

# A promotion operator on rigged configurations

Qiang Wang<sup>1</sup> †

<sup>1</sup> Department of Mathematics, University of California, One Shields Avenue, Davis, CA 95616-8633, U.S.A.

---

In a recent paper, Schilling proposed an operator  $\overline{\text{pr}}$  on unrestricted rigged configurations  $RC$ , and conjectured it to be the promotion operator of the type  $A$  crystal formed by  $RC$ . In this paper we announce a proof for this conjecture.

**Keywords:** promotion operator, rigged configurations, affine crystals

---

## 1 Introduction

Rigged configurations appear in the Bethe Ansatz study of exactly solvable lattice models as combinatorial objects to index the solutions of the Bethe equations [5, 6]. Based on work by Kerov, Kirillov and Reshetikhin [5, 6], it was shown in [7] that there is a statistic preserving bijection  $\Phi$  between Littlewood–Richardson tableaux and rigged configurations. The description of the bijection  $\Phi$  is based on a quite technical recursive algorithm.

Littlewood–Richardson tableaux can be viewed as highest weight crystal elements in a tensor product of Kirillov–Reshetikhin (KR) crystals of type  $A_n^{(1)}$ . KR crystals are affine finite-dimensional crystals corresponding to affine Kac–Moody algebras, in the setting of [7] of type  $A_n^{(1)}$ . The highest weight condition is with respect to the finite subalgebra  $A_n$ . The bijection  $\Phi$  can be generalized by dropping the highest weight requirement on the elements in the KR crystals [1], yielding the set of crystal paths  $\mathcal{P}$ . On the corresponding set of unrestricted rigged configurations  $RC$ , the  $A_n$  crystal structure is known explicitly [12]. One of the remaining open questions is to define the full affine crystal structure on the level of rigged configurations. Given the affine crystal structure on both sides, the bijection  $\Phi$  has a much more conceptual interpretation as an affine crystal isomorphism.

In type  $A_n^{(1)}$ , the affine crystal structure can be defined using the promotion operator  $\text{pr}$ , which corresponds to the Dynkin diagram automorphism mapping node  $i$  to  $i + 1$  modulo  $n + 1$ . On crystals, the promotion operator is defined using jeu-de-taquin [13, 15]. In [12], Schilling proposed an algorithm  $\overline{\text{pr}}$  on  $RC$  and conjectured [12, Conjecture 4.12] that  $\overline{\text{pr}}$  corresponds to the promotion operator  $\text{pr}$  under the bijection  $\Phi$ . Several necessary conditions of promotion operators were established and it was shown that in special cases  $\overline{\text{pr}}$  is the correct promotion operator.

---

†Partially supported by NSF grants DMS–0501101, DMS–0652641, and DMS–0652652.

In this paper, we show in general that  $\Phi \circ \text{pr} \circ \Phi^{-1} = \overline{\text{pr}}$  (i.e.,  $\Phi$  is the intertwiner between  $\text{pr}$  and  $\overline{\text{pr}}$ ):

$$\begin{array}{ccc} \mathcal{P} & \xrightarrow{\Phi} & \text{RC} \\ \text{pr} \downarrow & & \downarrow \overline{\text{pr}} \\ \mathcal{P} & \xrightarrow{\Phi} & \text{RC}. \end{array}$$

Thus  $\overline{\text{pr}}$  is indeed the promotion on RC and  $\Phi$  is an affine crystal isomorphism.

Another reformulation of the bijection from tensor product of crystals to the rigged configurations in terms of the energy function of affine crystals and the inverse scattering formalism for the periodic box ball systems was given in [8, 9, 10, 11].

This paper is organized as follows. In Section 2, we review the definitions of crystal paths and rigged configurations and state the main result in Theorem 2.27. In Section 3, we outline the proof by a running example. Due to the limitation of space, details of the proof are not included. A long version of this paper that contains all technical details is in progress [17] and will appear elsewhere.

## 2 Preliminaries and the main result

Throughout this paper the positive integer  $n$  stands for the rank of the Lie algebra  $A_n$ . Let  $\bar{I} = [n]$  be the index set of the Dynkin diagram of type  $A_n$ . Let  $\mathcal{H} = \bar{I} \times \mathbb{Z}_{>0}$  and define  $B_n$  to be a finite sequence of pairs of positive integers

$$B_n = ((r_1, s_1), \dots, (r_K, s_K))$$

with  $(r_i, s_i) \in \mathcal{H}$  and  $1 \leq i \leq K$ . We use  $L_n$  as a finite (multi-)set of pairs of positive integers

$$L_n = \{(r_1, s_1), \dots, (r_K, s_K)\}$$

with  $(r_i, s_i) \in \mathcal{H}$  and  $1 \leq i \leq K$ . We omit the subscript  $n$  when its value is irrelevant or clear from the context. We also write  $L(B)$  for the underlying (multi-)set of  $B$ . When  $L$  is used this way, it is called the **multiplicity array** of  $B$ .

