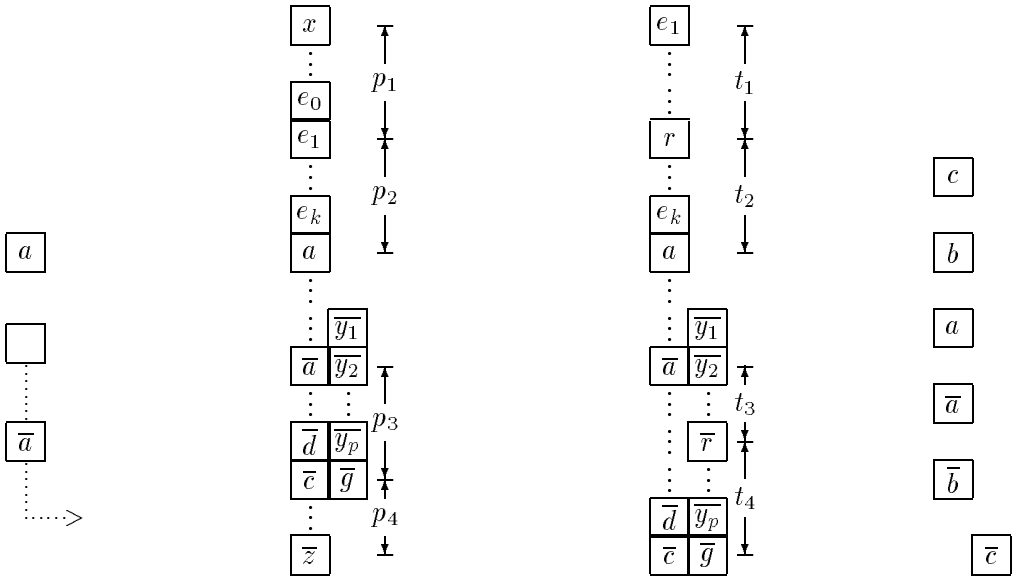


Figure 8.4: 1CC violation and ordinary vertical slides

8.4(a) $\text{pos}(a) + \text{pos}(\overline{a}) \leq a$. Moreover, the empty box must leave the column horizontally via either an ordinary, type II or IV slide (note that for a II or IV slide, it is possible that a or $\overline{a}$ might move, but this will only decrease $\text{pos}(\cdot)$, so for any violation to occur we assume they don't move).

Consider first the case when the box leaves the column via an ordinary or type II slide. In this case we will show that the a–$\overline{a}$ pair is part of some left (r,a) configuration and that when the entry $\overline{a}$ moves up one position during the journey of the empty box, a type IV special slide must occur, contrary to assumption. First note that a possible 1CC violation can only occur if $\text{pos}(a) + \text{pos}(\overline{a}) = a$. So let us assume such a relation holds. Referring to Figure 8.4(b), let us assume that the empty box moves out of the column at the position occupied by $\overline{d}$, so that $y_p > c$ (this must be true for both an ordinary slide and a type II slide). Note that since the entry $\overline{a}$ has to move vertically, we must have $\overline{a} \leq \overline{y_1} \Leftrightarrow y_1 \leq a$. In fact, since we are assuming such a slide is an ordinary slide, we must have $y_1 < a$ (otherwise a type IV slide will occur). Also, let e_1 be the smallest letter $> c$ lying above a (we include the possibility that $e_1 = a$).

We now claim that $\{e_1, \ldots, e_k\} \cap \{y_1, \ldots, y_p\} \neq \emptyset$. For a contradiction, suppose this is not so. Then since both of these sets are subsets of $\{c+1, c+2, \ldots, a\}$, we must have $p_2 + p_3 + 1 \leq a - c - 1$. Let us now assume that there exists an entry e_0 $(\leq c)$ lying immediately above e_1. Then Lemma 2.1 applied to the entries e_0 and c implies

$$p_1 + p_4 + 1 \leq c. \tag{8.3}$$

If such an e_0 doesn't exist, then $\text{pos}_B(e_1) = 1$ (i.e. $p_1 = 0$) and since $\text{pos}_B(\overline{c}) \leq c$, (8.3) still holds. Together these imply that $p_1 + p_2 + p_3 + p_4 + 2 \leq a - 1$, contrary to the assumption that $\text{pos}(a) + \text{pos}(\overline{a}) = p_1 + p_2 + p_3 + p_4 + 2 = a$.

Given that this intersection is nonempty, choose the smallest letter in such an intersection, call it r say (see Figure 8.4(c)). From the diagram, we have $p_2 = t_1 + t_2$, $p_3 = t_3 + t_4$. The fact that r is minimal implies $t_1 + t_4 - 1 \leq r - c - 1$. Also, (8.3) implies $p_2 + p_3 \geq a - c - 1$. Thus

$$t_2 + t_3 = (p_2 + p_3) - (t_1 + t_4) \geq a - c - 1 - (r - c) = a - r - 1.$$

But from the definition of the (r,a) configuration, we know that $t_2 + t_3 \leq a - r - 1$. Hence

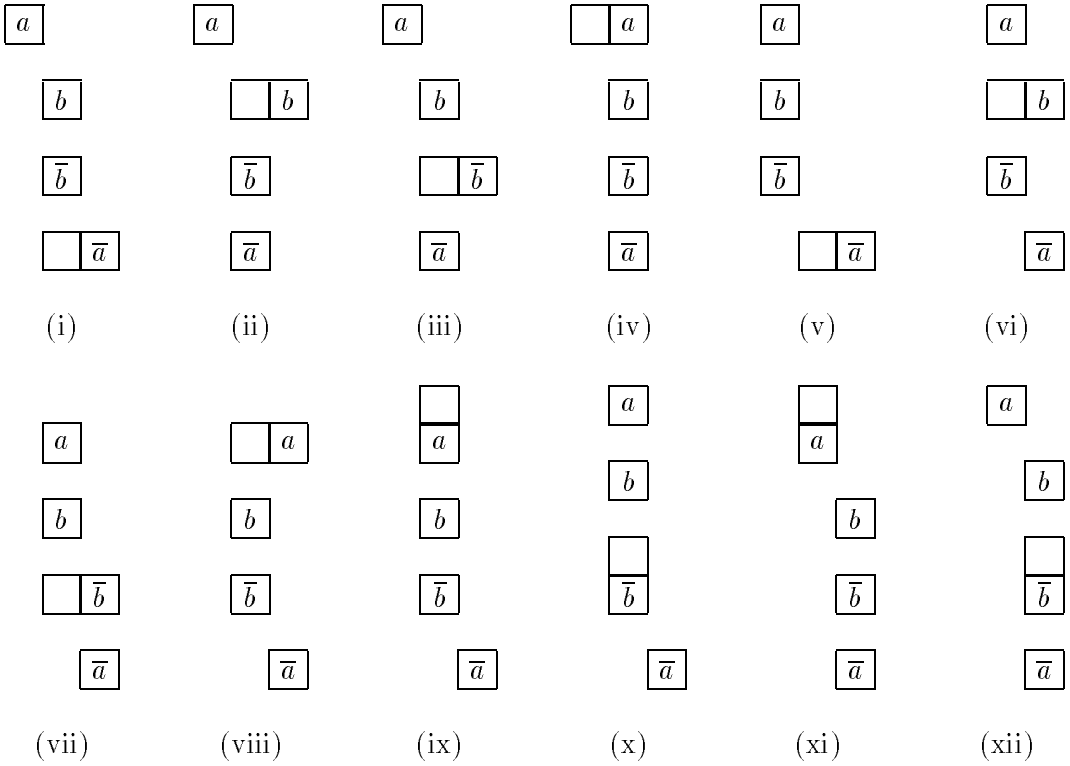


Figure 8.5: Ordinary slides and the 2CC

$t_2 + t_3 = a - r - 1$ and so when the entry $\overline{a}$ slides up, a type IV special slide must occur, as promised.

Finally, suppose the empty box moves out of the column via a type IV slide. Then we have the situation as it appears in Figure 8.4(d) with $p(c, b|\overline{b}, \overline{c}) = b - c - 1$, $\mathrm{pos}(a) + \mathrm{pos}(\overline{a}) = a$. Then

$$p(c, a|\overline{a}, \overline{c}) = p(c, b|\overline{b}, \overline{c}) + \mathrm{pos}(a) - \mathrm{pos}(b) + \mathrm{pos}(\overline{a}) - \mathrm{pos}(\overline{b}) \geq a - c - 1$$

However, by definition $p(c, a|\overline{a}, \overline{c}) \leq a - c - 1$ and so we must have $p(c, a|\overline{a}, \overline{c}) = a - c - 1$. Thus, when $\overline{a}$ moves up vertically, a type IV slide must occur, contrary to assumption.

Two column condition

Turning to ordinary slides which might invalidate the 2CC, we divide them into 2 categories: those while occur due to an ordinary horizontal slide and those due to an ordinary vertical slide. There are 8 instances of the former, and 4 of the latter, pictured schematically in Figure 8.5. Let us consider each of these cases in turn, showing that such violations never arise.

Case (i)

Assume for a contradiction that $p(a, b|\overline{b}, \square) \geq b - a$. Let c denote the entry to the immediate right of the entry a. Then we have $c \geq a + 1$ (certainly $c \geq a$ due to the row condition, but if $c = a$, we would have an illegal left (a, b) configuration). Let $\overline{e}$ be the entry in the cell immediately above $\overline{a}$ (note that this must indeed be barred). Certainly $e > a$. Applying Lemma 2.2 to the $(c, b|\overline{b}, \overline{e})$ configuration, we have $p(c, b|\overline{b}, \overline{e}) < b - a - 1$ and so $p(a, b|\overline{b}, \square) = p(c, b|\overline{b}, \overline{e}) + 1 < b - a$ which is a contradiction.