$B$  represents a sequence of rectangles where the  $i$ -th rectangle is of height  $r_i$  and width  $s_i$ . We sometimes use the phrase “leftmost rectangle” (resp. “rightmost rectangle”) to mean the first (resp. last) pair in the list. We use  $|B| = K$  for the number of pairs in  $B$ , and  $B[i] = (r_i, s_i)$  as the  $i$ -th pair in  $B$ . Similarly, we can think of  $L$  as a (multi-)set of rectangles. It is sometimes useful (especially in the setting of rigged configurations) to consider the multiplicity of a given  $(a, i) \in \mathcal{H}$  in  $L$  by setting  $L_i^{(a)} = \#\{(r, s) \in L \mid r = a, s = i\}$ .

Given a sequence of rectangles  $B$ , we will use the following operations for successively removing boxes from it. In the following subsections, we define the set of paths  $\mathcal{P}(B)$  and rigged configurations  $\text{RC}(L(B))$ , and discuss the analogous operations defined on  $\mathcal{P}(B)$  and  $\text{RC}(L(B))$ . They are used to define the bijection  $\Phi$  between  $\mathcal{P}(B)$  and  $\text{RC}(L(B))$  recursively. The proof exploits this recursion.

**Definition 2.1** [1, Section 4.1,4.2].

1. If  $B = ((1, 1), B')$ , let  $\text{lh}(B) = B'$ . This operation is called **left-hat**.
2. If  $B = ((r, s), B')$  with  $s \geq 2$ , let  $\text{ls}(B) = ((r, 1), (r, s - 1), B')$ . This operation is called **left-split**.

3. If  $B = ((r, 1), B')$  with  $r \geq 2$ , let  $\text{lb}(B) = ((1, 1), (r - 1, 1), B')$ . This operation is called **box-split**.

### 2.1 Inhomogeneous lattice paths

**Definition 2.2** Given  $(r, s) \in \mathcal{H}$ , define  $\mathcal{P}_n(r, c)$  to be the set of semi-standard Young tableaux of (rectangular) shape  $(s^r)$  over the alphabet  $\{1, 2, \dots, n + 1\}$ .

Recall that for each semi-standard Young tableau  $t$ , we can associate an ambient weight  $\text{wt}(t) = (\lambda_1, \lambda_2, \dots, \lambda_{n+1})$  where  $\lambda_i$  is the number of times that  $i$  appears in  $t$ . Moreover,  $\mathcal{P}_n(r, s)$  is endowed with a type  $A_n$ -crystal structure, with the Kashiwara operator  $e_a, f_a$  for  $1 \leq a \leq n$  defined by the signature rule. For a detailed discussion see for example [4, Chapters 7 and 8].

**Definition 2.3** Given a sequence  $B_n$  as defined above,  $\mathcal{P}_n(B_n) = \mathcal{P}_n(r_1, s_1) \otimes \dots \otimes \mathcal{P}_n(r_K, s_K)$ .

As a set  $\mathcal{P}_n(B_n)$  is a sequence of rectangular semi-standard Young tableaux. It is also endowed with a crystal structure through the tensor product rule. The Kashiwara operators  $e_a, f_a$  for  $1 \leq a \leq n$  naturally extend from semi-standard tableaux to a list of tableaux using signature rule. For  $b_1 \otimes b_2 \otimes \dots \otimes b_K \in \mathcal{P}_n(B_n)$ ,  $\text{wt}(b_1 \otimes b_2 \otimes \dots \otimes b_K) = \text{wt}(b_1) + \text{wt}(b_2) + \dots + \text{wt}(b_K)$ . We note here that the convention of tableaux tensor product in this paper follows that of [1, Section 2], which is opposite to the Kashiwara's original convention [2].

**Definition 2.4** Let  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_{n+1})$  be an list of nonnegative integers, define

$$\mathcal{P}_n(B_n, \lambda) = \{p \in \mathcal{P}_n(B_n) \mid \text{wt}(p) = \lambda\}$$

**Example 2.5** Let  $B_3 = ((2, 2), (1, 2), (3, 1))$ . Let

$$p = \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 2 & 3 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 1 & 2 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 1 \\ \hline 2 \\ \hline 4 \\ \hline \end{array}$$

Then  $p$  is an element of  $\mathcal{P}(B_3)$  and  $\text{wt}(p) = (3, 4, 1, 1)$ .

As in the above example, we often omit the subscript  $n$ , writing  $\mathcal{P}$  instead of  $\mathcal{P}_n$  when  $n$  is irrelevant or clear from the discussion.

We often refer to a rectangular tableau just as a ‘‘rectangle’’ when there is no ambiguity, for example, the leftmost rectangle in  $p$  of above example is the tableau:

$$\begin{array}{|c|c|} \hline 1 & 2 \\ \hline 2 & 3 \\ \hline \end{array}.$$

The following maps on  $\mathcal{P}(B)$  are the counterparts to the maps  $\text{lh}$ ,  $\text{lb}$  and  $\text{ls}$  defined on  $B$ .