Case (ii)

We must analyse how the empty box arrived at it's position in the cell immediately to the left of b. Let us call the column to the left of a, the column containing a, and the column containing

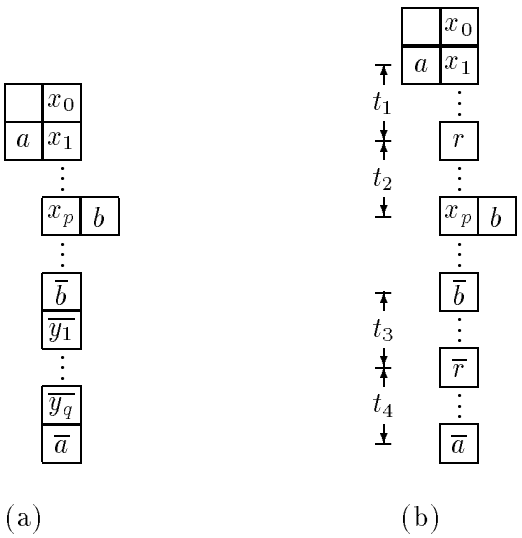


Figure 8.6: Case (ii) of the 2CC violations

$\overline{b}$, $\overline{a}$ C_0, C_1 and C_2 respectively. First note that if the box entered C_2 from (weakly) above a (via an ordinary, type I or III slide) then applying Lemma 2.2 immediately after such a slide to the $(a, e|\overline{b}, \overline{a})$ configuration (where e is the entry immediately to the left of b) shows that $p(a, e|\overline{b}, \overline{a}) < b - a$ and so a 2CC violation will never arise. Thus we only consider the case where the box enters below the row containing a, and above the row containing b. We consider 2 subcases

(a) the box enters C_2 via an ordinary or type I slide

(a) the box enters C_2 via a type III slide

In subcase (a), note that if the box enters via a type I slide, then either the a–$\overline{a}$ pair in columns C_1, C_2 were created during the slide (in which case the entries in those cells were both $> a$ and an application of Lemma 2.2 shows we have no problem) or one or both of a , $\overline{a}$ moved down/up respectively (and again, application of Lemma 2.2 shows there can never be a violation) or nothing moved. Thus we can assume the latter.

We next claim that the empty box must have entered column C_1 from C_0 from *above* a (and hence a must have slid up through an ordinary vertical slide). If this is not the case, then it either entered via an ordinary, type I or type III slide and in all cases a did not move, so an application of Lemma 2.2 to the $(a, e|\overline{b}, \overline{a})$ configuration show that there won't be a 2CC violation.

Given that the box passes through a vertically, assume the situation appears as in Figure 8.6(a). Note that since a slides vertically, we must have $x_0 > a$ (certainly $x_0 \geq a$ but if $x_0 = a$ a type III slide would occur), and that $x_p < b$ (certainly $x_p \leq b$, but if $x_p = b$ then the empty box would slide out of C_2 in the row *above* b). For a 2CC violation to occur, we assume $p(a, x_p|\overline{b}, \overline{a}) = b - a - 1$ i.e. $(p-1)+(q+1) = b-a-1$. We claim that $\{x_0, x_1, \ldots, x_p\} \cap \{y_1, \ldots, y_q\} \neq \emptyset$. Indeed, if the intersection were empty, then since both sets are subsets of $\{a+1, a+2, \ldots, b-1\}$ we would have $p+1+q \leq b-a-1$, in contradiction to the above assumption. Let r be the maximal element of such an intersection. Refering to Figure 8.6(b), it follows from the maximality of r that $t_2 + t_3 - 1 \leq b - r - 1$. Thus

$$t_1 + t_4 = p(a, x_p|\overline{b}, \overline{a}) - (t_2 + t_3) \geq r - a - 1$$

But by definition of the (a,r) configuration we must have $t_1+t_4 \leq r-a-1$. Hence $t_1+t_4 = r-a-1$ and so when the entry a slides up we must have a type III slide, not an ordinary vertical slide, contrary to assumption.

Finally, turning to subcase (b), there are several things which may occur when the box slides from column C_1 to column C_2 via a type III slide: the diagonal pair a–$\overline{a}$ might have been created, or a (resp $\overline{a}$) might have moved down (resp. up) or both may have moved. Also, $\overline{b}$ and the entry to the left of b might move up or down respectively. Analysis of all of these situations with Lemma 2.2 shows that strife can only occur when nothing is created and nothing moves - in this case, we can repeat the analysis of subcase (a) and show that the entry a must have moved down vertically as part of some (a,r) configuration, giving a type III slide there, contrary to assumption.

Case (iii)

This is a type II slide, contrary to assumption.

Case (iv)

This is a type I slide, contrary to assumption.

Case (v)

This is similar to case (ii), except with type II and IV slides taking the place of I and III slides. The arguments will not be repeated.

Case (vi)

This is similar to case (i) - let $\overline{c}$ be the entry immediately to the right of $\overline{b}$ in the diagram. Then $c < b$ and the entry immediately above b, e say, must satisfy $e < b$ also. Applying Lemma 2.2 to this $(a, e|\overline{c}, \overline{a})$ configuration gives the required result.

Case (vii)

This is a type II slide, contrary to assumption.

Case (viii)

This is a type I slide, contrary to assumption.

Case (ix)

The argument here is similar to that presented for the vertical slides which might violate the 1CC. Namely, one can show that in the only situation which might cause a problem, there exists some r such that the entries a and $\overline{a}$ are part of some right (a,r) configuration and that moving a up one position, will cause a type III slide, not an ordinary vertical slide contrary to assumption. The details are omitted.

Case (x)

This is a type IV slide.

Case (xi)

This is a type III slide.

Case (xii)

A similar argument to case (ix).

8.5 Type I, II Sliding

Let us begin with the row conditions which must be satisfied by the tableau after a type I slide. We recall the situation in Figure 8.7(a). It is clear that before the slide

$$t_k \leq i_k \leq u_k, \qquad k = 1, \ldots, p \qquad\qquad v_k \geq j_k \geq w_k \qquad k = 1, \ldots, q$$

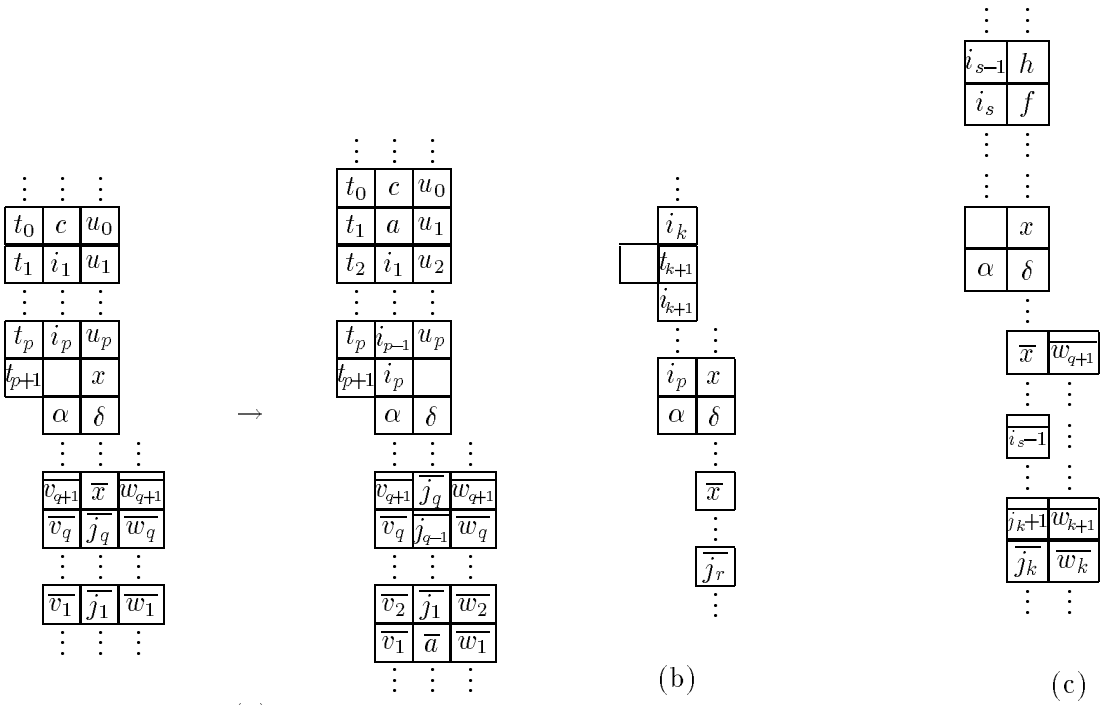


Figure 8.7: Type I slide

In particular, $a < i_1 \leq u_1$ and $i_k < u_{k+1}$ for $1 \leq k < p$. Similarly $v_1 \geq j_1 > a$ and $v_{k+1} > j_k$ for $1 < k \leq q$. It thus remains to check that $t_1 \leq a$, $t_{k+1} \leq i_k$ for $1 \leq k \leq p$ and $a \geq w_1$, $j_k \geq w_{k+1}$ for $1 \leq k \leq q$. First, we make the following

Claim: The empty box has slid vertically from above i_1 by a sequence of ordinary vertical slides.

Proof. Suppose not, and that at some stage, the box is located between the entries i_k and i_{k+1}, having entered that position from the column on the left. Then the following are the only scenarios possible:

(a) The box came from an ordinary horizontal slide and the situation is as it appears in Figure 8.7(b).

(b) The box came from a type I slide and so there was an entry p as part of a p–$\overline{p}$ pair which was located between the entries i_k and i_{k+1} i.e. $i_k < p < i_{k+1}$.

(c) The box came from a type III slide.