**Definition 2.6** [1, Sections 4.1,4.2].

1. Let  $b = c \otimes b' \in \mathcal{P}((1, 1), B')$ . Then  $\text{lh}(b) = b' \in \mathcal{P}(B')$ .
2. Let  $b = c \otimes b' \in \mathcal{P}((r, s), B')$ , where  $c = c_1 c_2 \dots c_s$  and  $c_i$  denotes the  $i$ -th column of  $c$ . Then  $\text{ls}(b) = c_1 \otimes c_2 \dots c_s \otimes b'$ .

$$3. \text{ Let } b = \begin{array}{|c|} \hline b_1 \\ \hline b_2 \\ \hline \vdots \\ \hline b_r \\ \hline \end{array} \otimes b' \in \mathcal{P}((r, 1), B'), \text{ where } b_1 < \dots < b_r. \text{ Then } \text{lb}(b) = \boxed{b_r} \otimes \begin{array}{|c|} \hline b_1 \\ \hline \vdots \\ \hline b_{r-1} \\ \hline \end{array} \otimes b'.$$

### 2.2 Rigged configurations

This section follows [12, Section 3.1]. In this paper we only consider rigged configurations of type  $A_n^{(1)}$ .

The (highest-weight) rigged configurations are indexed by the multiplicity array  $L$  and a dominant weight  $\Lambda$ . The sequence of partitions  $\nu = \{\nu^{(a)} \mid a \in \bar{I}\}$  is a  $(L, \Lambda)$ -**configuration** if

$$\sum_{(a,i) \in \mathcal{H}} im_i^{(a)} \alpha_a = \sum_{(a,i) \in \mathcal{H}} iL_i^{(a)} \Lambda_a - \Lambda, \tag{2.1}$$

where  $m_i^{(a)}$  is the number of parts of length  $i$  in partition  $\nu^{(a)}$ ,  $\alpha_a$  is the  $a$ -th simple root and  $\Lambda_a$  is the  $a$ -th fundamental weight. Denote the set of all  $(L, \Lambda)$ -configurations by  $C(L, \Lambda)$ . The **vacancy number** of a configuration is defined as

$$p_i^{(a)} = \sum_{j \geq 1} \min(i, j)L_j^{(a)} - \sum_{(b,j) \in \mathcal{H}} (\alpha_a | \alpha_b) \min(i, j)m_j^{(b)}.$$

Here  $(\cdot | \cdot)$  is the normalized invariant form on the weight lattice  $P$  such that  $A_{ab} = (\alpha_a | \alpha_b)$  is the Cartan matrix (of type  $A_n$  in our case). The  $(L, \Lambda)$ -configuration  $\nu$  is **admissible** if  $p_i^{(a)} \geq 0$  for all  $(a, i) \in \mathcal{H}$ , and the set of admissible  $(L, \Lambda)$ -configurations is denoted by  $\overline{C}(L, \Lambda)$ .

A partition  $p$  can be viewed as a linear ordering  $(p, \gg)$  of a finite multiset of positive integers, where elements of different values are ordered by their value, and elements the same value are given an arbitrary ordering. Implicitly, when we draw a Young diagram of  $p$ , we are giving such an ordering. A labeling of a partition  $p$  is then a map  $J^{(p)} : (p, \gg) \rightarrow \mathbb{Z}_{\geq 0}$  satisfying that if  $i, j \in p$  are of the same value and  $i \gg j$  then  $J^{(p)}(i) \geq J^{(p)}(j)$  as integers. The pairs  $(x, J^{(p)}(x))$  are referred to as strings;  $x$  is referred to as the length or size of the string and  $J^{(p)}(x)$  as the **label** or **quantum number**.

A rigging  $J$  of an (admissible)  $(L, \Lambda)$ -configuration  $\nu$  is a sequence of maps  $J = (J^{(a)})$ , each  $J^{(a)}$  is a labeling of the partitions  $\nu^{(a)}$  with the extra requirement that for any part  $i \in \nu^{(a)}$

$$0 \leq J^{(a)}(i) \leq p_i^{(a)}.$$

The difference  $cJ^{(a)}(i) = p_i^{(a)} - J^{(a)}(i)$  is referred to as the **colabel** or **coquantum number** of the part  $i$ . A part is said to be **singular** if its colabel is 0. Since  $cJ$  and  $J$  uniquely determine each other, we sometimes refer to a string by  $(x, cJ^{(p)}(x))$  when it is more convenient.

**Definition 2.7** The pair  $(\nu, J)$  described above is called a (restricted-)rigged configuration. The set of all rigged  $(L, \Lambda)$ -configurations is denoted by  $\overline{RC}_n(L, \Lambda)$ . In addition, define  $\overline{RC}(L) = \bigcup_{\Lambda \in P^+} \overline{RC}(L, \Lambda)$ .

The equation (2.1) provides an obvious way of defining weight function on  $\overline{RC}(L)$ . Namely, for  $rc \in \overline{RC}(L)$

$$\text{wt}(rc) = \sum_{(a,i) \in \mathcal{H}} iL_i^{(a)} \Lambda_a - \sum_{(a,i) \in \mathcal{H}} im_i^{(a)} \alpha_a \tag{2.2}$$

When working with rigged configuration, it is often convenient to take the fundamental weights as basis for the weight space. On the other hand, we presented weights in the ambient weight space when working with lattice paths. Conceptually, this distinction is not necessary, as weights can be considered as abstract vectors in the weight space. Identifying the fundamental weight  $\Lambda_i$  with  $(1^i, 0^{n+1-i})$  we can switch from one representation to the other.

**Remark 2.8** From the above definition, it is clear that  $\text{RC}(L)$  is not sensitive to the ordering of the rectangles in  $L$ .