In case (a), since $\{i_1, \ldots, i_p\} \sqcup \{j_1, \ldots, j_q\} = \{a+1, a+2, \ldots, x-1\}$, there exists some r such that $j_r = t_{k+1}$. Thus $\{i_{k+1}, \ldots, i_p\} \sqcup \{j_{r+1}, \ldots, j_q\} = \{t_{k+1}+1, t_{k+1}+2, \ldots, x-1\}$. Moreover, from the diagram $p(t_{k+1}, x|\overline{x}, \overline{j_r}) = (p-k) + (q+1-r) = x - t_{k+1}$. But by definition of the 2CC, we must have $p(t_{k+1}, x|\overline{x}, \overline{j_r}) < x - t_{k+1}$, so we have a contradiction.

In case (b), a similar argument holds, since the entries $\{i_1, \ldots, i_p\}$ and $\{\overline{j_1}, \ldots, \overline{j_q}\}$ don't move in that particular slide.

In case (c), there was an b, part of an (a, b) configuration which disappears as part of the type III slide and we must consider 2 subcases as to where the entry b lied before the III slide. If b was located amongst the string $i_{k+1}, \ldots, i_p, \alpha$ then a similar argument to cases (a) and (b) shows there was a r such that $j_r = b$ which was in an invalid (b, x) configuration. Alternatively, b was located below $(\alpha =)y$, in which case y must have been in the cell immediately to the left of x, which is also a contradiction as we are assuming that $x < y (= \alpha)$. □

It follows from the above claim that $t_{k+1} \leq i_k$ for all $k = 1, \ldots, p$. In addition, we must show that $t_1 \leq a$. To see this, note that when the empty box was at the top of the column of $i_1, \ldots, i_p$, then the previous move must have been an (ordinary) vertical slide or a horizontal slide (either

ordinary, type I or III). In the first case, since $t_1 \leq c \leq a-1$, we have $t_1 < a$ as required. If the slide was an ordinary horizontal slide, then t_1 must have been located between c and i_1 before such a slide. If $t_1 > a$ there, then an argument similar to that in the claim above yields a contradiction. If the slide was a type I or III slide, then t_1 had moved down one position during the slide. i.e. before it was immediately to the left of c, so that $t_1 \leq c < a$ and we are done.

It remains to check the inequalities $j_k \geq w_{k+1}$, $1 \leq k \leq q$ and $a \geq w_1$. For this refer to Figure 8.7(c), where the entry i_s is the smallest integer $> j_k$ lying above a. Certainly the entries bewteen i_s and the empty box and between $\overline{j_k}$ and $\overline{x}$ must form the disjoint union of the set $\overline{j_k+1}, \overline{j_k+2}, \ldots, \overline{x-1}$. Thus

$$p(f, x|\overline{x}, \overline{w}_{k+1}) = x - j_k - 1. \tag{8.4}$$

But Lemma 2.2 tells us that $p(f, x|\overline{x}, \overline{w_{k+1}}) \leq x-1-\min(f, w_{k+1})$. Thus we have $\min(f, w_{k+1}) \leq j_k$. But $f \geq i_s \geq j_k + 1$ and so we can conclude that $w_{k+1} \leq j_k$. A similar argument shows that $a \geq w_1$.

Turning to the 1CC, let us first consider the column containing the i's. If any $i_r = p$ was part of a p–$\overline{p}$ pair, then the above claim implies that the 1CC will still hold after the type I slide. Also, if there exists an entry $\overline{a}$ in the barred part of that column, then the above arguments show that before the empty box entered the column above the i's, the entry which occupied the cell now occupied by a was always $< a$, so Lemma 2.1 tells us that the 1CC will hold for the new a–$\overline{a}$ pair.

Similarly, let us consider the column containing the j's. Suppose there is an i and k such that $u_i = j_k$ say. Vertical constraints imply that such a u_i must occur in a row above that containing i_s in Figure 8.7(c). That is, $\mathrm{pos}(u_i) \leq \mathrm{pos}(h)$. Thus using (8.4), we have

$$\begin{aligned} \mathrm{pos}(u_i) + \mathrm{pos}(\overline{j_k}) \leq \mathrm{pos}(h) + \mathrm{pos}(\overline{j_k}) &= \mathrm{pos}(x) + \mathrm{pos}(\overline{x}) - p(f, x|\overline{x}, \overline{w_{k+1}}) - 2 \\ &\leq x - (x - j_k - 1) - 2 = j_k - 1 \end{aligned} \tag{8.5}$$

and hence the 1CC will hold when $\overline{j_k}$ moves up one position. We must also check the case when there already exists an entry a in the unbarred section of the column, as the creation of $\overline{a}$ may produce an a–$\overline{a}$ pair which fails the 1CC. The argument here is the same as in the case of the column containing the i's.

Finally, we examine the 2CC. There are 3 possibilities for a 2CC violation to occur:

(a) there exist i, k such that $u_i = j_k$ which is part of a (e, u_i) configuration, where $e < a$.