**Definition 2.9** Let  $L$  be a multiplicity array. Define the set of **unrestricted rigged configurations**  $\text{RC}(L)$  as the closure of  $\overline{\text{RC}}(L)$  under the operators  $f_a, e_a$  for  $a \in \bar{I}$ , with  $f_a, e_a$  given by:

1. Define  $e_a(\nu, J)$  by removing a box from a string of length  $k$  in  $(\nu, J)^{(a)}$  leaving all colabels fixed and increasing the new label by one. Here  $k$  is the length of the string with the smallest negative rigging of smallest length. If no such string exists,  $e_a(\nu, J)$  is undefined.
2. Define  $f_a(\nu, J)$  by adding a box to a string of length  $k$  in  $(\nu, J)^{(a)}$  leaving all colabels fixed and decreasing the new label by one. Here  $k$  is the length of the string with the smallest non positive rigging of largest length. If no such string exists, add a new string of length one and label  $-1$ . If the result is not a valid unrestricted rigged configuration  $f_a(\nu, J)$  is undefined.

The weight function (2.2) defined on  $\overline{\text{RC}}(L)$  extends to  $\text{RC}(L)$  with no change.

As their names suggested,  $f_a$  and  $e_a$  are indeed the Kashiwara operators with respect to the weight function above, and define a crystal structure on  $\text{RC}(L)$ . This was proved in [12].

From the definition of  $f_a$ , it is clear that the labels of parts in an unrestricted rigged configuration may be negative. It is natural to ask what shapes and labels can appear in an unrestricted rigged configuration. There is an explicit characterization of  $\text{RC}(L)$  which answers this question [1, Section 3]. The statement is rather long and is not directly used by our proof, so we will just give rough outline and leave the interested reader to the original paper for detail: In the definition of  $\overline{\text{RC}}(L)$ , we required that the vacancy number associated to each part non-negative. We drop this requirement for  $\text{RC}(L)$ . Yet a vacancy numbers in  $\text{RC}(L)$  still serves as the upper bound of the label, much like the role a vacancy number plays for a restricted rigged configuration. For restricted rigged configurations, the lower bound for the label of a part is uniformly 0. For unrestricted rigged configurations, this is not the case. The characterization gives a way on how to find lower bound for each part.

**Example 2.10** Here is an example on how we normally visualize a restricted/unrestricted rigged configuration. Let  $B_3 = ((2, 2), (1, 2), (3, 1))$ , and  $L_3 = L(B_3)$ . Then

$$\text{rc} = \begin{array}{ccc} \square \square & -1 & \square \quad 1 \quad \square \quad -1 \end{array}$$

is an element of  $\text{RC}(L_3, -\Lambda_1 + 3\Lambda_2)$ .

In this example, the sequence of partitions  $\nu$  is  $((2), (1), (1))$ . The number that follows each part is the label assigned to this part by  $J$ . The vacancy numbers associated to these parts are  $p_2^{(1)} = -1, p_1^{(2)} = 1$ , and  $p_1^{(3)} = 0$ . Note that the labels are all less than or equal to the vacancy numbers, in the case that they are equal, e.g. parts in  $\nu^{(1)}$  and  $\nu^{(2)}$ , those parts are called singular as restricted rigged configuration.

In this example  $\text{rc} \in \text{RC} \setminus \overline{\text{RC}}$ , the idea is the same for  $\text{rc} \in \overline{\text{RC}}$ .

The following maps on  $\text{RC}(L(B))$  are the counterparts to the lh, lb and ls map defined on  $B$ .

**Definition 2.11** [1, Section 4.1,4.2].

1. Let  $rc = (\nu, J) \in \text{RC}(L(B))$ . Then  $\overline{\text{lh}}(rc) \in \text{RC}(L(\text{lh}(B)))$  is defined by: first set  $\ell^{(0)} = 1$  and then repeat the following process for  $a = 1, 2, \dots, n - 1$  or until stopped. Find the smallest index  $i \geq \ell^{(a-1)}$  such that  $J^{(a)}(i)$  is singular. If no such  $i$  exists, set  $\text{rk}(\nu, J) = a$  and stop. Otherwise set  $\ell^{(a)} = i$  and continue with  $a + 1$ . Set all undefined  $\ell^{(a)}$  to  $\infty$ .

The new rigged configuration  $(\tilde{\nu}, \tilde{J}) = \overline{\text{lh}}(\nu, J)$  is obtained by removing a box from the selected strings and making the new strings singular again.

2. Let  $rc = (\nu, J) \in \text{RC}(L(B))$ . Then  $\overline{\text{ls}}(rc) \in \text{RC}(L(\text{ls}(B)))$  is the same as  $(\nu, J)$ .
3. Let  $rc = (\nu, J) \in \text{RC}(L(B))$ . Then  $\overline{\text{lb}}(rc) \in \text{RC}(L(\text{lb}(B)))$  is defined by adding singular strings of length 1 to  $(\nu, J)^{(a)}$  for  $1 \leq a < r$ . Note that the vacancy numbers remain unchanged under  $\overline{\text{lb}}$ .

### 2.3 The bijection between $\mathcal{P}(B)$ and $\text{RC}(L(B))$

The map  $\Phi : \mathcal{P}(B, \lambda) \rightarrow \text{RC}(L(B), \lambda)$  is defined recursively by various commutative diagrams. Note that it is possible to go from  $B = ((r_1, s_1), (r_2, s_2), \dots, (r_K, s_K))$  to the empty crystal via successive application of lh, ls and lb. For further details see [1, Section 4].

**Definition 2.12** Define the map  $\Phi : \mathcal{P}(B, \lambda) \rightarrow \text{RC}(L(B), \lambda)$  such that the empty path maps to the empty rigged configuration and such that the following conditions hold:

1. Suppose  $B = ((1, 1), B')$ . Then the following diagram commutes:

$$\begin{array}{ccc}
 \mathcal{P}(B, \lambda) & \xrightarrow{\Phi} & \text{RC}(L(B), \lambda) \\
 \text{lh} \downarrow & & \downarrow \delta \\
 \bigcup_{\mu \in \lambda^-} \mathcal{P}(\text{lh}(B), \mu) & \xrightarrow{\Phi} & \bigcup_{\mu \in \lambda^-} \text{RC}(L(\text{lh}(B)), \mu)
 \end{array}$$

where  $\lambda^-$  is the set of all partitions obtained from partition  $\lambda$  by removing a box.

2. Suppose  $B = ((r, s), B')$  with  $s \geq 2$ . Then the following diagram commutes:

$$\begin{array}{ccc}
 \mathcal{P}(B, \lambda) & \xrightarrow{\Phi} & \text{RC}(L(B), \lambda) \\
 \text{ls} \downarrow & & \downarrow \overline{\text{ls}} \\
 \mathcal{P}(\text{ls}(B), \lambda) & \xrightarrow{\Phi} & \text{RC}(L(\text{ls}(B)), \lambda)
 \end{array}$$

3. Suppose  $B = ((r, 1), B')$  with  $r \geq 2$ . Then the following diagram commutes:

$$\begin{array}{ccc}
 \mathcal{P}(B, \lambda) & \xrightarrow{\Phi} & \text{RC}(L(B), \lambda) \\
 \text{lb} \downarrow & & \downarrow \overline{\text{lb}} \\
 \mathcal{P}(\text{lb}(B), \lambda) & \xrightarrow{\Phi} & \text{RC}(L(\text{lb}(B)), \lambda)
 \end{array}$$

## 2.4 Promotion operators

The **promotion** operator  $\text{pr}$  on  $\mathcal{P}_n(B)$  is defined in [15, page 164]. For the purpose of our proof, we will phrase it as a composition of one **lifting** operator and then several **sliding** operators defined on  $\mathcal{P}_n(B)$ .

**Definition 2.13** *The lifting operator  $l$  on  $\mathcal{P}_n(B)$  lifts  $p \in \mathcal{P}_n(B)$  to  $l(p) \in \mathcal{P}_{n+1}(B)$  by adding 1 to each box in each rectangle of  $p$ .*

**Definition 2.14** *Given  $p \in \mathcal{P}_{n+1}(B)$ , the sliding operator  $\rho$  is defined as the following algorithm: find in  $p$  the rightmost rectangle that contains  $n + 2$ , remove one appearance of  $n + 2$ , apply jeu-de-taquin on this rectangle to move the empty box to the opposite corner, fill in 1 in this empty box. If no rectangle contains  $n + 2$ , then  $\rho$  is the identity map.*

**Definition 2.15**

$$\text{pr}(p) = \rho^m \circ l(p)$$

where  $m$  is the total number of  $n + 2$  in  $p$ .

This promotion operator is used to construct the affine crystal structure on  $\mathcal{P}(B)$ . See [15, page 164] for a detailed discussion.

The proposed promotion operator  $\overline{\text{pr}}$  on  $\text{RC}_n(L)$  is defined in [12, Definition 4.8]. To draw the parallel with  $\text{pr}$  we will phrase it as a composition of one lifting operator and then several sliding operators defined on  $\text{RC}(L)$ .

**Definition 2.16** *The lifting operator  $\bar{l}$  on  $\text{RC}_n(L)$  lifts  $\text{rc} \in \text{RC}_n(L)$  to  $\bar{l}(\text{rc}) \in \text{RC}_{n+1}(L)$  by setting  $\bar{l}(\text{rc}) = f_1^{\lambda_1} f_2^{\lambda_2} \cdots f_{n+1}^{\lambda_{n+1}}(\text{rc})$ , where  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_{n+1})$  is the ambient weight of  $\text{rc}$   $\sum_{i \in [n+1]} \lambda_i = \sum_{(r,s) \in L} r \cdot s$ . Notice here we use the fact that  $\text{RC}_n(L)$  is naturally embedded in  $\text{RC}_{n+1}(L)$  by simply treating the  $(n + 1)_{st}$  partition  $v^{(n+1)}$  to be  $\emptyset$ .*

**Definition 2.17** [12, Definition 4.8] *Given  $\text{rc} \in \text{RC}_{n+1}(L)$ , the sliding operator  $\bar{\rho}$  is defined as the following algorithm: Find the smallest singular string in  $\text{rc}^{(n+1)}$ . Let the length be  $\ell^{(n+1)}$ . Repeatedly find the smallest singular string in  $(\nu', J')^{(k)}$  of length  $\ell^{(k)} \geq \ell^{(k+1)}$  for all  $1 \leq k < n + 1$ . Shorten the selected strings by one and make them singular again.*

**Definition 2.18**

$$\overline{\text{pr}}(\text{rc}) = \bar{\rho}^m \circ \bar{l}(\text{rc})$$

where  $m$  is the number of boxes in  $\text{rc}^{(n+1)}$ .