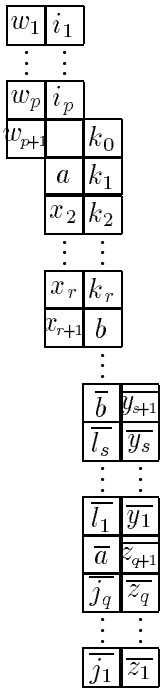
(b) there exist r, s such that $i_r = v_s$ which is part of a (e, i_r) configuration, where $e < a$.

(c) there exist r, s (resp. i, k) such that $i_r = v_s$ (resp. $u_i = j_k$) which occur as part of an (a, i_r) (resp. (a, u_i)) configuration.

In case (a), the argument is similar to the 1CC case described above. From the definition of the 2CC, we know $p(e, x|\overline{x}, \overline{e}) < x - e$, and so it follows from this and (8.5) that $p(e, u_i|\overline{j_k}, \overline{e}) < j_k - e - 1$. Thus when j_s moves up one position during the type I slide, the 2CC will remain valid. The argument for case (b) is similar. In case (c), the argument is even simpler: since both i_1 and j_1 are $\geq a+1$ Lemma 2.2 implies that for any $i_r = v_s$ for example, $p(i_1, i_r|\overline{v_s}, \overline{j_1}) < i_r - (a+1)$ and so moving i_r down by one during the slide will not invalidate the 2CC.

To close this subsection, we remark that the proof of the validity of the type II slide follows a similar analysis as the type I slide and will not be repeated.

8.6 Type III, IV Sliding



Let us consider the case of a type III slide. We must check that the 4 conditions for symplectic tableaux are preserved. Again, the vertical conditions clearly hold. For the horizontal conditions, consider the diagram on the left. During the type III slide, the entries $i_1,\ldots,i_p,k_0,\ldots,k_r$ move down one position so we need only concern ourselves with the entries to their immediate left, which could give rise to a violation of the horizontal condition. Similarly, the entries $\overline{j_1},\ldots,\overline{j_q},\overline{a},\overline{l_1},\ldots,\overline{l_s}$ move up one position, so we only need to look at the entries to their immediate right.

To prove that $w_{m+1} \leq i_m$, for $1 \leq m \leq p$, we argue as in the previous section, that the empty box has slid vertically down from above i_1. The reasoning is almost identical: for example, if it enters between i_k and i_{k+1} via an ordinary slide, one can show there is some u lying above a, and $\overline{u}$ lying below $\overline{a}$ such that the union of the entries between u and a and between $\overline{u}$ and $\overline{a}$ is disjoint and equal to $\{u+1,u+2,\ldots,a-1\}$. This implies that $p(u,b|\overline{b},\overline{u}) = p(a,b|\overline{b},\overline{a})+a-u+1 = b-u$, using the fact that $p(a,b|\overline{b},\overline{a}) = b-a-1$. This is clearly impossible. The arguments for entering via a type I or III slide are identical to the previous section.

Similarly, we must prove $x_{m+1} \leq k_m$, for $1 \leq m \leq r$ (certainly $a \leq k_0$ as part of the conditions for a type III slide). The crucial point here is that since $p(a,b|\overline{b},\overline{a}) = b-a-1$, and b is the smallest integer satisfying that condition, we must have $\{k_0,k_1,\ldots,k_r\} \sqcup \{l_1,\ldots,l_s\} = \{a+1,a+2,\ldots,b-1\}$. In other words, you can consider the sequence $\overline{l_1},\ldots,\overline{l_s}$ to be the consecutive sequence $a+1,a+2,\ldots,b-1$ interrupted by the absences of $k_0,\ldots,k_r$. In this setting, consider the entries x_{m+1} and $\overline{k_m+1}$. It follows that

$$p(x_{m+1},b|\overline{b},\overline{k_m+1}) = (r-m)+(b-(k_m+1)-(r-m)) = b-k_m-1$$

However, Lemma 2.2 implies $p(x_{m+1},b|\overline{b},\overline{k_m+1}) \leq b-1-\min(x_{m+1},k_m+1)$. Hence we must have $x_{m+1} \leq k_m$ as promised.

A similar argument to that presented in the above paragraph is used to show that $l_m \geq y_{m+1}$ for $1 \leq m \leq s$, while to show $j_m \geq z_{m+1}$ for $1 \leq m \leq q$, we can use a similar argument as was used in the previous section (used to show $j_k \geq w_{k+1}$ in Figure 8.7(a)).

Turning to the 1CC, the fact that this continues to hold in the column containing the i's follows the same argument as used in the previous section. For the column containing the j's, note that none of the k's can ever be part of some p–$\overline{p}$ pair, likewise with the l's. It remains to check the j's themselves, and here we can follow the same argument as in the previous section (for the type I slide).

For the 2CC, the arguments are similar to the previous section, and will not be repeated. Finally, we remark that the case of type IV slides can be analysed in a similar manner.

8.7 Reverse slides

In this subsection, we will describe in what manner the reverse slides described in Section 4.2 really are "reverse". Indeed, we shall see that if one performs a sequence of forward slides on an empty box in a tableau, and then performs a sequence of reverse slides on the result, the paths taken by the empty box may not be the same. However the paths, and the corresponding

tableaux, coincide at several points, in particular the initial point (after the bumping–sliding transition) and the final point (when the empty box moves to the edge of the tableau).