**Remark 2.19** *It is a easy matter to show that  $\bar{l} = \Phi(l)$  (that is  $\bar{l} = \Phi \circ l \circ \Phi^{-1}$ ). Indeed, we could have defined  $l(p) = f_1^{\lambda_1} f_2^{\lambda_2} \cdots f_{n+1}^{\lambda_{n+1}}(p)$ , where  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_{n+1})$  is the weight of  $p$ .*

There is a question in Definition 2.14 on whether a sequence of  $m$   $\rho$  operators can be always applied. The same question can be asked for Definition 2.17. The following are examples on how things could go wrong:

**Example 2.20** *Let*

$$p = \begin{array}{|c|c|} \hline 1 & 1 \\ \hline 4 & 4 \\ \hline \end{array} \in \mathcal{P}_3((2, 2)) \tag{2.3}$$

If we try to construct  $\rho(p)$ , we realize that after removing a copy of 4 and move the empty box to the upper left corner we get  $\begin{array}{|c|c|} \hline & 1 \\ \hline 1 & 4 \\ \hline \end{array}$ , and filling the empty box with 1 will violate the column-strictness of semi-standard Young tableaux.

On the RC side, let

$$rc = \begin{array}{|c|c|} \hline & \\ \hline & \\ \hline \end{array} 0 \quad \begin{array}{|c|c|} \hline & \\ \hline & \\ \hline \end{array} 0 \quad \emptyset \in RC_3((2, 2))$$

We see that  $\bar{\rho}(rc)$  is not well-defined.

Therefore,  $\rho$  and  $\bar{\rho}$  are partial functions on  $\mathcal{P}_{n+1}$  and  $RC_{n+1}$ . This, however, will not cause problems in our discussion because of the following two remarks.

**Remark 2.21**  $\rho$  is well-defined on  $\rho^k(\text{Img}(l))$  for any  $k$ . This follows from the well known fact that if  $T$  is a semi-standard rectangular tableau, and if we remove all cells that contain the largest number (which is a horizontal strip in the last row) and apply "jeu de taquin" to move these empty cells to the upper left corner, then these empty cells form a horizontal strip.

Thus we could have just restricted the domain of  $\rho$  to:

**Definition 2.22**

$$\text{Dom}(\rho) = \bigcup_{k=0,1,2\dots} \rho^k(\text{Img}(l)) \tag{2.4}$$

**Remark 2.23** It is not known at this stage that  $\bar{\rho}$  is fully defined on  $\Phi(\text{Dom}(\rho))$ . In fact, it is a consequence of our proof.

### 2.5 Combinatorial R-matrix and right-split

Let  $B = ((r_1, s_1), \dots, (r_K, s_K))$  be a sequence of rectangles, and let  $\sigma \in S_K$  be a permutation of  $K$  letters. The  $R$ -matrix is the affine crystal isomorphism  $R : \mathcal{P}(B) \rightarrow \mathcal{P}(\sigma(B))$ , which sends  $u_1 \otimes \dots \otimes u_K$  to  $u_{\sigma(1)} \otimes \dots \otimes u_{\sigma(K)}$ , where  $u_i \in \mathcal{P}(r_i, s_i)$  is the unique tableau of content  $(s_i^{r_i})$  and  $\sigma(B) = ((r_{\sigma(1)}, s_{\sigma(1)}), \dots, (r_{\sigma(K)}, s_{\sigma(K)}))$ . It was shown in [7, Lemma 8.5] that  $\Phi \circ R = \bar{R} \circ \Phi$ , where  $\bar{R}$  is the identity map. Together with the fact that  $R$  preserves the crystal structure [12] this implies the following result.

**Remark 2.24** The following diagram commutes:

$$\begin{array}{ccc} \mathcal{P}(B) & \xrightarrow{\Phi} & RC(L(B)) \\ R \downarrow & & \downarrow \bar{R}=\text{id} \\ \mathcal{P}(\sigma(B)) & \xrightarrow{\Phi} & RC(L(\sigma(B))). \end{array}$$

**Remark 2.25** Since any  $\sigma \in S_K$  is generated by simple transpositions, it suffices to work with cases where  $|B|=2$ , and  $\sigma$  being the only simple transposition is  $S_2$ .

**Definition 2.26**  $rs, \bar{rs}$  are called **right-split**.  $rs$  operates on sequences of rectangles as follow: Let  $B = ((r_1, s_1), \dots, (r_K, s_K))$ , and suppose  $s_K > 1$  (i.e, the rightmost rectangle is not a single column). Then  $rs(B) = ((r_1, s_1), \dots, (r_K, s_K - 1), (r_K, 1))$ , that is,  $rs$  splits one column off the rightmost rectangle.



$\overline{rs}$  operates on  $RC(L(B))$  as follow: If  $rc \in RC(L(B))$ , then  $\overline{rs}(rc) \in RC(L(rs(B)))$  is obtained by increasing the riggings by 1 for all parts in  $\nu^{(r_K)}$  of size less than  $s_K$ . Observe that this will leave the co-riggings of all parts unchanged.

$rs$ , which operates on  $\mathcal{P}(B)$ , is defined to be  $\Phi \circ \overline{rs} \circ \Phi^{-1}$ .

### 2.6 The main result

We now state the main result of this paper.

**Theorem 2.27** Let  $B = ((r_1, s_1), \dots, (r_K, s_K))$  be a sequence of rectangles, and  $\mathcal{P}(B)$ ,  $RC(L(B))$ ,  $\Phi$ ,  $pr$ , and  $\overline{pr}$  as given as above. Then the following diagram commutes:

$$\begin{array}{ccc}
 \mathcal{P}(B) & \xrightarrow{\Phi} & RC(L(B)) \\
 pr \downarrow & & \downarrow \overline{pr} \\
 \mathcal{P}(B) & \xrightarrow{\Phi} & RC(L(B)).
 \end{array} \tag{2.5}$$

## 3 The outline of the proof

In this part, we outline the proof by a running example. The details of the proofs can be found in [17].

Recall we want to prove (2.5). By Remark 2.19 it suffices for us to show that the following diagram commutes:

$$\begin{array}{ccc}
 \text{Dom}(\rho) & \xrightarrow{\Phi} & \Phi(\text{Dom}(\rho)) \\
 \rho \downarrow & & \downarrow \overline{\rho} \\
 \text{Dom}(\rho) & \xrightarrow{\Phi} & \Phi(\text{Dom}(\rho)).
 \end{array}$$

In particular, we need to show that  $\overline{\rho}$  is defined on  $\Phi(\text{Dom}(\rho))$ .

As an abbreviation, for any  $p \in \text{Dom}(\rho)$ , we use  $\mathcal{D}(p)$  to mean the following statement: “ $\overline{\rho}(\Phi(p))$  is well-defined and the diagram

$$\begin{array}{ccc}
 p & \xrightarrow{\Phi} & \bullet \\
 \rho \downarrow & & \downarrow \overline{\rho} \\
 \bullet & \xrightarrow{\Phi} & \bullet
 \end{array}$$

commutes”.

For  $p, q \in \text{Dom}(\rho)$  we write  $\mathcal{D}(p) \rightsquigarrow \mathcal{D}(q)$  to mean that  $\mathcal{D}(p)$  reduces to  $\mathcal{D}(q)$ , that is,  $\mathcal{D}(q)$  is a sufficient condition for  $\mathcal{D}(p)$ .

We will use the following  $p \in \mathcal{P}_3((2, 2), (3, 2), (2, 2))$  as the starting point of the running example:

$$p = \begin{array}{|c|c|} \hline 2 & 2 \\ \hline 4 & 4 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 2 & 3 \\ \hline \end{array}.$$

After lifting to  $\mathcal{P}_4$  we have:

$$l(p) = \begin{array}{|c|c|} \hline 3 & 3 \\ \hline 5 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array} \in \text{Dom}(\rho).$$

Our goal is to show  $\mathcal{D}(l(p))$  by a sequence of reductions. Note that the rightmost 5 (which is  $n+2$  where  $n=3$ ) appears in the second rectangle. Thus  $\rho$  acts on the second rectangle. The motivation behind our reductions is to try to get rid of boxes from the left and make  $\rho$  act on the leftmost rectangle:

**Step 1**

$$\mathcal{D}\left(\begin{array}{|c|c|} \hline 3 & 3 \\ \hline 5 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array}\right) \stackrel{\text{ls}}{\sim} \mathcal{D}\left(\begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array}\right)$$

This is called a ls-reduction.

**Step 2**

$$\mathcal{D}\left(\begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array}\right) \stackrel{\text{lb}}{\sim} \mathcal{D}\left(\begin{array}{|c|} \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 3 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array}\right)$$

This is called a lb-reduction.

**Step 3**

$$\mathcal{D}\left(\begin{array}{|c|} \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 3 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array}\right) \stackrel{\text{lh}}{\sim} \mathcal{D}\left(\begin{array}{|c|} \hline 3 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array}\right)$$

This is called a lh-reduction.

**Step 4** Another application of lh-reduction.

$$\mathcal{D}\left(\begin{array}{|c|} \hline 3 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array}\right) \stackrel{\text{lh}}{\sim} \mathcal{D}\left(\begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array}\right)$$

We repeat above reductions until the rightmost tableau containing 5 becomes the first tableau in the list. After that we want to further simplify the list, if possible, to get rid of boxes from right by pushing them column-by-column to the left using the  $R$ -matrix map  $R$ , until we reach the place where can prove  $\mathcal{D}(\bullet)$  directly:

**Step 8**

$$\mathcal{D}\left(\begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline \end{array}\right) \stackrel{\text{rs}}{\sim} \mathcal{D}\left(\begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 3 \\ \hline 4 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 2 \\ \hline 3 \\ \hline \end{array}\right)$$

This is called a rs-reduction.