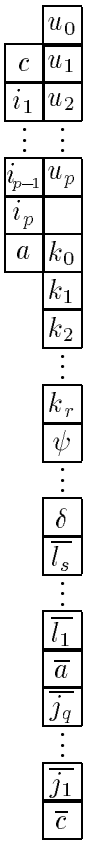
Consider first the case of the type I forward slide (refer to Figure 8.7(a)). There are two possible scenarios:

(a) $\{u_1, \ldots, u_p\} \cap \{j_1, \ldots, j_q\} \neq \emptyset$.

(b) $\{u_1, \ldots, u_p\} \cap \{j_1, \ldots, j_q\} = \emptyset$ i.e. $i_k = u_k$ for all $1 \leq k \leq p$.

In case (a), let s be the largest integer in $\{u_1, \ldots, u_p\} \cap \{j_1, \ldots, j_q\}$. If $u_r = s$, then the maximality of s implies that $u_m = i_m$ for $r+1 \leq m \leq p$. If we now start to reverse slide the empty box in the right-hand diagram in Figure 8.7(a), it is clear that we will undertake a sequence of ordinary reverse vertical slides until the empty box is immediately below $u_r = s$. Using the maximality of s one can show that we have $p(a, s|\overline{s}, \overline{a}) = s - a - 1$ and hence the next slide is a type III^R reverse slide. Since the smallest integer which does *not* appear below the empty box nor above $\overline{s}$ is x, the pair x–$\overline{x}$ gets created during the III^R slide and the pair a–$\overline{a}$ gets deleted. Moreover, the empty box gets inserted between i_r and i_{r+1}. From the previously derived inequalities $t_{k+1} \leq i_k$, the empty box now slides vertically up to the top of the column of i's, and the the situation now appears as it did before the type I forward slide.

In case (b), if we start reverse sliding the empty box in the right-hand diagram of Figure 8.7(a), then we will perform a sequence of ordinary vertical slides until the empty box is immediately below u_0. At that stage, we then have 2 subcases depending on whether $u_0 \geq a$ or $u_0 < a$. In the former case, it is clear that we have the necessary conditions for a III^R slide (since $u_0 < u_1 = i_1$ and so all the elements $\overline{a+1}, \overline{a+2}, \ldots, \overline{u_0 - 1}, \overline{u_0}$ appear immediately above $\overline{a}$). After performing such a slide, the empty box appears at the top of the column of i's and the entries are precisely as they appeared before the forward slide. In the latter case, the situation is ripe for a I^R reverse slide, and again using the fact that $u_m = i_m$ for $1 \leq m \leq p$ after such a reverse slide the situation will be as at the beginning of the forward slide.



Next, consider a type III forward slide, the aftermath of which appears in the diagram on the left. We recall that $\{i_1,\ldots,i_p\} \sqcup \{j_1,\ldots,j_q\} = \{c+1, c+2, \ldots, a-1\}$ and $\{k_0,\ldots,k_r\} \sqcup \{l_1,\ldots,l_s\} = \{a+1, a+2, \ldots, b-1\}$. There are 3 possible situations to consider:

(a) there is an m, n such that $u_m = j_n$.

(b) there is an m, n such that $u_m = l_n$ (we include the case $u_m = l_0 := a$).

(c) the conditions in (a) and (b) do not hold so that $i_m = u_m$ for $1 \le m \le p$.

In case (a), the argument is the same as case (a) above for the type I slide, namely we will slide up vertically until the empty box is just below $u_m := t$ where t is the maximal element in $\{u_1,\ldots,u_p\} \cap \{j_1,\ldots,j_q\}$ at which point a type III^R reverse slide takes place. Note that the entries below t are $i_{m+1},\ldots,i_p,k_0,\ldots,k_r$ while the entries above $\overline{t} = \overline{j_n}$ are $\overline{j_{n+1}},\ldots,\overline{j_q},\overline{a},\overline{l_1},\ldots,\overline{l_s}$. From the remarks above, smallest (barred or unbarred) integer not appearing in the union of these entries is b. Hence after carrying out the reverse slide, the pair b–$\overline{b}$ gets created as required, while the empty box then slides vertically to the top of the column of i's, and the situation is how it appeared before the initial forward III slide.

In case (b), if $u_m = l_n$ for some m, n, then we must have $u_p \ge a$ and hence a similar argument as above shows that the *very first* reverse slide must be a type III^R slide, the effect of which is the same as in the (a) case.

Finally in case (c), as in case (b) for the forward type I slide, we will vertically slide the box until just below the entry u_0, at which stage we must have a reverse I^R or III^R slide according as $u_0 < a$ or $u_0 \ge a$ and the final result is as appears before the forward III slide.

We remark that the cases of reversing the forward II and IV slides are similar.