**Step 9**

$$\mathcal{D}\left(\begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 3 \\ \hline 4 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 2 \\ \hline 3 \\ \hline \end{array}\right) \overset{R}{\rightsquigarrow} \mathcal{D}\left(\begin{array}{|c|} \hline 3 \\ \hline 4 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 2 \\ \hline 3 \\ \hline \end{array}\right)$$

This is called a R-reduction.

Now since the rectangle that  $\rho$  acts on is no longer the leftmost one we can go back to Step 1. Repeat above steps until  $\rho$  acts on the leftmost rectangle again, then we need one more R-reduction:

**Step 13**

$$\mathcal{D}\left(\begin{array}{|c|c|} \hline 2 & 3 \\ \hline 3 & 4 \\ \hline 4 & 5 \\ \hline \end{array} \otimes \begin{array}{|c|} \hline 2 \\ \hline 3 \\ \hline \end{array}\right) \overset{R}{\rightsquigarrow} \mathcal{D}\left(\begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 2 \\ \hline 3 & 3 \\ \hline 4 & 4 \\ \hline \end{array}\right)$$

Repeating, we will eventually reach one of the following two cases (not mutually exclusive):

- Base case 1:  $p$  is a single rectangle that contains  $n + 2$  (which is 5 in the above example); or
- Base case 2:  $p = S \otimes q$  where  $S$  is a single column that contains  $n + 2$  (which is 5 in the above example), and  $n + 2$  does not appear in  $q$ .

In the above example, we reached the second case.

**Step 14** Now we have to prove this base case directly:

$$\mathcal{D}\left(\begin{array}{|c|} \hline 3 \\ \hline 5 \\ \hline \end{array} \otimes \begin{array}{|c|c|} \hline 2 & 2 \\ \hline 3 & 3 \\ \hline 4 & 4 \\ \hline \end{array}\right)$$

## Acknowledgements

The author would like to thank Anne Schilling for suggesting this problem, and kindly providing financial support for this work.

The investigation of this problem depended heavily on MuPAD-Combinat in forming and experimenting ideas. The author would like to thank Nicolas M. Thiéry for his always-there-to-help on MuPAD-Combinat.

## References

- [1] L. Deka, A. Schilling, *New fermionic formula for unrestricted Kostka polynomials*, J. Combinatorial Theory, Series A **113** (2006) 1435–1461.
- [2] M. Kashiwara, T. Nakashima, *Crystal graphs for representations of the q-analogue of classical Lie algebras*, J. Algebra **165** (1994), no. 2, 295–345.
- [3] William Fulton *Young Tableaux* London Mathematical Society Student Texts 35.
- [4] J. Hong, S.-J. Kang, *Introduction to quantum groups and crystal bases*, Graduate Studies in Mathematics, **42**, American Mathematical Society, Providence, RI, 2002. xviii+307 pp.

- [5] S. V. Kerov, A. N. Kirillov and N. Yu. Reshetikhin *Combinatorics, the Bethe ansatz and representations of the symmetric group*, Zap.Nauchn. Sem. (LOMI) **155** (1986) 50–64. (English translation: J. Sov. Math. **41** (1988) 916–924.)
- [6] A. N. Kirillov and N. Y. Reshetikhin, *The Bethe Ansatz and the combinatorics of Young tableaux*, J. Soviet Math. **41** (1988) 925–955.
- [7] A.N. Kirillov, A. Schilling, M. Shimozono, *A bijection between Littlewood-Richardson tableaux and rigged configurations*, Selecta Mathematica (N.S.) **8** (2002) 67–135.
- [8] A. Kuniba, M. Okado, R. Sakamoto, T. Takagi, Y. Yamada, *Crystal interpretation of Kerov-Kirillov-Reshetikhin bijection*, Nucl. Phys. B **740** (2006) 299–327.
- [9] R. Sakamoto, *Crystal Interpretation of Kerov-Kirillov-Reshetikhin Bijection II. Proof for  $sl_n$  Case*, J.Algebr.Comb. **27** (2008) 55–98.
- [10] R. Sakamoto, *Kirillov–Schilling–Shimozono bijection as energy functions of crystals*, International Mathematics Research Notices (2009) 2009: 579-614.
- [11] R. Sakamoto, *Finding Rigged Configurations From Paths*, RIMS Kokyuroku Bessatsu **B11** (2009) 1–17.
- [12] A. Schilling, *Crystal structure on rigged configurations*, International Mathematics Research Notices, Volume 2006, Article ID 97376, Pages 1–27.
- [13] M. P. Schützenberger, *Promotion des morphismes d’ensembles ordonnés*, Discrete Math. **2** (1972) 73–94
- [14] A. Schilling, S.O. Warnaar, *Inhomogeneous lattice paths, generalized Kostka polynomials and  $A_{n-1}$  supernomials*, Commun. Math. Phys. **202** (1999) 359–401.
- [15] M. Shimozono, *Affine type A crystal structure on tensor products of rectangles, Demazure characters, and nilpotent varieties*, J. Algebraic Combin. **15** (2002), no. 2, 151–187.
- [16] Reiho Sakamoto, *Kirillov-Schilling-Shimozono bijection as energy functions of crystals*, preprint arXiv:0711.4185v1
- [17] A. Schilling, Q. Wang, *Promotion operator on rigged configurations of type A*, In preparation. Draft available by emailing the authors