Acknowledgements

The author thanks the following people for helpful discussions or correspondence: O. Foda, P. Jarvis, M. Kashiwara, T. Miwa, S. Okada, M. Okado, Y. Pugai, A. Schilling, D. Uglov, O. Warnaar. Special thanks are due to T. Welsh for many helpful discussions and comments and for reading through a previous draft. The financial support of the Japan Society for the Promotion of Science is acknowledged.

Note added

After completing this work, we became aware of ref. [16] where the C_n Knuth relations are related to operations on the tableaux of De-Concini-Sheats (which are related to the those of Kashiwara and Nakashima) through Sheat's "symplectic jeu-de-taquin" [25]. The author thanks B. Leclerc for informing him of this related work, and C. Lecouvey for providing it.

References

[1] G. Benkart and J. Stroomer. Tableaux and insertion schemes for spinor representations of the orthogonal Lie algebra $so(2n+1,C)$. *J. Comb. Th. A*, 57:211–237, 1991.

[2] A. Berele. A Schensted-type correspondence for the symplectic group. *J. Comb. Th. Ser. A*, 43:320–328, 1986.

[3] C. De Concini. Symplectic standard tableaux. *Adv. in Math.*, 34:1–27, 1979.

[4] M. Fulmek and C. Krattenthaler. A bijection between Proctor's and Sundaram's odd orthogonal tableaux. *Discr. Math.*, 161:101–120, 1996.

[5] W. Fulton. *Young Tableaux*. Cambridge University Press, 1st edition, 1997.

[6] S-J. Kang, M. Kashiwara, K. Misra, T. Miwa, T. Nakashima, and A. Nakayashiki. Perfect crystals of quantum affine Lie algebras. *Duke Math. J.*, 68:499–607, 1992.

[7] M. Kashiwara. Crystallizing the q-analogue of universal enveloping algebra. *Commun. Math. Phys.*, 133:249–260, 1990.

[8] M. Kashiwara. On crystal bases. *Can. Math. Soc. Conference Proceedings*, 16:155–197, 1995.

[9] M. Kashiwara and T. Nakashima. Crystal graphs for representations of the q-analogue of classical Lie algebras. *J. Alg.*, 165:295–345, 1994.

[10] R. C. King. Weight multiplicities for the classical groups. In *Lecture Notes in Physics*, volume 50, pages 490–499. Springer-Verlag, New York, 1975.

[11] R. C. King and N. G. I. El-Sharkaway. Standard Young tableaux and weight multiplicities of the classical Lie groups. *J. Phys. A*, 16:3153–3177, 1983.

[12] R. C. King and T. A. Welsh. Construction of orthogonal group modules using tableaux. *Lin. Multlin. Alg.*, 33:251–283, 1993.

[13] D. E. Knuth. Permutations, matrices and generalized Young tableaux. *Pacific J. Math.*, 34:709–727, 1970.

[14] K. Koike and I. Terada. Young diagrammatic methods for the restriction of representations of complex classical Lie groups to reductive subgroups of maximal rank. *Adv. in Math.*, 79:104–135, 1990.

[15] A. Lascoux, B. Leclerc, and J-Y. Thibon. Crystal graphs and q-analogs of weight multiplicities for the root system A_n. *Lett. Math. Phys.*, 35:359–374, 1995.

[16] C. Lecouvey. Schensted-type correspondence, plactic monoid and jeu-de-taquin for type C_n. Uni. Caen preprint, 1999.

[17] P. Littelmann. A plactic algebra for semi-simple Lie algebras. *Adv. in Math.*, 124:312–331, 1996.

[18] I. G. Macdonald. *Symmetric functions and Hall polynomials*. Oxford University Press, Oxford, 2nd edition, 1995.

[19] T. Nakashima. Crystal base and a generalization of Littlewood-Richardson rule for the classical Lie algebras. *Commun. Math. Phys.*, 154:215–243, 1993.

[20] S. Okada. A Robinson-Schensted type algorithm for $so(2n, C)$. *J. Alg.*, 143:334–372, 1991.

[21] S. Okada. Robinson-Schensted type algorithms for the spinor representations of the orthogonal Lie algebra $so(2n, C)$. *J. Alg.*, 158:155–200, 1993.

[22] R. A. Proctor. A Schensted algorithm which models tensor representations of the orthogonal group. *Can. J. Math.*, 42:28–49, 1990.

[23] R. A. Proctor. Young tableaux, Gelfand patterns and branching rules for classical Lie groups. *J. Alg.*, 164:299–360, 1994.

[24] C. Schensted. Longest increasing and decreasing subsequences. *Can. J. Math.*, 13:179–191, 1961.

[25] J. T. Sheats. A symplectic jeu-de-taquin bijection between the tableaux of King and of De-Concini. Univ. North Carolina preprint, 1995.

[26] S. Sundaram. The Cauchy identity for $sp(2n)$. *J. Comb. Th. A*, 53:209–238, 1990.

[27] S. Sundaram. Orthogonal tableaux and an insertion algorithm for $SO(2n + 1)$. *J. Comb. Th. A*, 53:239–256, 1990.